Материалы международной научной конференции по проблемам безопасности и противодействия терроризму

Интеллектуальный Центр Московского государственного университета им. М. В. Ломоносова,
2–3 ноября 2005 года

К 250-летнему юбилею Московского государственного университета имени М. В. Ломоносова

Московский государственный университет им. М. В. Ломоносова

Институт проблем информационной безопасности МГУ

Академия криптографии Российской Федерации

Материалы международной научной конференции по проблемам безопасности и противодействия терроризму

(Интеллектуальный Центр МГУ, 2–3 ноября 2005 г.)

ББК 32.81В6
М34

Организация и проведение четвертой Общероссийской конференции «Математика и безопасность информационных технологий» (МаБИТ-05) было поддержано грантом РФФИ № 05-01-10141-г.

РФФИ

М34 **Материалы** международной научной конференции по проблемам безопасности и противодействия терроризму. Интеллектуальный Центр МГУ. 2–3 ноября 2005 г. — М.: МЦНМО, 2006.

В подготовке и проведении конференции участвовали:
Университет штата Нью-Йорк;
Университет им. Генриха Гейне;
Университет Кембриджа;
Аппарат Совета Безопасности Российской Федерации;
Программа НАТО «Безопасность через науку»;
ОАО «Газпром»;
Академия информационных систем.

© Коллектив авторов, 2006.
© МЦНМО, 2006.

Я рад приветствовать участников международной научной конференции по проблемам безопасности и противодействия терроризму. В программу конференции включены доклады по широкому спектру научных направлений — математических, компьютерных, философских, политологических, психологических. И все они посвящены поиску ответа на один вопрос: как сделать жизнь человека в современном мире более безопасной?

XXI век принес человечеству новые глобальные процессы, вызовы и угрозы. Стремительное развитие информационно-телекоммуникационных технологий приводит к массовому освоению информационного пространства, а в этом есть и свои «плюсы», и свои «минусы». Научный анализ этих «плюсов» и «минусов», а также научное обоснование механизмов безопасного поведения в информационном пространстве — одна из главных задач нашей конференции.

Надеюсь, что два дня работы конференции будут плодотворными, дадут новый импульс научным исследованиям, а через год мы вновь встретимся в стенах Московского университета и обсудим полученные результаты.

Академик В. А. Садовничий,
Председатель конференции,
Ректор Московского государственного университета им. М. В. Ломоносова.

Содержание

Общая информация о конференции

Четвертая Общероссийская научная конференция «Математика и безопасность информационных технологий» (МаБИТ-05)

Организаторы — Московский государственный университет им. М. В. Ломоносова, Академия криптографии Российской Федерации.

Семинар-круглый стол «Информационная война и борьба с терроризмом: основные проблемы и международное сотрудничество»

Организаторы — Московский государственный университет им. М. В. Ломоносова, Программа по изучению безопасности университета Кембриджа (Великобритания) при поддержке аппарата Совета Безопасности Российской Федерации и при участии Университета штата Нью-Йорк, Университета им. Г. Гейне (г. Дюссельдорф).

Круглый стол «Комплексная безопасность в отраслях промышленности топливно-энергетического комплекса»

Организаторы — Московский государственный университет им. М. В. Ломоносова, ОАО «Газпром», при поддержке аппарата Совета Безопасности Российской Федерации.

Семинар-совещание руководителей проектов по приоритетному направлению «Безопасность и противодействие терроризму» ФЦНТП

Организаторы — Московский государственный университет им. М. В. Ломоносова, Рабочая группа Научно-координационного совета ФЦНТП.

Сопредседатели конференции:

- В. А. Садовничий — ректор МГУ им. М. В. Ломоносова;
- В. П. Шерстюк — помощник Секретаря Совета Безопасности РФ;
- Н. Н. Андреев — президент Академии криптографии РФ;
- С. К. Ушаков — зам. Председателя Правления ОАО «Газпром».

Руководители семинара-круглого стола «Информационная война и борьба с терроризмом: основные проблемы и международное сотрудничество»:

- Р. Рогозинский — содиректор семинара, Университет Кембриджа;
- В. В. Соколов — содиректор семинара, ИПИБ МГУ;
- А. В. Беляева — координатор семинара, Фонд гражданских инициатив в политике Интернет (Россия);
- Р. Госенде — координатор семинара, Университет штата Нью-Йорк;
- Я. фон Кноп — координатор семинара, Университет им. Г. Гейне (г. Дюссельдорф).

Руководители круглого стола «Комплексная безопасность в отраслях промышленности топливно-энергетического комплекса»:

- В. Н. Пожарский — начальник Управления Службы Безопасности ОАО «Газпром»;
- А. И. Ефимов — начальник Управления Службы Безопасности ОАО «Газпром»;
- Ю. Г. Попов — начальник отдела Управления Службы Безопасности ОАО «Газпром».

Оргкомитет конференции:

- В. В. Ященко — председатель Оргкомитета, зам. директора ИПИБ МГУ.
- В. Н. Сачков — вице-президент Академии криптографии РФ;
- С. Ф. Хомяков — первый заместитель Генерального директора Службы Безопасности ОАО «Газпром»;
- В. Н. Пожарский — начальник Управления Службы Безопасности ОАО «Газпром».

Секретариат конференции:

- Р. А. Шаряпов — отв. секретарь оргкомитета (ИПИБ МГУ);
- В. И. Солодовников (Академия криптографии РФ);
- Ю. В. Малинин (Академия информационных систем);
- М. И. Анохин;
- Г. В. Баранова;
- Т. А. Браташ;
- М. Е. Семина;
- А. В. Соколова.

Часть I
Приветственные выступления

Выступление ректора МГУ академика В. А. Садовничего

Уважаемые коллеги!

Разрешите открыть Международную научную конференцию по проблемам безопасности и противодействия терроризму.

Прежде всего я хотел бы представить членов президиума — основных организаторов конференции:

- помощник Секретаря Совета Безопасности Российской Федерации, директор Института проблем информационной безопасности МГУ Шерстюк Владислав Петрович;
- вице-президент Академии криптографии Российской Федерации Сачков Владимир Николаевич;
- заместитель начальника Службы Безопасности ОАО «Газпром» Хомяков Сергей Фёдорович;
- директор программы по изучению безопасности университета Кембриджа Рогозинский Рафал.

В президиуме также руководители органов государственной власти Российской Федерации:

- руководитель Федерального агентства по информационным технологиям Матюхин Владимир Георгиевич;
- заместитель Председателя Комитета Государственной думы по безопасности Дятленко Валерий Владимирович;
- первый заместитель руководителя Федеральной службы по техническому и экспортному контролю Назаров Борис Викторович.

В зале заседаний конференции присутствуют ученые и специалисты из органов государственной власти, силовых структур, высших учебных заведений и научных организаций, промышленности и бизнеса. На конференцию приехали представители 15 зарубежных стран, в том числе университетов Кембриджа, штата Нью-Йорк, Дюссельдорфа, академии наук Китая, исследовательских центров Канады и Швейцарии, а также большинства стран СНГ. Россия представлена авторитетными учеными и специалистами по проблемам безопасности из более чем 60 высших учебных заведений, исследовательских центров академий наук и промышленности, разработчиками технологий и средств обеспечения безопасности.

Важнейшей особенностью нашей конференции является ее междисциплинарность. Мы будем обсуждать острейшие проблемы человечества — безопасность, противодействие терроризму, формирование глобального информационного пространства. Для решения этих проблем необходимы методы самых различных наук — математики, физики, химии, биологии, психологии, социологии, юриспруденции и так далее, и так далее. Поэтому в программе конференции доклады представителей различных научных школ. Я вижу в зале многих активных участников нашего междисциплинарного межведомственного семинара по научным проблемам информационной безопасности, который работает в МГУ с марта 2001 года. Прошло уже 19 заседаний, в прошлом году мы издали сборник наиболее важных докладов, а к нынешней конференции сборник переиздан и все участники конференции его получили.

Одной из составных частей нашей конференции является является четвертая общероссийская конференция «Математика и безопасность информационных технологий» (МаБИТ-2005), которую мы уже традиционно проводим в МГУ в конце октября. Труды конференции МаБИТ-2004 все участники сегодня получили. В программе нашей конференции около 40 математических докладов по криптографии и компьютерной безопасности.

Уважаемые коллеги! Все мы прекрасно понимаем, что информационное пространство не имеет границ, и поэтому ни одна страна в одиночку не может решить проблемы обеспечения безопасности

глобального информационного пространства. Необходимость международного научного сотрудничества в области информационной безопасности отмечается и в Доктрине информационной безопасности Российской Федерации, утвержденной Президентом России В. В. Путиным в сентябре 2000 г., и в национальной стратегии США по обеспечению безопасности киберпространства, утвержденной Президентом США Дж. Бушем в феврале 2003 г., и в аналогичных концептуальных документах ряда других стран. Ученые Московского университета активно участвуют в различных международных конференциях и проектах, совместно с зарубежными коллегами разрабатывают новые предложения по обеспечению информационной безопасности. В последнее время заработали новые механизмы международного научного сотрудничества — Программа НАТО «Безопасность через науку» и Научный Совет «Россия–НАТО». В этом году Научный Совет «Россия–НАТО» определил пять приоритетов своей работы, один из них — противодействие кибертерроризму. Экспертом от России при Научном Совете «Россия–НАТО» по проблемам кибертерроризма является один из заместителей директора Института проблем информационной безопасности МГУ. На прошлой неделе в Брюсселе по его докладу на заседании Научного Совета были рассмотрены и в принципе поддержаны наши предложения по проектам на 2006 год.

С ноября прошлого года в соответствии с поручением Президента России В. В. Путина реализуется Федеральная целевая научно-техническая программа «Исследования и разработки по приоритетным направлениям развития науки и техники». Одно из приоритетных направлений — «Безопасность и противодействие терроризму». Рабочую экспертную группу по этому направлению возглавляет В. П. Шерстюк. В рамках нашей конференции пройдет Семинар-совещание руководителей 13 проектов, которые уже реализуются в ФЦНТП по этому направлению.

По инициативе руководства ОАО «Газпром» в рамках конференции пройдет круглый стол «Комплексная безопасность в отраслях промышленности топливно-энергетического комплекса».

Мы будем свидетелями очень интересных докладов, дискуссий, работы секций и круглый столов. Я благодарю всех гостей, приехавших в Московский университет на нашу конференцию. Я благодарю всех участников конференции за возможность придти и участвовать в интересных дискуссиях и интересных заседаниях. Позвольте мне пожелать успешной работы нашей конференции и объявить ее открытой.

Выступление руководителя Федерального агентства РФ по информационным технологиям В. Г. Матюхина

Уважаемые коллеги!

Разрешите приветствовать вас от имени Федерального агентства по информационным технологиям России и научного сообщества.

Вопросы информационной безопасности являются многоаспектными и иногда приобретают новые грани, грани нами не совсем осознанные. Как известно, сейчас проводится активная политика по внедрению информационных технологий в систему управления государством, экономикой с целью резкого повышения эффективности управляющих воздействий.

В Федеральном агентстве разработаны принципы реализации информационной инфраструктуры, необходимой для создания «электронного правительства». В основе лежат вопросы, связанные с созданием единого киберпространства и эффективной реализации цифровой подписи, вопросы создания памяти на базе распределения активных хранилищ и создание единой национальной системы идентификации. Все эти три задачи я считаю основными, и как раз изучение этих пространств, на наш взгляд, и создает то информационное взаимодействие, которое необходимо для единства управления, для юридической значимости создаваемых систем управления. Активно ведутся работы по созданию системы удостоверяющих центров. В июне месяце была презентация корневого удостоверяющего центра, разработанного в рамках программы электронной России. И я надеюсь, что научно-технических вопросов в этом случае не осталось, они все были решены, остались вопросы нормативной базы и некоторые юридические вопросы, которые необходимо решить, но для этого уже нужна политическая воля, а не столько интеллект. По вопросам создания распределенных хранилищ сейчас совместно с Московским государственным университетом развернута работа по созданию макета двухкластерной системы с целью определения направления основных работ, подготовки некоторых программных документов. Что касается информационного пространства, то всем хорошо знакомы социальные карты, карты школьника, студента. Сейчас происходит целый бум развития пространства идентификационных элементов. Его надо упорядочить, понять права доступа, исходя из вопросов безопасности, ну и конечно из вопроса создания биометрического паспорта.

Все эти три задачи, которые необходимо решить для реализации электронного правительства в виде правил, регламентов, соответствующей архитектуры, имеют одну слабую часть — это вопрос об информационной безопасности. Необходимы доработки во всех трех направлениях, причем фундаментального характера. Центральными я бы, конечно, выделил вопросы организации информационной безопасности, территориальной распределенности хранилищ информации, вопрос о решении доступа к информации с гарантированной защитой. Системы создаются национального уровня, и уровень защиты естественно должен быть соответствующий.

Я хотел бы пожелать участникам конференции плодотворного обмена мнениями, выработки единых позиций, решения таких глобальных вопросов, как вопросы информационной безопасности.

Выступление первого заместителя руководителя Федеральной службы по техническому и экспортному контролю Б. В. Назарова

Уважаемые участники и гости международной научной конференции!

Разрешите от имени директора Федеральной службы по техническому и экспортному контролю Сергея Ивановича Григорова сердечно поприветствовать вас и пожелать успешной работы.

Сегодня одним из самых быстроразвивающихся секторов мировой экономики являются информационные технологии. Особенностью нынешнего этапа развития информационных технологий является необычайно высокая степень их интеграции во все сферы человеческой деятельности. Наряду с неоспоримыми благами, обусловленными этими процессами, возникает и целый спектр новых угроз, вызовов и рисков национальной безопасности. Эти новые глобальные реальности ведут к тому, что традиционные формы и способы работы сил обеспечения безопасности и правопорядка быстро устаревают. Требуются принципиально новые подходы, иной уровень профессионализма и оснащенности. Нельзя бороться с терроризмом и преступностью XXI века методами и средствами XX векв. При этом требуется консолидация усилий как государственных, так и негосударственных организаций. В этих условиях, очевидно, что обеспечение информационной безопасности является жизненно важным вопросом. Особую значимость в современных условиях приобретает парирование угроз, связанных с террористическими проявлениями. Именно этой теме и посвящена наша с вами конференция. Реалии сегодняшнего дня показывают, что на первый план при решении вопросов информационной безопасности государства выдвигается обеспечение безопасного функционирования информационно-телекоммуникационных систем критически важных объектов. Это обусловлено тем, что несанкционированное воздействие на указанные системы может иметь самые катастрофические последствия, вызвать дезорганизацию управления государством, нанести значительных ущерб, привести к техногенным и экологическим катастрофам, в том числе с большим количество человеческих жертв и крупными материальными потерями. Эти вызовы и угрозы носят глобальный, международный характер. С учетом этого практически все промышленно развитые страны уделяют пристальное внимание вопросам информационной безопасности. Примером этого может служить «Национальный план защиты информационных систем США», нашедший свое продолжение в «Национальной стратегии защиты киберпространства». В Российской Федерации базой для организации и проведения работ в этой сфере является «Концепция национальной безопасности» и «Доктрина национальной безопасности». Одним из органов, который реализует требования этих документов, является Федеральная служба по техническому и экспортному контролю, которую я здесь и представляю.

Следует отметить, что в рамках административной реформы, проводимой под руководством Президента РФ Владимира Владимировича Путина, полномочия и функции службы были расширены с учетом новых угроз в области информационной безопасности. Приоритетами в нашей работе на ближайшую перспективу является формирование нормативно-правовой и нормативно-методической базы в области обеспечения ключевых систем информационной инфраструктуры, разработка типовой модели угроз безопасности, а также частных моделей угроз для конкретных типов категорий информационных систем, разработка на основе моделей угроз требований и норм по защищенности ключевых систем и некоторые другие.

С особым удовлетворением следует отметить международный формат нашей встречи. Базисом для нашего международного взаимодействия по парированию угроз информационной безопасности являются межправительственные соглашения с республиками Казахстан, Белоруссия и Украины о сотрудничестве в области защиты информации.

В завершение выступления позвольте выразить уверенность в том, что результаты конференции дадут новый импульс развитию концептуальных подходов и практических мер по решению проблем безопасности и противодействия терроризму.

Выступление вице-президента Академии криптографии РФ В. Н. Сачкова

Уважаемые коллеги! Уважаемые гости!

Позвольте мне от имени Академии криптографии сердечно приветствовать всех участников конференции, пожелать успешной и плодотворной работы и новых творческих контактов в сотрудничестве по обеспечению информационной безопасности.

В последние десятилетия активное развитие телекоммуникационных информационных систем сыграло важную роль в определении путей дальнейшего развития современной криптографии. Компьютеризация систем связи и управления существенно усложнила и поставила ряд новых проблем информационной защиты. Значительно усложнился характер угроз за счет увеличения возможности доступа к средствам управления, обработки, передачи информации и многообразия средств воздействия с целью изменения содержания и нарушения нормальной работы систем. В связи с этим наряду с традиционными средствами криптографической защиты появилась необходимость создания новых методов обеспечения информационной безопасности. Решение проблем обеспечения целостности и идентификации корреспондентов информации, защиты потоков связи в компьютеризированных телекоммуникационных системах потребовали существенного расширения предмета криптографии и методов их исследования. Существенную роль в исследованиях стала играть разработка методов защиты компьютеризированных информационных систем в условиях агрессивной компьютерной и программной среды и наличия деструктивного воздействия программ вирусов. Революционизирующую роль в развитии современной криптографии сыграло появление асимметричного шифрования, которое существенно дополнило традиционные методы симметричного шифрования. Следует отметить и дальнейшее развитие самого симметричного шифрования за счет значительного увеличения скорости обработки информации, усложнения схем выработки и распределению ключей и т. п. Новые возможности стойкого криптографического шифрования информации созданы благодаря появлению квантовой криптографии. Исследования в этой области проводятся в Российской Академии Наук, в Московском государственном университете и в Академии криптографии РФ. Традиционно большую роль играют в развитии криптографии высокопроизводительные вычислительные системы, которые в значительной степени определяют параметры перспективных криптографических алгоритмов и средств защиты информации.

Важным аспектом развития криптографии как науки является смежная с ней область гуманитарных проблем информационной безопасности. По инициативе ректора МГУ Садовничего Виктора Антоновича в последние годы эта область активно развивается в Московском государственном университете. Созданный под руководством Виктора Антоновича междисциплинарный семинар является важным форумом для обсуждения этих вопросов и сыграл определяющую роль в их обсуждении, в том числе и на настоящей конференции.

Безусловно, актуальнейшую роль играет борьба с международным терроризмом, в частности борьба с кибертерроризмом и информационным терроризмом. Обсуждение этих вопросов с участием коллег из других стран поможет не только отыскать пути борьбы с этим международным злом, но и будет способствовать консолидации ученых и специалистов, работающих в этой области.

В заключение позвольте еще раз приветствовать и криптографов, и всех специалистов в области информационной безопасности, как научно-технического, так и гуманитарного профиля, и гостей. И пожелать всем успехов в работе конференции и круглых столов, и выразить надежду на дальнейшее международное сотрудничество в этой области.

Выступление заместителя председателя комитета Государственной Думы по безопасности В. В. Дятленко

Уважаемые участники и гости конференции!

От имени комитета Государственной думы по безопасности разрешите приветствовать вас по поводу начала работы нашей конференции, одного из наиболее значимых научных форумов, посвященных обсуждению проблемных вопросов — обеспечение безопасности и противодействие терроризму.

Начало XXI века отмечено серьезной активизацией деятельности террористических организаций. В настоящее время, по мнению ряда экспертов, в мире насчитывается около 500 террористических организаций и групп различной экстремистской направленности. Мы столкнулись с вызовами, на которые еще нет симметричного ответа. Терроризм шагнул в область высоких технологий, поэтому особое внимание уделяется разработке проблем обеспечения информационной безопасности. Опасность терроризма заключается еще и в том, что в сфере высоких технологий, организуя свою подрывную деятельность, он провоцирует власть на ответное насилие, что в свою очередь дестабилизирует демократические институты общества, а значит, способствует распространению нарушения прав и свобод гражданина. Одной из основных задач, решаемых в настоящее время законодателями является разработка нормативно-правовой базы, обеспечивающей оптимальное сосуществование силовых и экономических мер борьбы. Мы исходим из принципа, что эффективная борьба с терроризмом во всех ее проявлениях возможна только в том случае, если к ней присоединится гражданское общество. Для этого должна быть выработана единая национальная стратегия противодействия терроризму и обеспечения информационной безопасности. В этой трудной работе мы надеемся на методологическую помощь науки.

Позвольте пожелать вам плодотворной работы и новых творческих решений.

Выступление первого заместителя генерального директора ОАО «Газпром» С. Ф. Хомякова

Уважаемые коллеги!

Разрешите от имени руководства «Газпрома» поздравить участников, гостей и организаторов конференции с ее началом. Конференция научно-практическая, посвященная вопросам безопасности и противодействия террористической угрозе, с нашей точки зрения, крайне актуальна в настоящее время. Собственно поэтому «Газпром» и выступил с инициативой провести специальную секцию, которая позволила бы обсуждать животрепещущий для нас вопрос. На двух аспектах хотелось бы остановиться. «Газпром» — это огромная организация, которая раскинула свои предприятия от Ледовитого океана до наших западных границ. И вопросы информационной безопасности становятся у нас в самую реальную практическую плоскость. Информационные потоки управления всем этим огромным хозяйством — это существенный объект нашей защиты, поэтому мы надеемся, что научные подходы, которые будут высказаны здесь, помогут нам в дальнейшем правильно работать в этом направлении.

Следующим аспектом, на котором бы хотелось специально остановиться, — это вопросы безопасности вообще и вопросы противодействия терроризму. Мне здесь приходится выступать с практической, так сказать, позиции, несмотря на то, что это научная конференция, я вынужден здесь говорить об экономических показателях. С Нового года акции «Газпрома» будут котироваться на зарубежных рынках, и западные инвесторы, возможно, будут их приобретать. Для них немаловажно, насколько мы можем управлять нашими рисками, а один из существенных рисков это терроризм. Я бы мог привести вам здесь печальную статистику, к несчастью, возрастания террористической активности на наших трубопроводах, на наших газоперекачивающих станциях, на наших других объектах. Это действительно есть. В этой связи мы бы надеялись на научные выводы этой конференции, которые бы позволили нам в дальнейшем предсказывать и уменьшать возможности террористических рисков.

Хотелось бы еще раз поздравить участников конференции с ее открытием и надеяться на дальнейшее плодотворное сотрудничество.

Часть II
Пленарные доклады

Актуальные проблемы обеспечения безопасности глобальной информационной инфраструктуры

В. П. Шерстюк, А. А. Стрельцов

Уважаемые участники конференции!
Уважаемые гости!
Дорогие дамы и господа!

Разрешите, прежде всего, выразить глубокую благодарность и признательность всем вам за то, что вы откликнулись на наше приглашение и сегодня собрались в этом замечательном здании, которое в соответствии с величием решаемых здесь задач совершенно справедливо называется Интеллектуальным центром.

Противодействие терроризму и обеспечение безопасности сегодня стали одним из важнейших нормативов российской и международной политики, разработка методов эффективного разрешения этих проблем стало самостоятельным научным исследованием. Для ее решения требуются усилия отечественных и зарубежных ученых разных специальностей, как гуманитариев, так и специалистов в области научно-естественных исследований. Трудно переоценить активное участие в этой работе классических университетов, других научных и учебных заведений. Они способны объединить усилия ученых разных специальностей и выйти на новое понимание сущности терроризма как явления нашей жизни, предложить методы повышения эффективности противодействия терроризму. Определенный опыт организации такового сотрудничества в области безопасности информационной инфраструктуры накоплен Московским государственным университета им. М. В. Ломоносова. Под руководством ректора университета академика В. А. Садовничего университет объединил под своим руководством ученых, заинтересованные федеральные органы исполнительной власти, наладил сотрудничество с консорциумом институтов и академий стран НАТО, занимающихся изучением вопросов безопасности, налажены творческие контакты с университетом штата Нью-Йорк. На прошедшей недели подписано соглашение о сотрудничестве с университетом им. Генриха Гейне (г. Дюссельдорф). С этой точки зрения представляется полезной дискуссия, которая состоялась в рамках конференции «Безопасность и частная жизнь в информационном обществе», прошедшая на прошлой неделе в Дюссельдорфе. Многие участники той конференции присутствуют сегодня в этом зале. Я рад их приветствовать, я рад приветствовать доктора фон Кнопа, одного из организаторов этой конференции. Надеюсь, что мы найдем возможность продолжить дискуссию по вопросам противодействия терроризму. Настоящий доклад имеет более узкую цель — обсуждение приоритетных направлений международного сотрудничества в области обеспечения безопасности глобальной информационной инфраструктуры. Данная проблема давно находится в центре внимания политического руководства России. Это нашло отражение в Доктрине информационной безопасности РФ, утвержденной президентов России В. В. Путиным в сентябре 2000 года. В этом документе обеспечение безопасности функционирования информационных инфраструктур отнесено к составляющим национальных интересов в информационной сфере. Поиск путей решения проблемы приобретает особую актуальность в связи с участием России в формировании глобального общества, предусмотренного Окинавской хартией 2000 года. Это понятно, глобальная инфраструктура становится важным фактором развития общества. В экономической сфере она способствует развитию нового сектора экономики, связанного с компьютеризацией знаний, созданием современных информационных технологий, с формированием индустрии информационных продуктов и устройств, расширением области применения электронной торговли, а также для создания условий для структурной перестройки рынка труда. По имеющимся данным за последние годы внутренние текущие затраты на исследования и разработки увеличиваются на 20–25% ежегодно. Объем услуг связи увеличивается ежегодно более чем на 30% и составил в 2004 году 3% от ВВП. Увеличился объем экспорта российских технологий в зарубежные страны, который составил в 2003 году 23 млрд. рублей,

а в 2004 году уже более 30 млрд. рублей. По некоторым оценкам объем рынка информационных технологий достигнет 600 млрд. рублей. В духовной сфере информационная инфраструктура способствует более полной реализации прав и свобод человека в области информационной деятельности, включая создание и распространение массовой информации, в области свободы выражения мнений, вероисповедования, мысли и слова. По некоторым оценкам к 2010 году количество активных пользователей сети Интернет в России будет насчитывать более 26 мл. человек. В этой системе уже представлены почти все федеральные органы государственной власти. В России интенсивно внедряются безбумажные технологии реализации функции государственного управления, базирующиеся на использовании глобальной информационной инфраструктуры и средств цифровой подписи. В социальной сфере информационная инфраструктура оказывает активное воздействие на развитие системы воспитания и образования граждан и социальное обеспечение. В политической сфере она представляет субъектов политической жизни и качественно новые условия для развития общественных коммуникаций, для ведения политической борьбы, для реализации властных полномочий, определенных функций государства, включая обеспечение его безопасности и обороноспособности. Таким образом, объективно усиливается роль глобальной информационной инфраструктуры в реализации интересов сограждан в обществе и государстве. С другой стороны, возрастают факторы, создающие угрозу безопасности информационной инфраструктуры. Среди основных угроз следует выделить терроризм.

Терроризм все более становится обыденным явлением в нашей жизни. Война, которую развернул терроризм, — это особая война, которая ведется негосударственными образованиями против народа, общества и государства для достижения определенных политических целей. В этой войне противник располагает значительными финансовыми и людскими ресурсами, активно использует для реализации террористических актов свободу информационной деятельности, гарантированную гражданам демократических государств, может организовать акции как с территории своей страны, так и с территории других стран. Объектами террористических атак могут стать информационные и телекоммуникационные системы критически важных объектов: энергетических, транспортных, финансовых и других инфраструктур в обществе, гидро- и атомных электростанций и других экологически опасных промышленных предприятий и т. д. Именно рассмотрению этой проблемы и посвящен круглый стол нашей конференции на тему «Комплексная безопасность в отраслях промышленности топливно-энергетического комплекса».

Существенную угрозу безопасности глобальной информационной инфраструктуры представляет компьютерная преступность, которая объективно создает условия для распространения террористической активности в информационную сферу. Современные информационные технологии представляют значительные возможности доступа отдельным лицам и преступным организациям к информационным и телекоммуникационным системам, к хранящимся и обрабатывающимся в этих системах информационным ресурсам и совершения на этой основе противоправных действий. В результате могут пострадать интересы личности, общества и государства. Так, по мнению специалистов, общий ущерб, причиненный мировой экономике компьютерными преступлениями в 2004 году увеличился почти вдвое по сравнению с 2003 годом и составил более 400 млрд. долларов США. Учитывая высокую латентность компьютерных преступлений, есть основания полагать, что реальный ущерб составляет значительно большую сумму.

Противодействие угрозам безопасности национальных информационных инфраструктур, являющихся сегментами глобальных информационных инфраструктур, является предметом заботы прежде всего органов государственной власти. В то же время некоторые аспекты этой проблемы уже сейчас находятся в центре внимания международного сообщества. Это подтвердило и обсуждение российского проекта резолюции по международной информационной безопасности на первом комитете Генеральной Ассамблеи ООН, которая завершилась на прошлой неделе. Этот проект предусматривает, в частности, продолжение исследования угроз в сфере информационной безопасности и возможных совместных мер международного сообщества по их нейтрализации. В его поддержку высказалось 163 страны.

Плодотворная дискуссия между учеными Российской Академии Наук и национальными академиями США по вопросу противодействия угрозе компьютерному терроризму положила начало процессу укрепления научного сотрудничества двух стран. Террористические акты, произошедшие в последнее время, показывают необходимость дальнейшего расширения этого сотрудничества. Терроризм как явление современного мира еще недостаточно хорошо изучен, но можно выделить общие черты порождаемых им угроз. Во-первых, террористические акты могут совершаться как гражданами атакуемого государства, так и гражданами других государств. Во-вторых, действия террористов, время

и место совершения ими террористических актов слабо предсказуемо. В-третьих, в отличие от других преступных деяний, террористические акты совершаются, как правило, для достижения определенных политических целей. В этих условиях построение системы противодействия терроризму вообще и компьютерному терроризму в частности требует значительных усилий как со стороны отдельных государств, так и международного сообщества в целом. По мере развития глобальной информационной инфраструктуры становится более ясным, что успех каждой страны по защите национальных секторов этой инфраструктуры в определенной степени будет зависеть от успехов в этой области других стран. Жизнь международного сообщества в целом и каждой страны в отдельности будет более защищенной от угрозы компьютерного терроризма тогда, когда будут найдены эффективные формы взаимодействия различных государств, разработаны и внедрены методы противодействия угрозам, средства выявления, пресечения и ликвидации последствий этих угроз. Можно выделить несколько основных направлений противодействия угрозе компьютерного терроризма.

Организационное направление связано с разработкой и реализацией органами исполнительной власти и гражданами системы специальных мероприятий. Они должны быть направлены на затруднение реализации актов компьютерного терроризма и повышение эффективности следственных действий по фактам подготовки и осуществления таких акций, а также на минимизацию связанных с этими актами негативных последствий. Одним из подобных важных мероприятий является совершенствование системы требований к информационной безопасности информационных и телекоммуникационных систем критически важных объектов инфраструктуры общества и их сертификации. Другим не менее важным мероприятием могла бы явиться разработка аналогичных требований к продуктам современных информационных технологий и создание системы добровольной сертификации этих продуктов. Это способствовало бы повышению уровня доверия между разработчиками и пользователями этих технологий. Не менее важной представляется проблема развития системы аудита государственных и негосударственных информационных и телекоммуникационных систем и критически важных объектов инфраструктуры.

Правовое направление противодействия угрозе компьютерного терроризма заключается в разработке и реализации правовых механизмов регулирования общественных отношений, связанных с проявлением угрозы. Эти механизмы должны, с одной стороны, способствовать своевременному выявлению и нейтрализации угрозы, снижения социальной опасности последствий ее проявления, а с другой, не уменьшать государственных гарантий прав и свобод человека и гражданина. В рамках данного направления должно осуществляться, прежде всего, совершенствование национального законодательства в области использования современных информационных технологий в различных сферах жизнедеятельности общества и функционирования государства, развития информационной инфраструктуры и деятельности органов исполнительной власти по противодействию угрозам компьютерного терроризма, взаимодействию органов государственной власти и негосударственных организаций в процессе осуществления этой деятельности. Наряду с этим представляется важным достижение межгосударственных договоренностей по вопросам выявления подготовки трансграничных террористических акций в отношении критически важных объектов национальных информационных инфраструктур и проведение следственных мероприятий по фактам осуществления таких акций.

Техническое направление противодействия угрозе компьютерного терроризма связно с дальнейшим развитием системного использованием средств защиты информации граждан, коммерческих и некоммерческих организаций, органов госвласти.

Кадровое направление противодействия компьютерному терроризму связано с развитием систем подготовки квалифицированных специалистов по правовому, организационному и техническому аспекту этой деятельности. В Российской Федерации этому направлению деятельности уделяется особое внимание. Сегодня более 100 высших учебных заведений осуществляют образовательную деятельность в этой области, обеспечивая подготовку около 2500 специалистов ежегодно. В то же время потребность государственных и негосударственных организаций в подготовке таких специалистов, по оценкам Министерства образования, оценивается примерно в 5000 человек. Важным резервом повышения эффективности совместной деятельности по противодействию компьютерному терроризму явилось бы расширение международного сотрудничества в этой области, создание стандартов образования по некоторым специальностям.

Развитие международного сотрудничества в области противодействия компьютерному терроризму вряд ли будет простой задачей. Определенные проблемы создает различие политических установок, социального и экономического положения взаимодействующих государств, а также отсутствие единой терминологической базы. Тем не менее, представляется, что разумная альтернатива международному

сотрудничеству отсутствует.

В заключение разрешите еще раз поблагодарить вас за участие в работе конференции и выразить надежду, что предстоящее обсуждение позволит не только углубить наше взаимопонимание в области безопасности, но и выйти на конкретные договоренности по совместному движению в направлении укрепления безопасности наших стран, Европы и всего мира.

Международное сотрудничество в области информационной безопасности

А. С. Кремер

Уважаемые члены президиума!
Уважаемые участники конференции!

Одним из примеров международного сотрудничества в области обеспечения информационной безопасности инфокоммуникационных сетей и систем, о котором я хочу вам доложить, является новый проект исследовательской комиссии по информационной безопасности Международного союза электросвязи. Проект называется «Базовый уровень информационной безопасности сетевых операторов». Решение об открытии проекта было принято по инициативе администрации связи РФ 8 апреля 2005 года. Проект имеет категорию «высший приоритет» и должен быть завершен в 2008 году разработкой серии рекомендаций Международного союза электросвязи.

Международная стандартизация является важным фактором представления интересов регуляторов, операторов и производителей оборудования. Поэтому неудивительно, что к работе над проектом наряду с российскими представителями уже подключились представители США, Канады, Японии, Китая, Кореи и Бразилии. Разрешите познакомить вас с 5 основными задачами проекта, а также с планами работы на текущий и следующий год.

Первая задача. Базовый уровень означает использование однотипных критериев операторами взаимодействующих сетей. Использование критериев должно зависеть от типа сети, например, проводная или беспроводная, и от принятого режима регулирования, например, обеспечение требуемого уровня безопасности является одним из условий получения лицензии или условием подключением к сети другого оператора.

Вторая задача. Разработанная в рамках выполнения проекта серия рекомендаций должна стать отражением баланса между предполагаемыми затратами оператора, ожидаемым результатом и возможностью его оценки, баланса национального права и установившейся сетевой практикой саморегулирования, баланса интересов пользователей, операторов и регуляторов. Серия рекомендаций должна содействовать созданию доказательной базы и эффективному правоприменению.

Третья задача. Необходимо организовать взаимодействие экспертов Международного союза электросвязи с экспертами других международных организаций, работающих в области стандартизации ISA, ETC, ATF и другими. Стандарт является реально действующим лишь в том случае, когда он имеет приложение. Для обеспечения государственных структур и бизнеса практическими рекомендациями по оснащению техническими средствами существующие стандарты в рамках выполнения проекта будут проанализированы с точки зрения эффективности их использования, текущего статуса и предполагаемых направлений модернизации.

Четвертая задача. Необходима гармонизация различных языков, на которых специалисты говорят об обеспечении информационной безопасности. Это языки юристов, страховщиков, оценщиков или эвалюаторов, технологов, правоприменительных органов, стандартизаторов.

Пятая задача. Основным критерием обеспечения информационной безопасности должно являться решение защищаемым объектом своих основных функциональных задач в условиях воздействия нарушителей.

В настоящее время можно говорить об эволюционной смене концептуальных моделей постановки решений задач в электросвязи. Основными признаками смены парадигмы связи являются отделение в сетевой архитектуре уровня формирования логики и содержания услуг от уровня передачи информации, постепенное превращение голосовых услуг из основного продукта коммуникаций в одно из многих приложений работающих поверх IP, постепенное превращение услуг связи из непосредственно потребляемого продукта в средства доступа к нетелекоммуникационным сервисам, переход от обеспечивающей телекоммуникационной отрасли к системнообразующей инфокоммуникационной. Смена

парадигмы связи должна находить свое отражение в смене парадигмы системы обеспечения информационной безопасности как неотъемлемой составной части сетей связи. Это выражается в переходе от защиты информации к обеспечению сетевой информационной безопасности, в переводе информационной безопасности из проблемы технической в проблему общества в целом. Подтверждением необходимости такого перевода является развитие сети Интернет, в которой каждый пользователь в определенной степени становится оператором.

Теперь несколько слов об организации работы. Для выполнения проекта в рамках исследовательской комиссии и Международного союза электросвязи по информационной безопасности в октябре текущего года образована так называемая фокус-группа. Такая организационная форма в соответствие с регламентом Международного союза электросвязи открывает возможность участия в проекте организациям, не являющимся членами МСЭЛ. В ближайшие планы работ по проекту входят: подготовка отчета о существующем заделе в исследуемой области; разработка предложений по составу базового уровня безопасности для различных типов операторов связи, проведение семинаров фокус-группы для консолидации подтверждения целей с участием заинтересованных сторон (семинар состоится в марте следующего года в Москве); разработка, согласование и распространение от имени сектора стандартизации Международного союза электросвязи анкет для сбора информации; подготовка обзора по результатам опроса; анализ стандартов, относящихся к безопасности базового уровня, возможных пробелов в идентифицированных стандартах; описание бизнес-приложений, для которых могут быть применимы требования к базовому уровню безопасности; анализ используемых методов оценки информационной сетевой безопасности; и, наконец, формирование предварительного варианта базового уровня безопасности и подготовка предложений для инициирования выпуска рекомендаций сектора стандартизации МСЭЛ.

Unconventional information warfare: challenge determination

R. Rohozinski

Rector Sadovnichij,
Fellows' scholars,
Ladies and Gentlemen,

It is indeed a rare pleasure for me to be here at Lomonosov University presenting on this topic. Indeed our two universities, Cambridge and Lomonosov, have had the rare opportunity of breeding the generations of leaders in both government and security sector, on whose shoulders it has fallen to deal with issues of both national security and national development.

Before I start, a few words about the Cambridge Program. In the last 5 years as the world has increasingly picked up its tempo of globalization new security actors has forced countries east, west, north, south to address new forms of threats. It was recognized within the security services and other responsible agencies on the west that the competencies that have served them well during the Cold War in terms of assessing the nature of the threat and reacting accordingly were no longer appropriate to actors, which were neither state based nor organizations at all, and yet whose capabilities approached those of nation states in the degree of fear and the destruction that they can cause. As a result of the need to help to reconceptualize the nature of security in this new era, the University of Cambridge together with the government and its other partners constituted a new program, which will help "think out of the box" about this issues and help guide those in responsible organs in formulating appropriate policy responses. In the last four years the Cambridge Security Program has convened cross agency meetings which included the Departments of Defense, Internal Security as far as Intelligence to help officers within these institutions and policy makers understand better the nature of the threat that they were looking at. The task was to help them understand the context in which the threat had emerged rather than helping them with their daily work, which is the operational and they new much better. The Advanced Network Research Group, which is the part of the Cambridge Security Program specifically, looks at the intersection between technical networks and technical systems such as the Internet, global communication networks, financial information networks and process control networks and what might be called human agency. Basically what we are interested in exploring is how individuals motivated in different ways can and do use these networks to achieve their political aims which may be contrary to the interests of security of the state and to the international order. Our Program works internationally. We work UK institutions as well as US and others. And a part of acting as facilitator for discussions we are also involved in active research in number of areas including information operations both offensive and defensive as well as the evolution of boarders and boundaries in cyberspace.

It is truly an honor for me that we have been able to hold a NATO Advanced Research Workshop jointly with a Conference on Information Security. This is basically for three reasons, I would say. First of all, NATO workshops generally have been very closed small events where few experts would get together in order to discuss issues. The fact that we have been able to convince a NATO Science Program, and here I would like to thank Bryan Heet, to widen the scope of the workshops, so to let us to be able to include a much broader range of actors, which include government, academia as well as the other sector. I think, it says something about the importance of all three sectors for looking at this particular topic.

Secondly, I think, we should be impressed about the fact that we have managed to have a topic such as this as part of NATO workshop which in composes both the Russian Federation as well as its partners on the west. Information security is a very very difficult topic. Information security when combined with build find transnational threats is even more difficult. It is difficult because the threat of terrorism gets at the very basis of physical security. It is difficult because the information security gets at the very basis of what is considered to be national prerogative and national security. It is also very difficult because the flip side of information security is information warfare and these issues touch upon what is probably the closest

held and last best kept secret of the state which includes the capacity for strategic signals intelligence as well as offensive information operations. For that again I would like to thank Rector Sadovnichij, professor Sherstuk, Streltsov, Sokolov for their hard work in making this event possible.

So my talk? I chose the title "Unconventional information warfare" very deliberately. Although most people turn their attention to the "National Strategy for the Protection of Critical Information Infrastructure" as being the guiding document for information security. This is not necessarily the case. This document has its antecedents in the previous exercise which was held back as far as 1991. I refer to 2000 Defense Science Board Report, recently made public, which in fact looked at the nature of unconventional threats to the national security of the United States in this case. This report identified two specific threats. First was the use of nuclear weapons by non-state actors in undeclared fashion. Second was the use of cyber warfare effective either destruction or denial of the national information infrastructure also by non-state actors. Both of these substudies effectively framed and were responded through both the "National Strategy of Critical Information Infrastructure Protection" as well as the "National Security Strategy of the United States". What is interesting about this 2000 Document is that it held an equivalent between the destructive and disruptive power, between weapons of mass destruction and cyber weapons, considering these to be of equal importance and an equal danger. So why is this important? This is important because since the 2000 Document there've been a major undertaking in terms of building the capacity to deal with the threat of what was known both as conventional information operation as well as defense information security. This effort is largely occurred quietly and underneath of hearsing and was led by the Department of Defense rather than the Department of Homeland Security. It has looked at establishing both the capacity for defending against catastrophic cyber attack as well as creating a capacity to proactively and preemptively take out the possible causes and sources of these attacks whether these are nation-states or, as it was conceptualized in the 2000 Document, non-state actors. So this capacity for creation of the conventional information security information operation capacity has increased. And if look at the differential results between two scenarios that were round: the first, a series of no warning exercises, run in 1997–1998, known as Eligible Receiver, in which a red team of intruders attempted to take control of a number of critical information systems including telephone exchanges and power lines, and the results of the simulation exercise run in 2005 Solid Horiser, two things are clear. One, the nature of the actor which was hypothesized as being a threat was the same, it didn't change. Despite the emergence of new threats in 2001 still the primarily threat was non-state based hackers looking to a political end, taking down the national information infrastructure. But secondly, the other point, was that the capacity to deter, identify and deal with the consequences of such an attack has exponentially improved between 1997 and 2005.

However, despite the existence of both elaborated defensive and offensive information operations strategy developed capacity, not just stated in the document, still the ability to be able to apply successfully this capacity to the new forms of security actors, which were primarily hypothesized as being motivated non-state actors, is not changed. Moreover, there was also recognition that the form of information warfare used by these actors was somehow different, unconventional. And yet the existing strategy mechanisms for dealing with these actors, for reasons we shall describe in this talk, seemed actually be decrecsent. In other words the increasing capacity seemed be leading to degressive effect. Moreover, some of the defensive measures that were being adopted were having a corrosive effect both in terms of the social benefits of the effectively the networks were tried to be defended as well as coming at large economic cost.

So what are the key questions that this raises? One, if the main concern, the main focus of these strategies are the new security actors and yet we are not successful with it, are we actually preparing for the right threat? This cyberterrorism which was the linchpin of the strategy from the 2000 is the one we have been focused on, are we actually focusing on the right threat from these actors? Secondly and more defiantly to the policy of the US (by the way, the reason I am focusing on the US here is because they have the most actively developed and declared strategies, so in the forum like this we can openly address it, but it doesn't mean that such a capacity doesn't exist within the other states as well), why despite the technological superiority and the ability to impose full spectrum information dominance, how the tactical solutions to deal with unconventional actors seem to be so inadequate and many cases so counter productive? So this presentation covers the following. One, review a little bit of contemporary information operations and new security actors. Basically what I want to focus on are two key assumptions that underpin these strategies. Secondly, I want to talk a little bit on the sort of ground stuff at the evidence that is available verifies or doesn't verify that these assumptions that we fore held are actually correct. Thirdly, I would like to present some conclusions and observations.

To understand how information operations as a doctrine emerged there is actually indebtedness here to this region. Concept of the revolution in military affairs holds at least some of its heritage back to marshal Nikolai Ogarkov, who postulated that at some time information becomes the key determinant of the ability to pursuit conflict, that that force better able to achieve information dominance, while denying it to that to the opponent, will all timely prevail. This concept basically was based upon three factors. One, the ability to achieve "God sight" while denying it to the opponent. This simply means the ability to have full pervasive and accurate picture of the informational space in which battle occurs while denying that ability to others. It also means depriving the enemy the situation of awareness by either distorting or deceiving his understanding or her understanding of the battle space itself. Thirdly, the delivery of highly accurate lethal force precisely when and where it is necessary.

Information operations as a doctrine has emerged within a larger doctrine of what is known as effect spaced warfare. What that means is that no longer is information simply an input to words "the achieving of military ends" but in effect the information dominance in of itself is used to compel or convince an opponent to comply or act in desired fashion. Therefore military force is only one part of a much broader spectrum of potential options, the larger part of it being the ability manipulate and shape the environment through information. An information operation as it is understood in the west, and this is a combination of doctrines not just one, is made up of two components. The first one is what you might call a psycho-social which deals with deception and psychological operations. It includes such things as psychological operations, the ability to instill either fear or other form of actions in the opponent prior to these in military force. I can give two examples of very good of this were just prior to the initiation of the operation "Iraqi Freedom". US Computer Network Operations Taskforce identified and sent personal e-mail messages to every single member at this regime that had an e-mail account effectively telling them that if you resist, go to work or continue to operate you will be killed. A more recent example which happened just last week in the West Bank, the defense force called 9000 number in the Northern Gaza Strip basically private numbers informing that every individual that if you are seen to harm the member of Islamic jihad or Hamas in their pursuing launching rockets against Israel, you house will be destroyed and the panic that it caused was phenomenal. So these are psychological operations.

Secondly, there is a kind of propaganda. Kind of propaganda is usually the public diplomacy, for example the use of public radio stations and others in order to create an alternative ideological model. There is operation security which goes without saying and which is simply securing the way that information is used with your particular whipping. And here is military deception. During the first Gulf War the fake thrust against a costal landing in Kuwait has opposed the White Sweeper Forces in the west is a good example of military deception. That is also civic action programming which basically means winning hearts and minds.

The second component is called material-technical. Material-technical refers to active technical measures which are used to either destroy commandeer or disturb the content of information systems belonging to an opponent. It includes such things as broad category computer network operations which is now become a core competency within the US strategic command as well as diversify throughout branches beyond forces. It includes computer network attack using logical and algorithm based attacks in order to take over or deny the use of information systems to an enemy. Computer network exploitation is basically hacking but the idea in the difference of the computer network attack is that the computer network exploitation is reconnaissance activity as well as an activity that is designed to create an active intelligence gathering capacity on networks that are targeted. Electronic warfare which includes the physical destruction of radio and electronic means. And of course the Holy Grail — signals intelligence, which has now evolved, so now it is no longer a question of gathering information in motion, which is traditional for the role of signals intelligence, but now also extends to attacking information at rest. The capacities are being developed in both offensive and defensive and have actually been institutionalized within structures and it also includes overt means. Some of these are technical, some of these are interesting enough and are also based of application of tactical segment but specifically on targeting networks

The dilemma with this rather complex organizational and institutional structure is that both effects based operations and information operations are underpins of the assumptions which are grounded in, on one hand a conventional threat, i.e., something that can be attacked because it is an institution or an organization, or secondly based upon the imagined threat. Another words giving the same characteristics as previously existed to a state to an actor which is a non-state. So what are these assumptions?

First assumption is that the threat that exists is from asymmetric actors willing to use weapons of mass destruction. This is a very important here. The idea is that it assumes that these actors are predisposed to go to creating the largest possible effect which in the roams of imagination within the Defensive Department

is the use of weapons of mass disruption, which is what the information weapons are done for. The way the new security actors have hypothesized are basically any actor that is able to leverage an increasingly globalized technology depended world in three particular ways. One is that they have agency that simply needs to have a group which is motivated. Secondly that that agency's willingness to act is multiplied and amplified through the use of technology. And thirdly, that technology is applied to the largest and most symbolically important mass effect.

Second assumption is that the structure, that I showed you before, Information Operations Doctrine assumes that this non-state actor new security actor can be disrupted, effected or shaped through the application of Information Operations Doctrine as it exists. Problems are: does the evidence actually vary these assumptions out because they are so core to the task itself? Well, let's take a look at it.

First of all let's look at cyber terrorism. First of all the most important thing to know is that there are no recorded instances of an extremists group causing significant damage through the use of cyber attacks. Although there have been many cases of the symbolic uses of cyber intrusions — another forms of defacement. There is no documented case where such an actor as this — a new security actor — has been able either to take over or significantly damage. I think it is an important thing to know because this is not a new question and seven years on it is important to see that it hasn't happened. So how have they used them? A lot of existing theory that has been written in independent studies has been drawn some clear examples. First of which was the Zapatista's use of networks in being able of both bring attention to their struggle and also the use of a very simple denial of service attack as a way in effect raising specter on nongovernmental politically motivated group using a network attack as a part of its political tactic. Problem with this Zapatistas, who have been eloquently written about by Arcylla and Rundfeld and whose case study has underpinned much of the thinking about how non-state actors use information systems, is that it kind of doesn't hold truth. I mean Zapatistas themselves were largely ignorant of technology nor they actually used the technology in their struggle Chiapas. Rather there were nongovernmental organizations primarily from Europe which networked within themselves and on behalf of Zapatistas in effect successfully were able to raise the level of their cause itself. So it seems if look at the retro spectrum of evidence that it was western nongovernmental organizations that found the Zapatistas and ascribed them networking function rather then the Zapatistas organically themselves were using this technology.

Second example, which is being used as a case study within the US Military, has been the campaign of network attack which was waged between propalestinian and proisraeli hackers during 2000. Now for those of you who don't know this event: in 2000 a group of proisraeli hackers managed to take down the main website of Hizbala, the Sheriat based group in South of Lebanon. As a result to that a group of propalestinian hackers systematically attacked the daedal domain, they managed to bring down a largest ISP for a day, they managed to bring down the websites of prime minister's office, of the Ministry of Foreign Affairs and because of the latency they introduced into the network, said, to have caused a 8% dip within the Israeli stock market on that particular day.

However it is important that first of all the limitations in terms of the attack itself were very small. In all cases the damage that it caused virtually through the denial of service was corrected within 24 hours. More importantly for both Palestinians as actors as well as Israelis leading the state these acts had no symbolic meaning whatsoever. In another words these attacks became the lore of the computer information security community. But their actual effects on the ground, their actual impact both symbolically as well as materially were absolutely negligible.

Most importantly in 2000–2003 the Center for Terrorism in Regular Warfare of the US Navy carried out a very interesting experiment where they brought together a number of former quite senior members of militant groups including ATA, Bask oriented group, the IRA, the PLO as well as the elements from the Chechen opposition and brought them together with a number of computer hackers in order to scenario, how militants would conceptualize the possible use of cyber weapons within campaigns that they would structure. What was interesting about this experiment as limited as it was that it found out there was very little interface or commonality between those two groups. At most measure members of militant groups were looking for symbolic victories. Symbolic victories which were directed not just at their opponents but more importantly to reinforce their position within the community that they served. What they needed was acts of symbolic violence which were understandable. The bottom line was that there was no equivalent to suicide bombing in cyber space. Therefore they saw cyber terrorist attacks as being high cost because they required a lot of planning, a lot expertise, where as potential benefit, which was symbolic act of violence which will reinforce their positions within their communities, was very small. As a result the experiment concluded that at least for conceivable generation of militant groups cyber terror attacks are actually were probably not

as significant or possible as was hypothesized previously. I think this is a very important symbolic finding, which I can say is also being verified through some of the work that presently our Program is undertaking in a number of key locations of the work.

To the second assumption. Can conventional information operation disrupt terrorist organizations? Well, in order to understand it, we have to understand: what is it that extremists groups do in terms of leveraging information networks if it is not cyber terrorism? First of all it is a fact that extremists groups do use the Internet for command and control, for fund raising, for data mining, for messaging, for networking and for recruitment. These kinds of functions are what Rundfeld and Arcylla have broadly called social network. Secondly, it s also clear that the extremists groups use the Internet as way of amplifying their message, exaggerating their importance and instilling fear. The examples of these use are wide scale distribution of videos, DVDs, exstreming websites that feature in effect videotaped attacks as well as more recently beheadings and others. It is very interesting that many of the groups we looked at effectively are no lager as a group than between 5 and 20 people. So this ability to amplify their message in effect the strategic multiplier that they use. That's why this particular use for them is far more important than cyber terror. So this second one involves psychological warfare. So basically if we look at it, the current use of information means by terrorist groups is largely if not most entirely held to psycho-social round. But the problem facing all of us is that in terms of targeting these groups is that this use almost completely indistinguishable from the normal use of the Internet or the media. In other words, there is nothing specific for it to effective in terms of targeting it using existing IO means. Moreover in terms of what these groups are after is an amplification psycho-social phenomenon. There is very little to counter in terms of the application of psychological warfare, deception or any of these other well defined means. Moreover what is also very important if we are talking about effecting those groups is that, for example, based upon groups that we know about right now, 80% of them live in diaspora communities which means there is no geographical center of gravity for many of them.

Secondly the actual cells that they are made up for are extremely difficult to penetrate because they are very closed. 20% of those who are operational and active in Afghanistan were actually made out of people who are related to each other. Further 70% were friends, almost all of whom were related through marriage, which means the ability to penetrate or message effectively within these groups is almost impossible. Moreover, when you look at the difference between cells, there is almost no homogeneity whatsoever, which means effectively that the attempts to profile these particular movements is almost impossible and at the same time their capacity is to spontaneously self form, self mobilise and remain isolated within other cells is almost infinite.

The paradox that is facing us when we look at this particular powder is that cost of the actions that we could think of that would degrade the ability of these actors to act by targeting their psycho-social effects are actually ineffective because, one, they are largely indistinguishable behind the noise of the Internet as it begins with and, secondly, when we target them, when they deliberately say: "You, that group of 20, we are going to target you for your messaging", by virtual the fact that most of them circumvented, we in effect build their incredibility because we have targeted them. So there is a paradoxical relationship here, unfortunately. It is a very interesting case that demonstrates just how inadequate existing IO doctrine has been facing this threat and which can be found in this case study. For those of you who don't now, the IDF (Israel Defense Forces) enjoys probably the most compliment security environment of any armed forces dealing with terrorist threat. They enjoy full tactical dominance over the Palestinians in all forms of arms; they have a full freedom of movement to operate anywhere within the territories and in a given time. There is a complaint international environment that recognizes Israeli right for self defense and the use of extra legal means such is targeted killings, the building of the security barrier and the legal basis for arbitrary arrest and detention. Moreover, for those of you who come from the signals intelligence community, Israel also possess the most extensive fixed signals intelligence infrastructure anywhere in the world, which was the part of the US guarantee that the underwrote the Oslo Accord allowing the Palestinians to take some of the responsibility in areas of occupation. They spend more on signals intelligence than any other country in the world and possessed a very pervasive and technologically sophisticated GIS enabled system for surveillance that allows them to keep track of individuals in real time throughout the territories. In addition their physical control of the territory, the ability to limit how people travel through the permit system allows them to use "compromat" as a way of gaining a lot of human intelligence on networks that they deal with and an entire Palestinians telecommunication infrastructure as well as the Internet is rotated through Israeli infrastructure. That means they have a fixed collection point on any external access. And yet despite the existence of both tactical dominance as well as full spectrum information dominance, they have been unable to prevail over determined actors such as Palestinians Islamic jihad and Hamas who have become an expert

in unconventional information warfare, focusing entirely on effects rather than on conventional military units. Without getting into details because I am already running out of time, the fact is this that both of these groups are highly technologically sophisticated in the way they communicate. For example, Hamas holds mass meetings using ever-changing ARC's streams to which literally Hamas people subscribe. They have completely separated their military structure which is based on atomized cells from political structure which is seen as bringing a great benefit to the community it serves by being uncorrupt and serving social needs. Which means that any time when IDF acts either through a targeting killing or through some other form of collective military reprising against the military army of Hamas the political strength of Hamas as a resistance actually rises. It acts as recruiting mechanism to these groups rather than not.

So some conclusions and observations. First of all I would suggest that if we look at the structures, strategies and needs that contemporary information operations information security at the state level targets, it actually misstargets new security actors. It plays to conventional strengths, which is attacks and defense within the state doctrines but it is not really real in terms of these particular actors themselves. The result is that the enormous investment of both capacity, institutions and strategy, I would suggest, resembles very much of "Majino line", meaning that it was a defense strategy built around the threat that no longer exists. Main problem is that, at least for strategic thinkers in the west, they have conflated two different kind of threats: one is non-state actors, which they don't understand and can't get a purchase on with potential strategic competitors such as China, India and others, who have developed and who are developing actively information operations as a way of asymmetrically gaining an advantage over these state based actors.

So main dangers that we are facing is one: over investment into IO against the unconventional actors such as Israel, which will untimely not bring any kind of strategic return and in effect diminished the degree of security we hold against them. Secondly, a danger is of negative collaterals and these includes things such as trying to create barriers and gateways on the Internet as a way of dealing with these actors, vulcanizing perhaps and splitting them off in international segments, using them or creating them through filtering protected zone, effectively eroding freedoms, and probably most importantly the danger of mewling corporate interests which are designed to address security with security itself, which means all of a sudden we have unaccountable practices being put into place to address a threat, which actually doesn't even exist, which all timely erodes freedoms and benefits that we get from these networks. The reality is that the sophistication of conventional and unconventional actors in this particular field will definitely grow and there are some transparent evident which I will discuss at the round table, which are interesting to see in this area or in others.

However, the problem is there are no simple solutions. Unconventional actors thrive because addressing them requires a degree of international coordination and police work, information "shareness" and willing to bring these actors and the communities that they support back into the political mainstream. For both reason of national sovereignty, which deals with signal intelligence, as well as politics, i.e. saying that we negotiating with terrorism, both of these things are very difficult. And yet at the same time they really run at the core being able successfully to address this.

Проблемы борьбы с компьютерной преступностью

Б. Н. Мирошников

Добрый день, дорогие участники конференции!

Очень приятно приветствовать вас в этом, не побоюсь преувеличения, выдающемся учебном заведении, которое недавно отметило свое 250-летие и является по-прежнему флагманом нашей научной школы. Это учебное заведение, которое на весь мир прославилось своими учениками и их делами, поэтому бесконечно приятно находиться в этом здании и, несмотря на то, что это совершенно новый корпус, все равно на нас на всех приятно действует груз двух с половиной веков сподвиженческой научной деятельности во славу науки. Одна из замечательных традиций этого заведения — это откликаться на нужды сегодняшнего дня, страны, народа, человечества. И одна из проблем, которая сегодня объединяет нас сегодня в этом зале, которая охватывает все больше и больше умов на всей планете, тема обеспечения информационной безопасности в нашем информационном мире, в наш информационный век. Проблема эта чрезвычайно остра и очень приятно, что российская научная школа активнейшим образом участвует в решении этой проблемы. Сегодня в стране уже сложилась школа, которая позволяет достаточно реально противостоять преступлениям, которые мы называем преступления в сфере информационных технологий. Во всем мире, как мы знаем, принято имя киберпреступность. Дело не в том, как мы называем эти явления, а, скорее всего, то, что мы понимаем под ними. И могу с удовольствием сказать, что мы все сходимся в определении этих понятий, хотя детально нам эти определения еще придется уточнять. И в этом смысле мы надеемся на помощь науки как в области математики и высоких технологий, так и в области юриспруденции, которые прекрасно сосуществуют под крышей этого учебного заведения. Мы надеемся на их помощь, потому что даже на сегодняшний день в разных странах разные группы специалистов из разных отраслей по-разному трактуют тему терроризма. Эта проблема, что называется, у всех на языке, она чрезвычайно актуальна и беспокоит нас всех, но, тем не менее, четкого определения ей нет. Поэтому в разных странах, называя одно и то же понятие терроризм, мы имеем в виду все-таки разные вещи. Вот эти разделения по типологии исполнения преступления, по целям и задачам преступления, по последствиям, по их реализации, по, в конце концов, общественному резонансу, наверное, подлежат осмыслению, анализу, и это все приведет нас к точным формулировкам. Почему мы все нуждаемся в точных формулировках, нет необходимости вас уговаривать. Мы, действительно, очень рассчитываем на помощь нашей науки, на помощь ученых-юристов, которые помогли бы привести в порядок этот понятийный аппарат.

Кроме того, сегодня существующая нормативно-правовая база в этой области, безусловно, нуждается в совершенствовании. Хотя я могу с удовольствием, с благодарностью от сотрудников правоохранительных органов, которые занимаются этой деятельностью, сказать слова благодарности тем, кто почти десять лет назад работал над этой темой. Тогда эти преступления еще не имели конкретного воплощения в жизнь, тогда многие преступления и их последствия представлялись умозрительно. Тем не менее, наш законодатель имел достаточно мудрости и сумел спрогнозировать ситуацию, чтобы вооружить правоохранительные органы, систему защиты соответствующей нормативной базой и ввел в Уголовный кодекс 28-ю главу. На мой взгляд, это очень прогрессивное явление по тому времени было, которое предвосхитило развитие событий и вооружило инструментом тех, кому положено этим заниматься. Но годы идут, технологии развиваются, к сожалению, и преступный мир развивается. Мы сегодня видим, как меняется его лицо, и, конечно же, правовая база, как вещь консервативная и инерционная, нуждается в совершенствовании и подтягивании, в актуализации. Мы ждем того, что наука, наша передовая наука, окажет содействие юристам, тем, кто конкретно воплощает эти мысли и пожелания в конкретный правовой акт, помогут нам это сделать. Она может реализоваться с тесном содействии с нами, ибо мы каждый день имеем дело с бедами, проблемами, преступлениями. Это и есть обратная связь с жизнью, которая и должна воплощаться в наших научных трудах, разработках и конкретных серьезных решениях на уровне и правительства, и Государственной Думы, и вообще властей. Это имеет очень большое значение, поэтому как в никакой другой области мы ценим взаи-

модействие с наукой. Киберпреступления, наверное, и отличаются от всех остальных преступлений, что помимо всех свойств, которые отличают их от других составов, они обязаны, просто обречены опираться на научные исследования, постоянно быть связаны с наукой. Сам состав этих преступлений требует постоянной экспертизы.

Имеем ли мы сегодня школу экспертов в области киберпреступлений? К сожалению, должен сказать, что нет. Хотя сегодня уже многие коллективы разработали методики, разработали методологию, подготовили соответствующих специалистов, которые имеют квалификацию и законные права проводить экспертизу в этой области. Без них ни одно разбирательство не проходит. Это очень важное обстоятельство. Причем, если одно из свойств компьютерной преступности — это покрытие в мгновение огромных территорий, в том числе и территорий, разделенных административными границами, в которых существуют разные правовые режимы, то это предполагает одинаковую подготовленность всех участников расследования, одинаковую вооруженность и техникой, и интеллектуальной мощью. Сегодня нам нужен сыщик компьютерный, нам нужен сыщик, который владеет этими технологиями, который вооружен правовой базой и может свои расследования проводить на высоком технологическом уровне, к сожалению, высокого уровня компьютерных преступлений. Вот такая диалектика, которая нас обязывает соответствовать этому. И опять я обращаюсь к науке, потому что подготовить методики, подготовить базу по проведению экспертизы, базу для подготовки оперативного состава, способного соответствовать нынешним требованиям, можно только в союзе с наукой.

Научные проблемы противодействия кибертерроризму

В. А. Васенин

Терроризм — это жестокая реальность наших дней, борьба с проявлением которой в настоящее время объединяет отдельных людей и целые государства. С развитием высоких технологий появляются новые способы и расширяется инструментальная база для реализации террористических актов. Одной из таких возможностей является кибертерроризм.По результатам потенциального ущерба такая разновидность терроризма не уступает ни одной из других высокотехнологичных форм его проявления. Более того, возможность реализации деструктивных воздействий из любой точки земного шара при объективных трудностях выявления источника угрозы и ряд других факторов делают кибертерроризм одной из наиболее опасных угроз для человечества. На сегодняшний день мир, к счастью, не имеет реальных фактов проявления кибертерроризма. Причиной тому является технологическая сложность реализации таких действий, недостаточный пока ещё уровень развития сетевой инфраструктуры. Однако темпы развития высоких технологий и мирового информационно-телекоммуникационного пространства свидетельствуют о том, что к борьбе с этой угрозой нужно готовиться уже сегодня. Первым шагом на этом пути должно стать изучение кибертерроризма как явления, упорядочение пространства объектов, субъектов и среды окружения, в котором оно может проявляться. Необходима формализация предметной области с тем, чтобы можно было эффективно оперировать с отдельными её элементами. Далее в конспективной форме изложены основные положения одного из возможных подходов к изучению кибертерроризма как явления, к разработке форм, методов и инструментальных средств противодействия ему, как развитие идей, представленных в [1].

1 Исходные посылки кибертерроризма, как явления. Национальные интересы в сфере безопасности информационных технологий

Практика показывает, что при изучении любого природного явления, технического объекта или феномена в сфере общественных отношений эффективность и конечный результат во многом определяются исходной системой понятий, степенью их формализации. Целью исследований, результаты которых представлены в настоящей работе, являются:

- формулировка основных положений, которые позволяют систематизировать подходы к изучению компьютерного терроризма;
- разработка концепции построения защиты от кибертеррористической угрозы;
- практическая реализация комплекса моделей, механизмов и инструментальных средств противодействия кибертерроризму.

Отправной точкой изучения любого объекта является его определение, которое призвано, исходя из целевых установок работы, представлений и опыта исследователей, аккумулировать основные атрибуты, характеризующие его проявления во взаимодействии со средой окружения. Терроризм — сложное, многофакторное явление, единого строгого определения которому пока не дано. Его изучение, поиск путей и средств противодействия сегодня ведётся на междисциплинарном уровне, включая не только традиционные физику, математику и информатику, но и психологию, политологию, юриспруденцию и экономику. Соответственно модель изучения явления на каждом из перечисленных направлений будет иметь свою специфику. В этой связи в качестве базового, объединяющего с точки зрения цели настоящего исследования многие характеристики терроризма, будем рассматривать следующее.

Терроризм — проявление крайнего экстремизма в действиях, основанных на разногласиях (национальных, транснациональных) отдельных групп лиц с государственными интересами и институтами (в политике, социальной сфере, на религиозной и криминальной почве) и направленных на создание в обществе атмосферы страха и напряженности, на формирование факторов, прямо или косвенно дестабилизирующих состояние национальной безопасности, с целью выдвижения к властным структурам требований, которые не могут быть удовлетворены в рамках существующего правового поля.

В его основе политическая мотивация на основе разногласий определенных групп лиц с государственными интересами и институтами, которые эти интересы реализуют.

Кибертерроризм — одно из направлений терроризма, которое:

- в качестве объектов деструктивного воздействия для достижения своих целей использует информационно-вычислительные комплексы и сетевые сегменты, поддерживающие системы, критически важные с точки зрения национальной безопасности;
- в качестве предмета воздействия использует средства вычислительной техники и их программное обеспечение.

Таким образом, первичной целью кибертеррористической атаки является критически важный объект (КВО), однако воздействие на него реализуется через компьютерную систему управления этим объектом.

В силу стремления террористов к созданию обстановки страха и напряженности в качестве потенциального объекта деструктивного воздействия рассматриваются КВО.

Под *критически важными* (с точки зрения национальной безопасности) понимается объект, который в случае частичной деградации или полной потери функциональности способен прямо и в течении относительно короткого интервала времени влиять на состояние национальной безопасности тех или иных ее составляющих: управление энергоресурсами (атомными, гидро), транспортными потоками (железнодорожными, авиационными), обороноспособностью, критическими производствами и подобными им.

В основу методологии противодействия кибертерроризму естественно положить подходы и методы традиционной информационной безопасности (ИБ) или, более точно (строго), — безопасности информационных технологий (БИТ).

Глобальной целью традиционной системы информационной безопасности является создание комплекса мер:

- превентивных — на законодательном, административном и операционном уровнях;
- обеспечивающих динамический мониторинг безопасности объектов национальной информационно-телекоммуникационной инфраструктуры (НИТИ) и адекватное угрозам оперативное реагирование.

Однако, следует отметить, что *состояние безопасности каждого объекта определяется его потребностями в защите от потенциальных угроз.* А такие потребности различны для разных объектов от операционных систем (ОС) или традиционных баз данных до сложных, многофункциональных, территориально распределённых структур, управляющих целыми секторами национального хозяйственного комплекса. Трудности достижения глобальной цели БИТ усугубляются ещё и тем обстоятельством, что нормативно-правовое поле, как база для эффективного решения этих задач, находится в стадии формирования. При огромном многообразии объектов защиты от разных угроз рост потребностей в необходимых для этого средствах значительно опережает технологические возможности. Так, возможности динамического мониторинга зависят от математического и алгоритмического обеспечения, технологических и технических средств. Разработка подобного обеспечения и таких средств для отмеченного многообразия объектов защиты и объективных трудностях формализации и унификации этого пространства представляет собой сложноразрешимую проблему. Таким образом, на сегодня можно констатировать, что *общество не обладает комплексом средств для решения главной задачи информационной безопасности в необходимом объеме.*

Отмеченное обстоятельство — одна из причин того, что на этом большом поле БИТ следует выделять отдельные, более узкие (хотя и не менее важные) в предметном и функциональном плане области, формализация и систематизация которых позволит более эффективно решать практические задачи. В качестве такой области можно рассматривать БИТ, поддерживающие критически важные объекты.

С этих позиций имеет прямой смысл в общем комплексе потребностей, задач и интересов выделить национальные интересы, суть которых коротко можно сформулировать в виде следующих положений.

- Защита базовых элементов национальной информационной инфраструктуры, прямо влияющих на состояние национальной безопасности.
- Обеспечение устойчивого функционирования национальной магистральной сетевой среды.
- Перманентное развитие комплексной системы обеспечения безопасности национально значимых сетевых информационно-вычислительных структур.
- Создание и поддержка национальной системы подготовки и переподготовки кадров в области безопасности информационных технологий.

Следует отметить, что последние два положения не столь критичны по времени последействия, однако, также влияют на состояние национальной безопасности.

С учётом изложенного налицо конфликт (столкновение) национальных интересов в сфере безопасности информационных технологий и интересов (целей) кибертерроризма.

Здесь и далее по тексту наряду с *КВО — критически важными (с позиций национальной безопасности) объектами*, будем рассматривать *КВОИ — информационно-телекоммуникационные системы как объекты управления КВО*.

Кибертерроризм можно изучать, исследовать подходы и средства противодействия ему. С этой целью необходимо создать:

- теоретическую (хорошо формализованную) базу этой предметной области, выделив объекты, субъекты, среду окружения, средства взаимодействия и ряд других определяющих эту область параметров;
- рекомендации по созданию национальных и межнациональных структур, способных своевременно и адекватно реагировать на соответствующие угрозы;
- инструментальные средства, обеспечивающие надлежащий уровень безопасности КВОИ.

Как осознание этого факта можно рассматривать открытие темы «Методы и средства противодействия компьютерному терроризму» в рамках Федеральной научно-технической программы: «Исследования и разработки по приоритетным направлениям науки и техники» на 2002–2006 гг.

Подводя итог изложенному выше, в качестве исходных посылок для изучения КТ на основе реализации национальных интересов следует рассматривать решение задач на следующих направлениях.

- Основные положения, идентифицирующие кибертерроризм как социальное явление (объекты, субъекты, среда окружения).
- Взаимоувязанный, систематизированный набор угроз, моделей и сценариев компьютерных атак на критически важные объекты.
- Система мер и мероприятий на законодательном, административном и операционном уровнях реализации информационной безопасности.
- Комплекс программно-технических средств, поддерживающих представительный набор механизмов, моделей и сценариев противодействия кибертеррористической угрозе.

2 Объекты, субъекты и среда окружения. Систематизация, категоризация, требования

Можно рассматривать следующую формальную схему взаимодействия объектов и субъектов в ходе реализации кибертеррористической атаки (см. рис. 1).

Как отмечалось ранее, конечной целью кибертеррористической атаки являются КВО, состояние функциональности которых прямо и быстро во времени влияет на те или иные аспекты национальной безопасности. Системы управления этими объектами (КВОИ), как правило, используют национальную

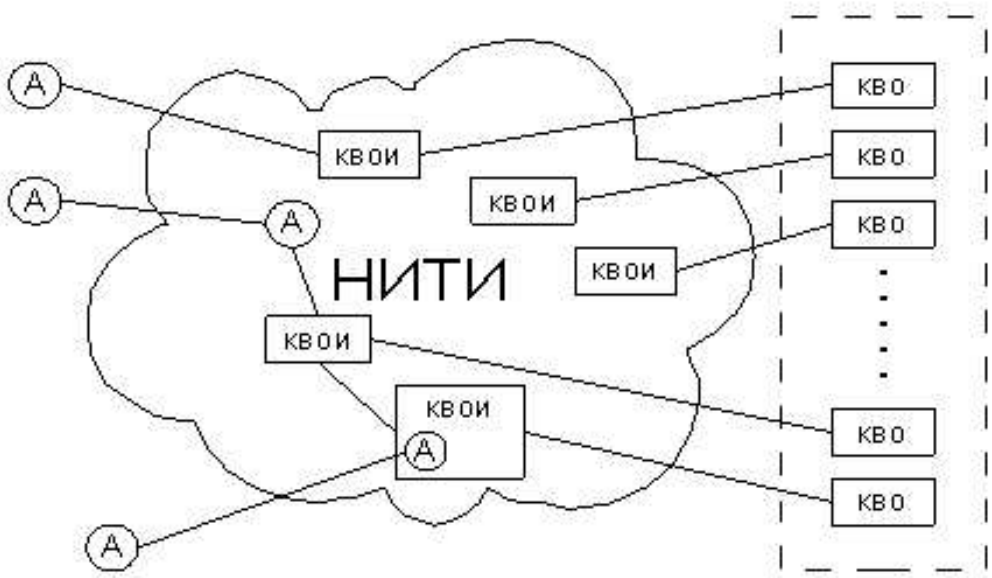


Рис. 1: Общая схема взаимодействия субъектов и объектов в ходе реализации кибератак

информационно-телекоммуникационную инфраструктуру. Агенты, обладающие тем или иным уровнем интеллектуальной поддержки, по отдельности или во взаимодействии друг с другом, используя открытые или организуя скрытые каналы передачи информации, пытаются воздействовать на КВОИ с тем, чтобы добиться деградации КВО или их полной недееспособности. В качестве ключевых для разработки эффективной системы мер и инструментальных средств противодействия кибертеррористической угрозе следует рассматривать следующие направления исследований:

- идентификация критических важных сегментов (КВС) и объектов НИТИ (КВОИ), их кластеризация;
- систематизация кибертеррористических угроз, классификация атак, способов и средств их реализации;
- разработка сценариев и моделей, обеспечивающих динамическое описание взаимодействия КВОИ, их отдельных элементов, процессов анализа их состояния и оперативного реагирования на аномальные ситуации;
- категоризация КВС и КВО с учетом оценки рисков успешных атак и степени их влияния на национальную безопасность.

Атомарным объектом таких исследований, которые в настоящее время активно проводятся во многих странах мира, являются критические сегменты. Их классификация, категоризация с позиций кибертеррористических угроз позволила выстроить структуру управления защитой КВС в высших эшелонах власти. Однако, и это необходимо отметить, такой уровень детализации не позволяет разработать требования, а на их базе комплекс средств защиты конечных объектов. Отмеченное обстоятельство намного уменьшает практическую значимость результатов проводимых исследований.

В постановке задачи, которая рассматривается в настоящей работе, основной акцент делается на выработке подходов к идентификации конечных, атомарных КВО и КВОИ. В качестве идентификаторов на первом этапе исследования рассматриваются две группы, первая из которых объединяет макроидентификаторы, отображающие более общие свойства, а вторая — микроидентификаторы, представляющие локальные свойства объектов.

Макроидентификаторы представляют сферу общественных отношений, в которых функционирует объект — сегмент хозяйственного комплекса, его значимость в территориально-производственной иерархии, архитектурно-топологические свойства.

В свою очередь, *микроидентификаторы* характеризуют типы потоков, защищенность каналов связи, характеристики программного обеспечения объекта.

Важное направление — систематизация угроз, классификация атак. К числу первостепенных задач на этом направлении относятся следующие.

- Разработка модели угроз КВОИ как множество угроз его критическим элементам (на уровне операционной среды, средства коммуникаций, информационные и другие ресурсы прикладного уровня).
- Классификация (таксономия) кибертеррористических атак и способов их реализации.
- Оценки рисков успешной реализации атак на объекты, объединяющие КВОИ и КВО (КВОИ + КВО).

К числу ключевых на этом направлении следует отнести задачу разработки формальных моделей зависимостей между угрозами, атаками и способами их реализации с оценками рисков для отдельных классов КВОИ + КВО.

Другое направление — разработка требований к способам и инструментальным средствам обеспечения безопасности различных классов объектов (КВОИ + КВО), уязвимых с позиции кибертеррористической угрозы. В их числе требования:

- к оценочным уровням доверия к техническим средствам, используемым в составе КВОИ, включая подсистему информационной безопасности;
- к архитектурно-технологическим решениям, механизмам и сервисам безопасности;
- к способам и средствам описания политик безопасности и перманентного контроля за их реализацией, а также ряд других требований.

3 Общие положения концепции построения защиты от кибертеррористического воздействия. Модели, критерии, тестовые испытания

Традиционные подходы к реализации безопасности информационных технологий основываются на следующих способах оценок степени защищённости объектов:

- математические модели, с определенной степенью адекватности описывающие оцениваемые объекты (ОО — объекты оценки);
- критериальные подходы (экспертные оценки защищенности) на всех этапах жизненного цикла ОО;
- на основе тестирования ОО (физического — на полигонах, имитационного — с помощью математических моделей).

Принимая во внимание сложный (в архитектурно-технологическом плане) характер КВОИ, как объекта оценки, уместен вопрос: «*Может ли быть КВОИ объектом оценки*»?. Ответ на этот вопрос утвердительный, однако подобная оценка потребует его декомпозиции на отдельные элементы и применения, как правило, моделей, учитывающих «тонкие» аспекты их взаимодействия в составе большой (исходной) системы.

Математические модели объектов, которые оцениваются с позиций их защищённости в традиционных подходах БИТ, как правило, делятся на две категории:

- модели, описывающие (специфицирующие в виде положений политик безопасности) потенциально уязвимые к атакам свойства ОО;
- модели, поддерживающие проверку соответствия реального ОО (системы, программного обеспечения) математическим моделям.

К математическим моделям первой из упомянутых категорий относятся *модели невлияния* и их развитие. Основоположниками этого направления по праву считаются Гоген и Месгауер [2, 3]. Значительный импульс его развитию дали автоматные модели [4]. В этой связи следует отметить и работы российских исследователей [5, 6].

Перспективными с позиций математического моделирования являются исследования, связанные с поддержкой предикатов безопасности на основе сообщений [7, 8], а также свойства алгебры процессов [9]. Продолжение работ на этом направлении в применении к КВОИ и их элементам позволит получить более детальные сведения об их уязвимости и возможностях противодействия кибератакам, а главное, — даст основания для разработки новых, более эффективных механизмов, моделей и инструментальных средств защиты.

Важное место на этом направлении занимают аналитические модели, логико-языковые средства, описывающие и специфицирующие политики безопасности на основе *дискреционной*, *многоуровневой (мандатной)*, *ролевой* и *смешанных* моделей логического разграничения доступа к ресурсам. Разработка таких моделей и средств в последние годы активно ведётся во многих странах, в том числе, — в комплексе с разработками новых механизмов обеспечения безопасности в ядрах ОС (SELinux, RSBAC, grsecurity и ряд других). Результаты исследований на этом направлении в рамках секционных докладов на конференции будут представлены москвичами К. А. Шапченко, О. О. Андреевым (ИПИБ МГУ) и И. В. Котенко, А. В. Тишковым из Санкт-Петербурга (СПИИ РАН).

Традиционные КВОИ представляют собой совокупность взаимодействующих и, как правило, перманентно изменяющих свойства ОО, многие из которых имеют собственную политику безопасности. В этой связи несомненный исследовательский интерес представляют и имеют хорошие перспективы практического применения модели КВОИ на основе их декомпозиции (многоуровневой) на отдельные компоненты, как ОО и установлении их взаимодействия на базе механизмов отношений доверия. Такие подходы на нашей конференции будут представлены в докладах В. Б. Савкина и А. А. Иткеса (ИПИБ МГУ).

Перечень типов математических моделей первой группы, конечно, не ограничивается описанными выше. Вместе с тем, следует заметить, что их эффективное применение для оценки защищённости практически значимых объектов, как правило, связано с использованием больших вычислительных ресурсов. В этой связи представляют интерес способы и методы, позволяющие эффективно использовать для моделирования большие вычислительные ресурсы.

Математические модели второй категории поддерживают проверку соответствия реального ОО специфицирующим их моделям. Так, например, при создании программных средств защиты объектов с высоким уровнем доверия, как правило, используются программные системы с открытым кодом. В этой связи для них крайне важно создание анализаторов исходных текстов программ на предмет обнаружения уязвимостей (переполнение буферов, утечек памяти и других), которые могут быть причиной различного уровня угроз. Такие модели исследуются и есть результаты в этом направлении в Институте системного программирования РАН [10], Институте проблем информационной безопасности МГУ [11]. Часть из них представлена в материалах прошлогодней конференции МаБИТ-04 [12].

Как отдельный и важный с точки зрения повышения уровня защищённости КВОИ тип математических моделей представляют те из них, которые направлены на верификацию моделей программ. Необходимость в такой верификации возникает не только для программных комплексов в целом, но и отдельных её компонент, вплоть до тех, которые реализуют механизмы безопасности в ядре ОС. Это важное направление. Оно частично будет представлено на настоящей конференции в докладах на секции К. А. Шапченко, И. В. Котенко и А. В. Тишкова.

К числу относительно нового и перспективного следует отнести подход, связанный с математическим моделированием на основе методики создания программного кода с включенным доказательством корректности (proof carrying code) [13].

Результаты решения перечисленных выше задач способны существенно повысить оценочный уровень доверия многих компонентов в составе такого сложного объекта защиты, каким является КВОИ.

Следующее направление связано с оценкой защищенности КВОИ на основе тестирования. Такие испытания могут проводиться как на физических полигонах, так и с использованием имитационных моделей. Первый из подходов более эффективен, однако трудоёмок и очень дорог (с точки зрения необходимых для его реализации ресурсов).

Второе направление менее эффективно с точки зрения точности оценок, требует меньших вложений ресурсов, однако — значительных интеллектуальных затрат на построение адекватных моделей. На этом направлении работы сегодня ведутся и очень интенсивно. Активно используются пакеты

для имитационного моделирования процессов в сетевых сегментах (NS2, OMNET++, INET и другие), которые дорабатываются под новые задачи безопасности информационных технологий. Разрабатываются подходы к созданию подобных программных комплексов и уже есть первые решения у российских исследователей [14]. Такие результаты также будут представлены завтра на пленарном заседании И. В. Котенко и А. В. Улановым (СПИИ РАН), а также в докладах на секции И. В. Батова и М. В. Большакова (ИПИБ МГУ).

Что касается критериальных подходов к оценке защищенности КВОИ на всех этапах жизненного цикла, здесь есть свои сложности. Во-первых, такие объекты уникальны (ОО — не серийный продукт). Вторая сложность связана с их, как правило, территориально распределенным характером и тем обстоятельством, что отдельные элементы ОО могут изменяться в ходе жизненного цикла системы в целом. Третий блок вопросов обусловлен разноплановостью решаемых задач, разнородностью аппаратно-программных средств, используемых в тех или иных компонентах большой системы. Как следствие, необходимы требования, обеспечивающие корректное объединение механизмов безопасности различных компонент КВОИ в единую систему. Отдельные механизмы такого объединения будут изложены в докладах на секции О. О. Андреева, В. Б. Савкина и А. А. Иткеса.

Представляется целесообразной разработка новых, как функциональных, так и требований доверия, профилей защиты для компонентов КВОИ на следующих уровнях:

- операционной среды;
- коммуникационной среды;
- прикладной среды (СУБД, другие прикладные сервисы).

Это очень важное направление исследований. Каких-то значимых практических результатов здесь пока не получено, однако результаты теоретических исследований дают веские основания полагать, что в ближайшее время они будут.

Анализ существующих объектов, которые с полным правом относятся к классам критически важных показывает, что они имеют следующие три основные топологии связности:

- с топологией «звезда»;
- со строгой иерархией связности объектов по территориальному принципу (корпоративные и ведомственные «деревья»);
- комплексы с большим количеством горизонтальных связей (системы межведомственного, межрегионального взаимодействия).

Наибольшие сложности и интерес с точки зрения исследований среди систем с отмеченными топологиями связности представляют КВОИ с большим количеством горизонтальных связей. Причиной тому являются трудности введения единых административных мер и операционных процедур, а также стандартизации на основе единого набора технологических решений и технических средств. Дополнительные вопросы возникают с выделением и классификацией угроз и атак, которым могут подвергаться как отдельные сегменты таких КВОИ, так и каналы связи. В качестве потенциальной уязвимости может использоваться несогласованность политик безопасности при взаимодействии сегментов. К объектам такого типа с полным правом можно отнести системы межведомственного взаимодействия органов государственного управления, формирующиеся в рамках федеральной целевой программы «Электронная Россия». Сложности построения таких систем известны. Однако некоторые подходы можно и нужно использовать уже сегодня. К числу принципов, на которых такие подходы могут строиться, можно отнести следующие:

- внутренняя схема взаимодействия субъектов/объектов каждого сегмента определена и жестко регламентирована;
- внешние взаимодействия строятся на основании процедуры установления отношений доверия.

На основе этих принципов возможно создание «островков» взаимного доверия, как предпосылок для обмена данными о политиках безопасности для их агрегации в единую политику «большой системы».

4 Подходы к реализации комплекса взаимоувязанных моделей, механизмов и инструментальных средств противодействия кибертерроризму

Концептуальные положения обеспечения безопасности критически важных объектов следует рассматривать как базу для построения комплекса взаимодействующих (взаимоувязанных) моделей, механизмов и инструментальных средств противодействия кибертерроризму. В этой связи безусловный интерес представляют основные направления внедрения этих положений в системы противодействия кибертерроризму на всех уровнях реализации комплексного подхода к безопасности информационных технологий. Далее коротко перечислим их без дополнительного обоснования места и роли в таком комплексе мер, принимая во внимание, что такая аргументация была представлена ранее.

С позиций *административного уровня* реализации комплексного подхода к обеспечению безопасности информационных технологий, используемых в составе систем управления критически важными объектами, к числу направлений первостепенного внимания следует отнести следующие.

- Разработка и реализация новых эффективных формальных моделей КВОИ и политик безопасности, инструментальных средств их описания (спецификации) и перманентного, оперативного контроля за выполнением.
- Анализ математического, алгоритмического обеспечения, инструментальных средств перспективных ОС и дистрибутивов на их основе с целью выявления механизмов, эффективных с позиции противодействия кибертеррористическим атакам.
- Разработка новых моделей логического разграничения доступа и их использование на основе уже существующих и вновь создаваемых дистрибутивов ОС.
- Разработка и внедрение в практику новых логико-языковых средств формального и эффективного описания (спецификации) политик безопасности отдельных компонентов КВОИ, обеспечивающих, в том числе, учет их взаимодействия на основе положений общей политики.

Операционный уровень предусматривает применение мер обеспечения безопасности объекта защиты под контролем персонала. К числу практических шагов первой очереди с точки зрения реализации математического обеспечения и инструментальных средств на операционном уровне реализации политики безопасности КВО, уязвимых в плане кибертеррористической угрозы, следует рассматривать следующие.

- Разработка и реализация мер и способов обеспечения безопасности КВОИ с участием персонала, в том числе, с применением инструментальных средств автоматизации типовых бизнес-процессов на этом уровне.
- Разработка заданий на безопасное сопровождение сложных объектов с высоким уровнем доверия, профилей их защиты от кибертеррористических атак.
- Разработка процессов контроля выполнения политик безопасности на отдельных сегментах КВОИ, их взаимодействия на основе положений общей модели и принятия мер оперативного реагирования на аномальные ситуации.
- Создание комплексов, в том числе, распределённых на гетерогенной среде, обеспечивающих непрерывный мониторинг состояния как отдельных элементов (узлов) КВОИ, так и системы в целом на предмет их функциональности, анализ и принятие мер оперативного реагирования на внештатные ситуации.
- Анализ программного кода (исходного и исполняемого) на предмет обнаружения киберуязвимостей и их устранения.

Имея в виду, что программно-технический уровень реализации механизмов и служб, поддерживающих процессы реализации политики безопасности контролируемого объекта без оперативного участия персонала, в качестве основных на ближайшее время для эффективной реализации защиты систем управления КВО от кибертеррористического воздействия следует рассматривать следующие.

- Разработка и практическая реализация механизмов традиционных программно-технических сервисов, учитывающих специфику КВОИ, как систем высокого оценочного уровня доверия, и поддерживающих эшелонированную архитектуру защиты, включая:
  - средства обнаружения и предупреждения вторжений на первом рубеже (анализ системных вызовов, трафика, экранирование и фильтрация, ряд других);
  - эффективные средства идентификации и аутентификации, авторизации и логического разграничения доступа на втором рубеже;
  - средства контроля целостности, активного мониторинга состояния безопасности, анализа и оперативного реагирования на атаки (не обнаруженные ранее) на третьем рубеже.
- Разработка и практическая реализация механизмов традиционных программно-технических сервисов, учитывающих специфику КВОИ, как систем высокого оценочного уровня доверия, и поддерживающих эшелонированную архитектуру защиты, должна проводиться на двух направлениях:
  - построение сервисов на основе уже «де-факто» существующих инструментальных средств;
  - построение сервисов на базе моделей, механизмов и инструментальных средств, поддерживающих новую функциональность и новые уровни доверия.

Заключение

Решение перечисленных выше основных и ряда других смежных задач приведет:

- к унификации понятийной и созданию нормативной базы, регламентирующей деятельность в области противодействия кибертерроризму;
- к разработке формальных, в том числе, математических моделей как кибертеррористической активности, так и защищаемых систем, средств защиты;
- к созданию системы перманентного мониторинга состояния кибертеррористической (или близкой к ней по идентификаторам) активности, анализа уязвимостей потенциальных угроз и выработки адекватных средств защиты.

Литература

[1] Васенин В. А., Галатенко А. В. Компьютерный терроризм и проблемы информационной безопасности в Интернете. В сбор. Высокотехнологичный терроризм. Материалы российско-американского семинара. Москва, 4–6 июня 2001 г. Российская академия наук в сотрудничестве с Национальными академиями США, с. 211–224.

[2] Goguen J. A., Meseguer J. Security Policies and Security Models. Proceeding of the IEEE Symposium on Security and Privacy, Oakland, CA, 1982.

[3] Goguen J. A., Meseguer J. Inference Control and Unwinding. Proceeding of the IEEE Symposium on Security and Privacy, Oakland, CA, 1984.

[4] Moskowitz I. S., Costich O. L., A classical Automata Approach to Noninterference Type Problems, Proceed. Of the Computer Security Foundations Workshop 5, Franconi, NH: IEEE Press., 1992.

[5] Грушо А. А., Тимонина Е. Е. Модель невлияния для сети. Обозрение прикладной и промышленной математики, т. 7 (1), Москва: ТВП, 2000.

[6] Галатенко А. В. Вероятностные модели гарантированно защищённых систем. Математика и безопасность информационных технологий. Материалы конференции в МГУ 23–24 октября 2003 г., М.: МЦНМО, 2004, с. 234–237.

[7] Mantel H. Possibilistic Definitions of Security — An Assembly Kit. Proceeding of the 13th IEEE Computer Security Foundations Workshop, Cambridge, United Kingdom, July 3–5, 2000, p. 185–199.

[8] Mantel H., David S. Controlled Declassifications based on Intransitive Noninterference. Proceedings of the 2th ASIAN Symposium on Programming Languages and Systems (APLAS 2004), Taipei, Taiwan, LNCS 3302, November 4–6, 2004, p. 129–145.

[9] Ryan P., Sneider S. Process Algebra and Non-interference. In IEEE Security Foundation Workshop, 1999, p. 214–227.

[10] Гайсарян С. С., Чернов А. В., Белевенцев А. А., Маликов О. Р., Мельник Д. М., Меньшиков А. В. О некоторых задачах анализа и трансформации программ. Труды Института системного программирования: Том 5. Под ред. В. П. Иванникова. М.: ИСП РАН, 2004, с. 7–41.

[11] Пучков Ф. М., Шапченко К. А. Статический метод анализа программного обеспечения на наличие угроз переполнения буферов. Программирование, 2005, № 4, с. 19–34

[12] Пучков Ф. М., Шапченко К. А. К вопросу о выявлении возможных переполнений буферов посредством статического анализа исходного кода программ. Математика и безопасность информационных технологий. Материалы конференции в МГУ 28–29 октября 2004 г., М: МЦНМО, 2005, с. 347–360.

[13] Appel A. Foundational Proof-Carring Code. In 16th Annual IEEE Symposium on Logic in Computer Science (LICS 01), June, 2001.

[14] Котенко И. В. Многоагентные модели противоборства злоумышленников и системы защиты в сети Интернет. Математика и безопасность информационных технологий. Материалы конференции в МГУ 28–29 октября 2004 г., М.: МЦНМО, 2005, с. 257–266.

Актуальные направления дискретной математики, связанные с приложениями в криптографии

М. М. Глухов, А. М. Зубков

Проблемы обеспечения безопасности информационных технологий практически неисчерпаемы. Важный элемент решения этих проблем — способы защиты информации от посторонних при ее хранении и передаче по каналам связи. Такие способы разрабатываются, изучаются и реализуются в криптографии — науке о процессах преобразования информации с целью исключить возможность неконтролируемого доступа к ней.

Криптография, в свою очередь, широко использует методы различных разделов математики, и прежде всего — дискретной математики, понимаемой в широком смысле. Яркие примеры влияния математических идей дают криптография с открытым ключом (основанная на сложности задач теории чисел) и конструкции современных блочных шифраторов (в которых используются преобразования над сложными алгебраическими структурами, как в AES).

Существует несколько причин, обосновывающих необходимость постоянных и интенсивных исследований стойкости существующих и разрабатываемых методов шифрования.

Действительно, ввиду важности секретной информации надежность криптографических методов защиты информации должна не вызывать сомнений в течение многих лет с момента зашифрования. Однако стойкость почти всех криптографических методов по существу основывается на уверенности в том, что злоумышленники не могут найти секретный ключ или зашифрованную с его помощью информацию. Источником такой уверенности является знание возможностей современных математических методов, вычислительных алгоритмов и технических устройств. Эти возможности со временем возрастают. Поэтому криптограф должен вносить улучшения в метод шифрования задолго до того, как рост этих возможностей станет представлять реальную опасность. Упомянем два примера:

- стандарт шифрования DES перестал считаться стойким в результате роста возможностей ЭВМ,
- стойкость алгоритмов шифрования с открытым ключом Диффи – Хеллмана и Эль Гамаля основывается на сложности некоторых задач теории чисел, однако количество случаев, когда такие задачи относительно просто решаются, постепенно увеличивается (например, в [3], [4] показано, что использование малых показателей в схеме RSA опасно).

Кроме того, криптографические способы защиты информации разрабатываются и изучаются, как правило, как формальные математические преобразования, однако они используются в реальном мире. Практическая реализация способов шифрования отражает современный уровень техники и обладает свойствами, не содержавшимися в исходной теоретической конструкции. Эти незапланированные свойства создают дополнительные возможности для получения информации посторонними лицами. Упомянем два примера такого типа:

- разностный анализ мощности ([5]), использующий данные о потреблении энергии смарт-картами при реализации протокола криптографии с открытым ключом,
- разностный анализ сбоев (см., например, [2],[6]), использующий специфические сбои в процессе вычислений.

Таким образом (ввиду того, что злоумышленники могут обнаружить и использовать уязвимые места систем криптографической защиты информации) необходимо постоянно поддерживать интенсивные и разнообразные исследования в различных областях «чистой» науки (не только непосредственно примыкающих к криптографии).

При разработке и исследовании криптографических устройств применяются различные методы и результаты теории чисел, алгебры, теории сложности алгоритмов, теории вероятностей. С другой

стороны, потребности криптографии (фактически — потребности обладателей секретной информации) являются богатым источником новых математических задач и теорий. Например, вычислительная теория чисел, вычислительная алгебра, теории конечных алгебраических структур (в частности, эллиптических кривых над конечными полями), теория сложности алгоритмов, исследования псевдослучайности, доказательства без передачи знания, теория проверки протоколов и т.п. возникли и развиваются в значительной мере под влиянием конкретных криптографических задач.

Значительная часть математических методов, лежащих в основе криптографии, разбивается на три класса: комбинаторно-алгебраические, теоретико-числовые и вероятностно-статистические. Каждый из этих классов содержит несколько направлений. Некоторые из таких направлений перечислены ниже, и по каждому из них в качестве примеров приводятся отдельные результаты, полученные различными авторами в последние годы и опубликованные в сборнике «Труды по дискретной математике», который издается Российской Академией наук совместно с Академией криптографии Российской Федерации, начиная с 1997 г.

1 Комбинаторно-алгебраические методы

Группы подстановок. Обзор последних результатов был сделан Б. А. Погореловым («Группы подстановок. Часть I. (Обзор за 1981 - 95 гг.)», [8], с. 237 – 281). Главными темами этого обзора являются: теорема О'Нэна – Скотта, максимальные подгруппы, примитивные группы подстановок, унипримитивные подгруппы, кратно-транзитивные группы, действия групп на k-орбитах, разрешимые и нильпотентные группы, операции с группами подстановок.

Несколько работ связано с порождением заданных подгрупп группы подстановок конечного множества различными совокупностями подстановок.

Например, пусть $V_n = \mathrm{GF}(2)^n$ отождествляется с Z_{2^n}, а S_{2^n} — группа подстановок над V_n. Пусть $g = (0, 1, \ldots, 2^n - 1) \in S_{2^n}$ — одноцикловая подстановка, а подстановка $D \in S_{2^n}$ определяется элементами $\alpha_0, \alpha_1 \in V_n$ и функцией $f : V_n \to \{0, 1\}$:

$$Dx = x \oplus \alpha_{f(x)},$$

где $\oplus$ обозначает сложение в $\mathrm{GF}(2)^n$. Пусть $G = \{g^k D, k = 0, 1, \ldots, 2^n - 1\} \subset S_{2^n}$. М. М. Глухов доказал (см. «О числовых параметрах, связанных с заданием конечных групп системами образующих элементов», [7], с. 43 – 66), что если

$$f(0, x_{n-2}, \ldots, x_0) + f(1, x_{n-2}, \ldots, x_0) = 1 \quad \text{для всех} \quad x_{n-2}, \ldots, x_0 \in \mathrm{GF}(2)$$

то существует такое $k \geqslant 5$, что множество G^k является 2-транзитивным.

Далее, пусть S_N — группа подстановок на $\{0, 1, \ldots, N-1\}$, $g = (0, 1, \ldots, N-1)$ — одноцикловая подстановка, а $h = (0, 1) \in S_N$ — транспозиция элементов 0 и 1. Обозначим через D диаметр S_N относительно системы порождающих элементов $\{g, h\}$, т.е. такое минимальное число d, что любую подстановку $c \in S_N$ можно представить в виде произведения не более d сомножителей из $\{g, h\}$. А.Ю.Зубов (см. «О диаметре группы S_N относительно системы образующих, состоящей из полного цикла и транспозиции», [8], с. 112 – 150) получил асимптотически эквивалентные верхние и нижние оценки для диаметра:

$$D \leqslant \left[\frac{N-1}{2}\right]\left(\left[\frac{N}{2}\right] + N - 1\right) + 2N - 1,$$

$$D \geqslant \frac{3N^2}{4} - 2N, \text{ если } N \text{ четно},$$

$$D \geqslant 3\left[\frac{N}{2}\right]^2 - N + 3, \text{ если } N \text{ нечетно}.$$

Ф. М. Малышев (см. «Наследование группой подстановок некоторых свойств семейств образующих», [14], с. 155 – 175) рассматривает группу подстановок G на конечном множестве Z, которое имеет несколько представлений в виде прямого произведения двух своих подмножеств; кроме того, существует система образующих группы G, каждый элемент которой оставляет неизменной одну из

двух координат в каком-то из этих представлений. Указаны условия на совокупности прямых произведений и системы образующих, гарантирующие соответственно транзитивность, примитивность, дважды транзитивность группы G, а также включение в G знакопеременной группы на Z.

Теоретико-групповая классификация функций и автоматов. К этому направлению относятся работы по классификации функций (в частности, булевых функций) относительно различных групп преобразований. Например, А. В. Черемушкин описал методы построения таблиц представителей классов эквивалентности булевых функций от n переменных относительно обобщенной линейной и аффинной групп («Методы аффинной и линейной классификации двоичных функций», [10], с. 273 – 314). Он построил также несколько новых классификаций для случаев $6 \leqslant n \leqslant 8$.

Теперь рассмотрим неавтономный регистр сдвига (НРС) порядка n, т.е. автомат без выхода с входным алфавитом $\mathrm{GF}(2)$, множеством состояний $\mathrm{GF}(2)^n$ и функцией перехода

$$\delta(x,(a_1,\ldots,a_n)) = (a_2,\ldots,a_n,x+f(a_1,\ldots,a_n)),$$

где $f(x_1,\ldots,x_n)$ — булева функция, линейно зависящая от x_1. НРС называют линейным, если $f(x_1,\ldots,x_n) = c_0x_1 + \ldots + c_{n-1}x_n$, при этом многочлен $\chi(x) = c_0 + c_1x + \ldots + c_{n-1}x^{n-1} + x^n$ над $\mathrm{GF}(2)$ называют характеристическим многочленом НРС. В. А. Башев доказал, что НРС линеен тогда и только тогда, когда его группа является расширением элементарной абелевой группы с помощью циклической группы. В терминах групп охарактеризованы также классы линейных НРС с характеристическим многочленом без кратных корней, неприводимым над $\mathrm{GF}(2)$ и примитивным («Теоретико-групповая характеризация неавтономных линейных регистров сдвига», [14], с. 52 – 68).

Исследование и построение отображений с заданными свойствами. К функциям и преобразованиям, используемым в криптографии, предъявляется много различных требований. Поэтому существует большое число работ, посвященных изучению свойств отображений и построению отображений с заданными свойствами. Одно из важнейших криптографических требований к дискретным функциям над полем или кольцом заключается в отсутствии у них некоторых свойств, присущих линейным функциям.

Для сравнения функций с линейными функциями существуют различные подходы (см., например, [1]). Укажем несколько работ из [7] — [14].

Дефицит $d(s)$ подстановки s на конечной группе G порядка n определяется как разность $n - r(s)$, где $r(s)$ — минимальное число трансляций группы G, которыми можно реализовать все переходы подстановки s. Другими словами, $d(s)$ равен числу трансляций G, не имеющих общих переходов с s. В. Н. Сачков (см. «Дефициты подстановок конечных групп», [13], с. 156 – 175) исследовал различные свойства дефицита случайной равновероятной подстановки ς; в частности, он показал, что среднее случайной величины $d(\varsigma)$ зависит только от порядка n группы G, и получил следующие формулы для среднего и дисперсии:

$$\mathbf{E}d(\varsigma) = n\sum_{k=0}^{n}\frac{(-1)^k}{k!},$$

$$\mathbf{D}d(\varsigma) = \frac{n}{e}\left(1-\frac{2}{e}\right) + \frac{n}{n-2}\left(\frac{1}{2e^2} + \theta\frac{2}{n-3}\right), \quad 0 < \theta \leqslant 1.$$

Пусть V — векторное пространство над $\mathrm{GF}(q)$. Преобразование $f : V \to V$ называется k-кусочно-линейным, если k — наименьшее число, для которого существуют такие линейные отображения $L_1,\ldots,L_k : V \to V$, что множество их значений при каждом $x \in V$ содержит значение $f(x)$:

$$f(x) \in \{L_1(x),\ldots,L_k(x)\} \quad \text{для всех} \quad x \in V.$$

Квазипроизводная обратимого преобразования $f : V \to V$ по направлению $a \in V$ определяется равенством $f_a(x) = f^{-1}(f(x+a) - f(x))$. Н. Д. Подуфалов доказал, что множество биекций $f : Z_p \to Z_p$, каждая квазипроизводная которых является k-кусочно-линейной при некотором $k \leqslant 3$, совпадает (за небольшим исключением) с множеством экспоненциальных функций $g(x) = \theta^x, x \in Z_p\backslash\{0\}$, $g(0) = 0$, где θ — первообразный корень по модулю p («О некоторых характеризациях экспоненциальных функций на линейных пространствах», [14], с. 216 – 239).

Величина k для k-кусочно-линейной функции s является аналогом величины $r(s)$ из работы В. Н. Сачкова, поскольку обратимые линейные преобразования пространства $\mathbf{Z}_p$ являются трансляциями группы $(\mathbf{Z}_p)^*$.

Пусть A — конечный алфавит, A^n — множество всех слов длины n над алфавитом A. В 1956 г. А. А. Марков доказал, что каждое биективное отображение $A^n \to A^n$, не размножающее искажений типа замены букв является суперпозицией подстановки на алфавите A и подстановки букв слова. М. М. Глухов обобщил эту теорему на инъективные отображения, не размножающие искажений типа замены букв, пропуска букв и вставки букв «Инъективные отображения слов, не размножающие искажений», [10], с. 17 – 32).

Пусть $(G, *)$ — квазигруппа. Отображение $f : G \to G$ называется сильно биективным, если биективны f и h, где $h(g) = g * f(g)$, $g \in G$. Если $(G, +)$ — абелева группа, то отображение $f : G \to G$ называется вполне сильно биективным, если биективны f и все отображения h_k, где $h_k(g) = kg + f(g)$, $k = 0, 1, \ldots, g \in G$. Эти понятия связаны с построением трансверсалей в квазигруппах. М. В. Федюкин описал класс сильно биективных отображений и нашел критерий вполне сильной биективности для элементарной абелевой p-группы (¡О некоторых классах сильно биективных и вполне сильно биективных отображений», [12], с. 226 – 238).

Линейные рекуррентные последовательности (ЛРП). Это направление имеет большую историю. Особенно много работ появилось во второй половине XX века в связи с использованием ЛРП над конечными полями и кольцами в криптографии. В нашей стране в последние годы больших успехов в изучении ЛРП достигли А. А. Нечаев и его ученики А. С. Кузьмин, В. Л. Куракин и др. Наряду со свойствами ЛРП исследуются также свойства линейных и полилинейных рекуррентных последовательностей над квазифробениусовыми модулями и кольцами Галуа. В частности, рассматриваются:

- условия, обеспечивающие максимальность периода,
- ранги координатных последовательностей,
- распределения элементов на циклах последовательностей,
- представления последовательностей.

Ряд этих результатов опубликован в «Трудах по дискретной математике: *Кузьмин А. С., Куракин В. Л., Нечаев А. А.* «Псевдослучайные и полилинейные последовательности», [7], с. 139 - 202; «Свойства линейных и полилинейных рекуррент над кольцами Галуа (I)», [8], с. 191 – 222; «Структурные, аналитические и статистические свойства линейных и полилинейных рекуррент», [9], с. 155 – 194; «Статистические свойства линейных рекуррент над кольцами Галуа и квазифробениусовыми модулями характеристики 4», [10], с. 91 – 128; «Вполне равномерные линейные рекурренты над кольцами Галуа и QF-модулями характеристики 4», [11], с. 103 – 158; *Нечаев А. А.* «Многомерные регистры сдвига и сложность мультипоследовательностей», [12], с. 150 – 164; «Конечные фробениусовы бимодули в теории линейных кодов», [14], с. 187 – 215; *Куракин В. Л.* «Биномиальная линейная сложность полилинейных последовательностей», [12], с. 82 – 138; «Полилинейные преобразования линейных рекуррентных последовательностей над модулями», [13], с. 89 – 113.

2 Теоретико-числовые методы

Большинство работ теоретико-числового характера связано с анализом и синтезом систем открытого шифрования или открытого распределения ключей. В частности, обсуждаются проблемы факторизации чисел и многочленов, дискретного логарифмирования, исследуются алгебраические структуры, пригодные для схем типа RSA или Эль-Гамаля.

Если a, m — взаимно простые натуральные числа, то их частное Ферма определяется равенством

$$Q(a, m) = (a^{\lambda(m)} - 1)m^{-1} \pmod{m},$$

где $\lambda(m)$ — экспонента группы $(\mathbf{Z}/m\mathbf{Z})^*$. Ю. В. Нестеренко применил частные Ферма к проблеме дискретного логарифмирования («Частные Ферма и p-адические логарифмы», [11], с. 173 – 188). Он построил класс таких троек (g, m, r), что m — период функции $Q(x, r)$, а $x \equiv Q(a, r)/Q(g, r) \pmod{r}$ — решение сравнения $g^x \equiv a \pmod{m}$. Для таких троек показательное сравнение решается сравнительно просто.

М. И. Анохин показал, что если вероятностный алгоритм A решает проблему Диффи – Хеллмана для множества N целых чисел (модулей) с вероятностью $p \geqslant \varepsilon(N)$, то существует вероятностный алгоритм B, который находит некоторые делители чисел из N с вероятностью $k(N)\varepsilon(N)$, $0 < k(N) < 1$. Более того, если алгоритм A полиномиален, то алгоритм B тоже полиномиален («О сводимости задачи факторизации целых чисел к задаче Диффи–Хеллмана», [9], с. 7 – 20).

В работе О.Н. Василенко («Некоторые тождества для тригонометрических сумм Гаусса и их приложения», [14], с. 69 – 78) предлагается использовать для схемы типа RSA кольцо В вычетов по модулю $p^k q$ кольца $\mathbf{Z}_K[1/q]$, где $\mathbf{Z}_K$ - кольцо целых алгебраических чисел кругового поля К, являющегося расширением поля $\mathbf{Q}$ примитивным корнем степени $p^k q$ из 1. С этой целью доказывается тождество для суммы Гаусса, которое можно использовать вместо обычного для схемы RSA тождества $(a^\alpha)^\beta \equiv a \pmod{n}$.

М. М. Глухов предложил использовать для схемы RSA кольцо вычетов по модулю $n = pq$ (p, q — простые числа) биквадратичного расширения поля $\mathbf{Q}$ и изучил строение этого кольца («Исследование колец вычетов биквадратичных расширений кольца целых чисел и схемы с открытым ключом», [13], с. 31 – 55).

В. Е. Тараканов изучал эллиптические кривые вида $y^2 = x^3 + Ax + B$ над полем $\mathbf{Z}_p$ при $p \neq 2, 3$ и $4A^3 + 27B^2 \neq 0$. Для отображения $\psi(x) = x^3 + Ax + B$ найдено число элементов, имеющих k прообразов при $k = 0, 1, 2, 3$, и описаны элементы группы эллиптической кривой порядков 3, 4. Построен также критерий того, что порядок точки делится на 2 («Об области значений кубического многочлена над конечным простым полем», [9], с. 283 – 294; «Свойства делимости точек эллиптических кривых над конечным полем», [10], с. 243 – 258).

3 Вероятностно-статистические методы

Вероятностно-статистические методы широко используются в теоретической криптографии. Разнообразие криптографических способов защиты информации и возможных атак на них порождает широкий спектр направлений вероятностно-статистических исследований.

Системы случайных уравнений. Задачи определения секретного ключа можно сводить к решению тех или иных систем уравнений над конечными алгебраическими структурами. Ввиду случайности исходных данных естественно считать, что уравнения в этих системах случайны.

Г. В. Балакин изучал различные методы решения некоторых классов систем уравнений типа

$$\varphi_i(x_{i,1}, x_{i,2}, \ldots, x_{i,k}) = b_i, \quad i = 1, \ldots, T,$$

относительно неизвестных $x_1, \ldots, x_n$ из конечного поля, где φ_i — известные функции,

$$b_i = \varphi(x^*_{i,1}, \ldots, x^*_{i,k}) + \varepsilon_i, \quad i = 1, \ldots, T,$$

и $\varepsilon_1, \ldots, \varepsilon_T$ — неизвестные независимые ошибки («Введение в теорию случайных систем уравнений», [7], с. 1 – 18; «Системы случайных уравнений над конечным полем», [8], с. 21 – 37; «Системы случайных булевых уравнений со случайным выбором неизвестных в каждом уравнении», [9], с. 21 – 28; «Критерии, выделяющие заведомо совместную систему уравнений с искаженной правой частью», [10], с. 7 – 16; «Последовательный критерий выделения системы линейных уравнений с искаженной правой частью», [11], с. 21 – 28; «Алгоритм нахождения множества наименьшей мощности, содержащего истинное решение с заданной вероятностью», [13], с. 7 – 21; «Об одном критерии выделения системы линейных уравнений с искаженной правой частью», [14], с. 25 – 33; *Балакин Г. В., Бачурин С. А.* «Оценка параметров метода последовательного подбора неизвестных», [12], с. 7 – 13).

В. Ф. Колчин изучал свойства вероятности совместности случайных систем линейных уравнений (в частности, пороговый эффект) и методы решения систем уравнений, возникающих в задачах классификации по парным сравнениям («О пороговом эффекте для систем случайных уравнений», [8], с. 183 – 190; «Вероятность совместности одной системы случайных уравнений специального вида», [9], с. 130 – 146; «Одна задача классификации с использованием парных сравнений», [10], с. 83 – 90).

Вероятностные модели конечных автоматов. Конечные автоматы как модели криптографических устройств обычно сложны для изучения. Чтобы исследовать типичные свойства конечных автоматов, принадлежащих некоторым классам, рассматривают различные вероятностные модели конечных автоматов (как правило, в виде цепей Маркова).

Ю. И. Максимов исследовал некоторые аналитические свойства цепей Маркова, соответствующих двоичным регистрам сдвига, возмущенным случайным шумом: спектры переходных матриц, скорости сходимости к равномерному распределению («О цепях Маркова, связанных с двоичными регистрами сдвига со случайными элементами», [7], с. 203 – 220). Например, показано, что если

$$y_{t+n} = a_{n-1}y_{t+n-1} + \ldots + a_0 y_t + z_t, \quad z_t = z_{t-1} + \xi_t,$$
$$\mathbf{P}\xi_t = 1 = (1+\Delta)/2,\ P\xi_t = 0 = (1-\Delta)/2,\ \Delta > 0, \quad t = 0, 1, \ldots,$$

и p_t — распределение вектора $(y_{t+n-1}, \ldots, y_t, z_t)$, а ω — равномерное распределение на $\mathrm{GF}(2)^{n+1}$, то

$$\|p_t - \omega\|^2 \leqslant \Delta^{[(t-1)/(n+1)]}.$$

В. Н. Сачков (см. «Вероятностные преобразователи и правильные мультиграфы. I», [7], с. 227 – 250; «Цепи Маркова итерационных систем преобразований», [12], с. 165 – 183; «Вероятностные преобразователи и суммы элементарных матриц.II», [14], с. 240 – 252) рассматривал цепи Маркова с конечным множеством состояний S, определенные рекуррентными соотношениями

$$y_{t+1} = f(y_t, x_t),\ t \geqslant 0,$$

где x_t — последовательность независимых одинаково распределенных случайных величин со значениями $\{1, \ldots, k\}$ и для каждого x функция $f(\cdot, x)$ является биекцией S. Условия эргодичности цепи y_t формулируются в комбинаторных терминах.

Ю. Н. Горчинский получил оценки среднеквадратичной скорости сходимости для матриц переходных вероятностей цепей Маркова, порожденных случайными блужданиями на множествах подстановок («Об улучшении оценок средних квадратических уклонений матриц перехода произведений независимых случайных величин на конечных группах подстановок», [9], с. 53 – 72; «О средних квадратических уклонениях матриц перехода на конечных группах подстановок четного порядка», [9], с. 73 – 94).

В. Г. Михайлов рассматривал свойства автоматов, состоящих из регистров сдвига с неравномерным движением («Исследование числа циклических точек автомата из регистров с неравномерным движением», [11], с. 167 – 172; «Исследование комбинаторно-вероятностной модели автоматов из регистров с неравномерным движением», [12], с. 139 – 149). Множество внутренних состояний автомата отождествляется с многомерным дискретным тором Q, а закон движения описывается совокупностью переходов в случайно выбираемые соседние точки. Показано, что математическое ожидание числа точек, лежащих на циклах, больше, чем у случайного отображения, в котором переход из каждой точки совершается равновероятно в любую точку Q.

В. А. Иванов исследовал влияние внешних и внутренних помех на работу конечных неавтономных автоматов из некоторых классов («Автоматные преобразования случайных последовательностей», [8], с. 151 – 168; «О влиянии внешних помех на работу дискретного автомата», [9], с. 95 – 110). Получены формулы для вероятности того, что помехи во входных и управляющих последовательностях приведут к искажению знака на выходе автомата.

М. И. Рожков указал условия, при которых сумма цепей Маркова на конечной группе тоже является цепью Маркова («О суммировании цепей Маркова на конечной группе», [9], с. 195 – 214).

С. Ю. Мельников показал, что множество совместных распределений заданных слов во входной и выходной последовательностях конечного автомата образует многогранник («Многогранники, характеризующие статистические свойства конечных автоматов», [13], с. 126 – 137).

Вероятностно-комбинаторные задачи. Вероятностно-комбинаторные задачи возникают в различных разделах криптографии; часто они интересны и с точки зрения теории вероятностей.

Г. И. Ивченко и Ю. И. Медведев в цикле работ применяли методы теории разделимых статистик к задачам, связанным со случайными размещениями частиц по ячейкам, случайными многочленами, случайными подстановками, урнами с переменным составом («Смеси вероятностных распределений и случайные размещения», [8], с. 169 – 182; «О структуре случайных многочленов над конечными полями», [9], с. 111 – 129; «Экстремальные характеристики случайного многочлена над конечным полем», [10], с. 71 – 82; «О случайных подстановках», [11], с. 73 – 92; «Исследование характеристик урновых схем с переменным составом», [12], с. 64 – 81; «Об одном классе неравновероятных подстановок случайной степени», [13], с. 75 – 88; «Статистика параметрической модели случайных подстановок», [14], с. 116 – 127). .

Б. А. Севастьянов нашел предельные распределения перманентов случайных $m \times n$-матриц с независимыми элементами в поле $\mathrm{GF}(p)$ («Распределение вероятностей перманентов случайных матриц с независимыми элементами в поле $\mathrm{GF}(p)$», [9], с. 235 – 248). Он же (см. «Структурные характеристики некоторых неравномерных отображений конечных множеств», [12], с. 184 – 193) рассматривал двудольные случайные отображения $f : X \to X$ конечного множества $X = X_1 \cup X_2$ в себя, равномерно распределенные на множестве всех таких отображений, что $f(X_1) \subseteq X_2$, $f(X_2) \subseteq X_1$. Показано, что если $|X_2| \to \infty$ и $|X_1|^2/|X_2| \to 0$, то при любом фиксированном $k \leqslant |X_1|$ и любых попарно не равных друг другу $x_1, \ldots, x_k \in X_1$ для любых $y_1, \ldots, y_k \in X_1$

$$\mathbf{P}f(f(x_j)) = y_j, j = 1, \ldots, k = |X_1|^{-k}(1 + O(|X_1|^2/|X_2|).$$

Ряд предельных теорем для распределений на конечных группах доказан в работах Ю. Н. Горчинского, И. А. Круглова, В. М. Капитонова, Ф. К. Алиева (*Горчинский Ю. Н., Круглов И. А., Капитонов В. М.* «Вопросы теории распределений на конечных группах», [7], с. 85 – 112; *Горчинский Ю. Н., Капитонов В. М.* «О средних квадратических уклонениях в строках матриц переходных вероятностей на конечных группах подстановок», [8], с. 88 – 100; *Алиев Ф. К.* «Произведения независимых одинаково распределенных случайных величин со значениями в конечной простой полугруппе», [8], с. 1 – 20). Ю. Н. Горчинский начал исследование отображений конечных групп, которые лишь частично совпадают с их автоморфизмами («О π-автоморфизмах конечных групп», [10], с. 33 – 50).

В. И. Шерстнев описал множество пар распределений на конечной абелевой группе, свертка которых является равномерным распределением, и показал, что оно образует выпуклый многогранник («Разложение равномерного распределения на конечной абелевой группе», [10], с. 315 – 318).

Последовательность независимых испытаний образует идеальную случайную последовательность. Изучение ее свойств необходимо, например, для построения статистических критериев, обнаруживающих отличие свойств наблюдаемой последовательности от свойств идеальной последовательности. В цикле работ В. Г. Михайлова и А. М. Шойтова доказан ряд предельных теорем для распределений числа пар цепочек, совпадающих либо точно, либо с точностью до переобозначений или подстановки элементов (*Михайлов В. Г.* «Неравенства для среднего числа повторений m-цепочек и для среднего числа непоявившихся m-цепочек из заданного класса», [9], с. 147 – 154; «Предельные теоремы пуассоновского типа для числа пар H-связанных цепочек», [13], с. 138 – 155; «Об особенностях асимптотического поведения числа пар структурно близких цепочек», [14], с. 176 – 185; *Шойтов А. М.* «Об одной особенности асимптотического поведения числа наборов H-эквивалентных n-цепочек в неравновероятной полиномиальной схеме», [13], с. 227 – 238; «Предельные распределения случайных величин, характеризующих связь цепочек полиномиальной схемы структурной эквивалентностью», [14], с. 312 – 326).

А. М. Зубков предложил эффективный метод точного вычисления распределений сумм некоторых зависимых случайных величин, основанный на использовании неоднородных цепей Маркова («Методы расчета распределений сумм случайных величин», [11], с. 51 – 60).

Статистические задачи. В цикле работ М. И. Тихомировой и В. П. Чистякова изучались различные модификации критерия Пирсона, предназначенные для проверки гипотезы о структуре случайной последовательности по частотам цепочек, образованных элементами этой последовательности («О статистических критериях отсутствующих s-грамм», [7], с. 265 – 278; «О статистиках χ^2, построенных по выходу конечного автомата», [8], с. 305 – 314; «Статистические критерии, построенные по частотам s-грамм из некоторого множества», [9], с. 295 – 302; «Нормальное приближение многомерного χ^2-распределения», [10], с. 259 – 272; «Об одной характеристике двухэтапной процедуры выбора из нескольких марковских гипотез», [11], с. 241 – 246; «Приближенное вычисление функционалов от предельных распределений некоторых статистик», [12], с. 213 – 225; «Предельные распределения некоторых статистик, связанных с рекуррентными событиями», [13], с. 201 – 212; «Многомерные χ^2-статистики в задачах разладки», [14], с. 281 – 298)..

Асимптотическая эффективность разделимых статистик изучалась Г. И. Ивченко и Ю. И. Медведевым («Об асимптотической эффективности разделимых статистик в полиномиальной схеме», [7], с. 121 – 138) для полиномиальной схемы и С. В. Полиным («Построение наиболее эффективных по Питмену разделимых статистик для различения гипотез о суперпозиции случайных отображений», [11], с. 189 – 204) для различения гипотез о суперпозиции случайных отображений.

В работах А. В. Лапшина построены и исследованы статистические оценки степени неравновероятности слагаемых, принимающих значения в конечной группе, по наблюдениям над суммами таких

слагаемых («Статистическое оценивание распределения слагаемого по серии наблюдений суммы независимых случайных величин на конечной абелевой группе», [10], с. 129 – 148; «Оценка одного параметра распределения случайной величины на конечной абелевой группе по сумме ее реализаций с элементами случайной подстановки», [14], pp. 139 – 147).

Перечисленные выше работы составляют примерно половину статей, опубликованных в [7] — [14], и дают довольно полное представление о характере этих сборников.

Литература

[1] О. А. Логачев, А. А. Сальников, В. В. Ященко. Булевы функции в теории кодирования и в криптографии. — М., МЦНМО, 2004.

[2] D. Boneh, R. A. DeMillo, R. J. Lipton. On the importance of checking cryptographic protocols for fault. — EUROCRYPT'97, Lect. Notes Comp. Sci., 1997, v.1233, pp.37–51.

[3] D. Coppersmith. Small solutions to polynomial equations, and low exponent RSA vulnerabilities. — J. Cryptology, 1997, v.10, № 4, pp.233 – 260.

[4] C. Coupé, P. Nguyen, J. Stern. The effectiveness of lattice attacks against low-exponent RSA. — PKC'99, Lect. Notes Comp. Sci., 1999, v.1560, pp. 204 – 218.

[5] P. Kocher, J. Jaffe, B. Jun. Differential Power Analysis. — CRYPTO'99, Lect. Notes Comp. Sci., 1999, v.1666, pp. 388-397.

[6] D. Wagner. Cryptanalysis of a provably secure CRT-RSA Algorithm. — CCS'04, October 25-29, 2004, Washington, DC, USA.

[7] Труды по дискретной математике (под ред. В. Я. Козлова). Т. 1, Москва, ТВП, 1997.

[8] Труды по дискретной математике (под ред. В. Я. Козлова). Т. 2, Москва, ТВП, 1998.

[9] Труды по дискретной математике (под ред. В. Я. Козлова). Т. 3, Москва, ФИЗМАТЛИТ, 2000.

[10] Труды по дискретной математике (под ред. В. Я. Козлова). Т. 4, Москва, ФИЗМАТЛИТ, 2001.

[11] Труды по дискретной математике (под ред. В. Я. Козлова). Т. 5, Москва, ФИЗМАТЛИТ, 2002.

[12] Труды по дискретной математике (под ред. В. Я. Козлова). Т. 6, Москва, ФИЗМАТЛИТ, 2002.

[13] Труды по дискретной математике (под ред. В. Я. Козлова). Т. 7, Москва, ФИЗМАТЛИТ, 2003.

[14] Труды по дискретной математике (под ред. В. Я. Козлова). Т. 8, Москва, ФИЗМАТЛИТ, 2004.

Программный полигон и эксперименты по исследованию противоборства агентов нападения и защиты в сети Интернет

И. В. Котенко, А. В. Уланов

1 Введение

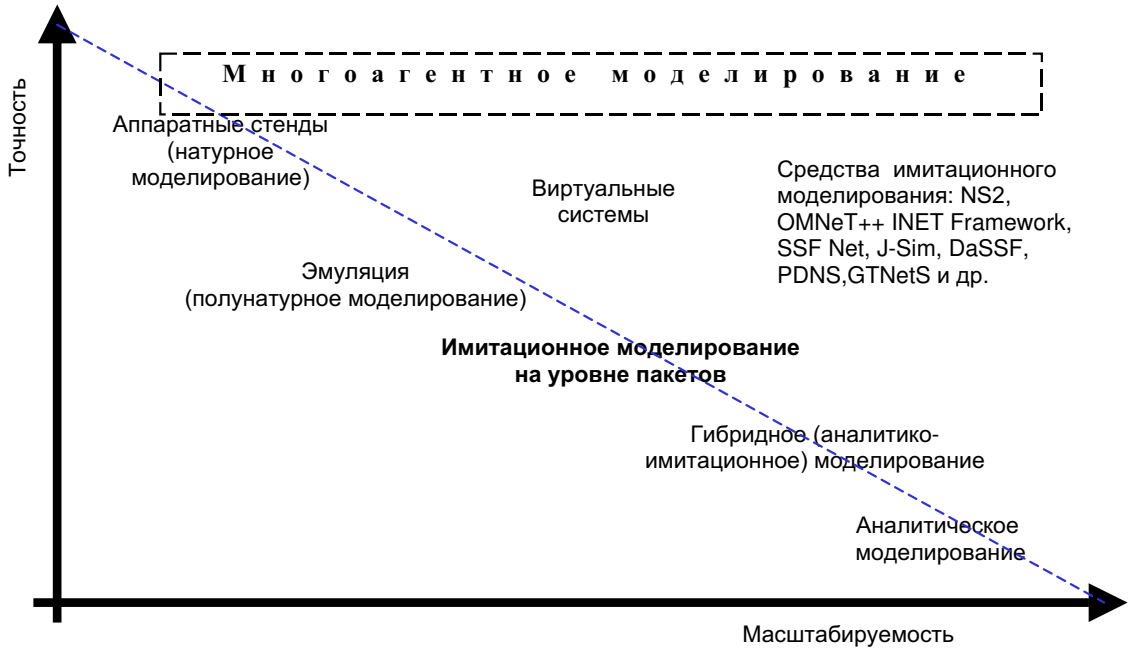


Рис. 1: Семейство моделей, используемых для исследовательского моделирования компьютерного противоборства

В работе развивается агентно-ориентированный подход к моделированию противоборства злоумышленников и систем защиты в виде антагонистического взаимодействия команд программных агентов, сформулированный в [1, 2, 3].

Работа выполняется при финансовой поддержке РФФИ (проект № 04-01-00167), программы фундаментальных исследований ОИТВС РАН (контракт № 3.2/03) и при частичной финансовой поддержке, осуществляемой в рамках проекта Евросоюза POSITIF (контракт IST-2002-002314).

Основное внимание в работе уделяется представлению разработанной программной среды (полигона) для многоагентного моделирования указанного противоборства, базирующегося на принципах имитационного моделирования на уровне пакетов (рис. 1), и описанию экспериментов по имитации распределенных атак «отказ в обслуживании» (атак DDoS), направленных на нарушение доступности информационных ресурсов, и механизмов защиты, реализующих их обнаружение, предотвращение и проактивное реагирование на атаки.

2 Подход к моделированию

Использование основанного на многоагентных технологиях моделирования процессов защиты информации в сети Интернет предполагает, что кибернетическое противоборство представляется в виде взаимодействия различных команд программных агентов[1, 2]. Выделяется, по крайней мере, две команды агентов, воздействующих на компьютерную сеть, а также друг на друга: команда агентов-злоумышленников и команда агентов защиты. Агенты различных команд соперничают для достижения противоположных намерений. Агенты одной команды сотрудничают для осуществления общего намерения.

Глобальная цель каждой из команд достигается совместными усилиями многих компонентов. Компоненты каждой из команд обладают следующими свойствами: автономность; наличие исходных знаний о себе, взаимодействующих сущностях и внешней среде; наличие знаний или жесткого алгоритма, позволяющего получать и перерабатывать внешние данные из среды; наличие цели и списка действий для достижения этой цели; осуществление коммуникаций для достижения общей цели.

Существует ряд подходов к организации командной работы агентов. Основополагающими являются классические подходы: теория общих намерений [4], теория общих планов [5] и комбинированный подход [6]. В теории общих намерений команда агентов обладает общими обязательствами и намерениями. Агенты имеют индивидуальные обязательства, которые являются их долговременной целью. Индивидуальные намерения агентов заключаются в выполнении этой цели. Групповой план является основой теории общих планов. Этот план задает совместное выполнение некоторого множества действий группой агентов. Команда агентов должна прийти к соглашению по выполнению групповых действий. Комбинированная теория объединяет оба подхода.

Многие подходы к организации командной работы агентов реализованы в программных реализациях различных многоагентных систем. Система GRATE* [7] является реализацией модели командной работы с общей ответственностью. В основу OAA [8] положены понятия: «доска объявлений» (blackboard) для агентских коммуникаций и ассистента (facilitator), управляющего ей. Основная идея системы CAST [9] заключается в использовании общей ментальной модели агентов для проактивного обмена информацией в целях эффективного командного поведения. В модели командной работы RETSINA-MAS [10] предполагается, что у всех агентов есть своя собственная копия частичного плана, чтобы они могли оценить свои возможности и выбрать подходящие роли. В «Robocup Soccer» [11] агенты имеют общие правила и знания, а также индивидуальные модели мира, которые управляют их кооперативным поведением. COGNET/BATON [12] — система для моделирования командной работы людей с использованием интеллектуальных агентов.

Предлагаемый подход к организации командной работы агентов базируется на совместном использовании элементов теории общих намерений, теории разделяемых планов и комбинированных подходов и учитывает опыт программной реализации многоагентных систем.

Структура команды агентов описывается в терминах иерархии групповых и индивидуальных ролей [1]. Листья иерархии отвечают ролям индивидуальных агентов, промежуточные узлы — групповым ролям. Механизмы взаимодействия и координации агентов базируются на трех группах процедур: (1) обеспечение согласованности действий; (2) мониторинг и восстановление функциональности агентов; и (3) обеспечение селективности коммуникаций (для выбора наиболее «полезных» коммуникационных актов).

Спецификация иерархии планов действий осуществляется для каждой из ролей. Для каждого плана описываются: начальные условия, когда план предлагается для исполнения; условия, при которых план прекращает исполняться; действия, выполняемые на уровне команды, как часть общего плана. Для групповых планов явно выражается совместная деятельность.

3 Атаки DDoS и механизмы защиты от них

В данной работе проверка предложенного подхода к многоагентному моделированию компьютерного противоборства была осуществлена на основе моделирования атак DDoS и механизмов защиты от них.

Концепция атаки DDoS заключается в том, что глобальная цель — «отказ в обслуживании» некоторого ресурса, достигается совместными усилиями многих компонентов, действующих на стороне атаки. Таким образом, исходная задача разбивается на более простые, которые поручаются отдельным специализированным компонентам. При этом на верхнем уровне цель остается общей для всех. На нижнем уровне формируются локальные цели, достижение которых направлено на решение общей задачи. Компоненты взаимодействуют между собой для координации локальных решений, что необходимо для достижения требуемого качества решения общей цели «отказ в обслуживании».

Известно несколько видов атак DDoS. Условно их можно разделить на две категории: истощение ресурсов сети и истощение ресурсов хоста. Атаки осуществляются с помощью посылки жертве большого количества пакетов (например, UDP и ICMP flood, а также Smurf, Fraggle — через промежуточные узлы), слишком длинных пакетов (Ping Of Death), некорректных пакетов (Land), большого количества трудоемких запросов (TCP SYN) и др.

Построение эффективной системы защиты от атак DDoS является сложной задачей. Стандартной мерой защиты подсети (не только от DDoS атак) является установка правил фильтрации любых пакетов от зарезервированных IP адресов (например, для сетевых пакетов, входящих с адресами из внутренней подсети, выходящих с адресами, отличающимися от внутренних, необычных по размеру; к тем и от тех портов, которые не задействованы в системе; по неиспользуемым протоколам и др.). Кроме того, применяется ограничение на трафик для каждого протокола и для входящих/выходящих потоков.

Зная эти меры, злоумышленник может использовать такие параметры атаки DDoS, что ее будет невозможно отличить от, например, запросов пользователей, вызванных повышенным интересом к данному серверу. Это приводит к усложнению механизмов защиты.

Общий подход к защите от атак DDoS заключается в следующем. Осуществляется сбор информации о нормальном для данной сети трафике с помощью сенсоров. Затем компонентом-анализатором в режиме реального времени осуществляется сравнение текущего трафика с модельным. Система пытается проследить источник аномалий (с помощью механизмов отслеживания («traceback»)) и выдает рекомендации по их отсечению или снижению их количества. В зависимости от выбора администратора безопасности (пользователя системы) системой применяется та или иная контрмера.

Можно выделить две основных задачи систем защиты: обнаружение атаки и непосредственно противодействие этой атаке.

Механизмы обнаружения атаки можно классифицировать по месту расположения и по способу обнаружения. Компоненты обнаружения могут располагаться в атакуемой сети, в исходной или промежуточной подсетях. Так или иначе, обнаружение атаки происходит в результате сравнения текущего трафика с модельным. Модель нормального для данной сети трафика строится на основе доступных данных: либо явно, либо после обработки на основе какого-либо метода. Эта модель строится, как правило, по нагрузке [13, 14, 15, 16, 17], по сигнатуре [18, 19, 20], по статистике [21, 22, 23, 24, 25, 26, 16, 17, 27], с использованием как традиционных статистических методов [28, 29], так и других методов (например, с использованием иерархической системы различных обучающихся классификаторов [30]).

Механизмы противодействия атакам DDoS можно классифицировать, как и механизмы обнаружения, учитывая место расположения и применяемый способ защиты. Место расположения определяется тем, для защиты какой подсети установлена данная система. Это может быть подсеть цели атаки, исходная или промежуточная подсеть. Эффективно построенная система противодействия, кроме собственной защиты, положительно влияет также и на остальную сеть в целом, например, блокируя внутри себя пакеты атаки. Способы защиты могут быть следующими: фильтрация пакетов (используется в большинстве случаев), фильтрация потоков [26], изменение количества ресурсов [31], перенос ресурсов [13], разграничение ресурсов [32, 33, 34, 27], аутентификация [13, 31, 35] и другие.

Дополнительно можно выделить три варианта применения фильтрации. Первый (традиционный) вариант — это стандартная фильтрация, выполняемая на одном хосте. Второй — с отражением («push-back») [14, 26, 15, 16, 17], когда фильтр применяется на каждой итерации все ближе к источнику атаки.

Третий — с отслеживанием («traceback») [36, 37, 38, 22, 39, 23, 24], когда источник атаки отслеживается, и фильтр применяется на ближайшем к нему хосте (маршрутизаторе).

4 Команда агентов атаки

Агенты атаки подразделяются, по крайней мере, на два класса: «демоны», непосредственно реализующие атаку, и «мастер», выполняющий действия по координации остальных компонентов системы.

На предварительном этапе демоны и мастер устанавливаются на доступные (уже скомпрометированные) узлы сети Интернет. Здесь важными параметрами являются количество и распределенность агентов. Затем происходит организация команды атаки: демоны посылают мастеру сообщения о том, что они существуют и готовы к работе, а мастер сохраняет информацию о членах команды и об их состоянии.

Злоумышленник задает общую цель команды — совершить атаку DDoS. Параметры атаки получает мастер. Его цель — разослать их всем доступным демонам. Далее в действие вступают демоны. Их локальная цель — исполнить команду мастера. Для этого на указанный узел отсылаются пакеты атаки с заданной мастером интенсивностью. После этого считается, что цель команды на данном этапе достигнута.

Периодически мастер опрашивает демонов, для того, чтобы узнать о том, что они находятся в работоспособном состоянии. Получая сообщения от демонов, мастер контролирует заданный режим выполнения атаки. Если от какого-либо демона не поступает сообщений о состоянии, мастер принимает решение об изменении параметров атаки. Например, он может послать команды всем или только определенным демонам об изменении интенсивности атаки.

Демоны могут выполнять атаку в различных режимах. Это влияет на возможности команды защиты по обнаружению и блокированию атаки, а также прослеживанию и устранению агентов атаки. Демоны могут отправлять пакеты атаки с различной интенсивностью, подменять адрес отправителя и делать это с различной частотой.

Злоумышленник может прекратить атаку. Он задает мастеру команду «завершить атаку». Затем мастер рассылает соответствующие команды демонам. Получив эту команду, демоны прекращают атаку.

5 Команда агентов защиты

В соответствии с общим подходом, выделены следующие классы агентов защиты [3]: первичной обработки информации («сенсоры»); обнаружения атаки («детекторы»); фильтрации («фильтры»); агенты-«расследования».

Рассмотрим основные функции этих агентов в одном из экспериментов, описанном в настоящей статье. В других экспериментах эти функции могут быть расширены, и возможно добавление дополнительных классов агентов.

В начальный момент времени агенты защиты устанавливаются на соответствующие их ролям узлы:

- сенсор — на пути следования трафика для защищаемого узла;
- детектор — на любой узел в подсети защищаемого узла;
- фильтры — на входе в подсеть защищаемого узла;
- агент расследования — за пределами подсети защищаемого узла на любом доступном из Интернета узле.

Общая цель команды агентов защиты — противостояние атаке DDoS. За ее выполнением следит детектор.

Сенсор обрабатывает информацию о сетевых пакетах и собирает статистические данные по трафику для защищаемого узла. Сенсор определяет величину всего трафика (BPS), а также адреса n узлов, создающих наибольший трафик (в реализованном прототипе — все хосты). Его локальная цель — предоставлять эти данные каждые k секунд детектору на обработку.

Локальная «цель» детектора — на основе данных от сенсора принять решение о наступлении атаки. В описанных в работе экспериментах, если детектор определяет, что параметр BPS превышает заданный предел (определяемый как процент от максимальной скорости пропускания канала связи), то он считает происходящее атакой DDoS. Он посылает свое решение и адреса n узлов, создающих наибольший трафик, фильтру и агенту расследования.

Локальная цель фильтра — выполнить фильтрацию трафика на основе данных от детектора. Если в сообщении содержится решение о проведении атаки, то фильтр начинает отбрасывать пакеты от указанных узлов.

Цель агента расследования — идентифицировать и вывести из строя агентов атаки. После приема сообщения от детектора он проверяет указанные адреса на наличие агентов команды атаки и пытается вывести идентифицированных агентов из строя. Для упрощения модели сделано допущение, что вероятность вывода из строя 30%.

При обнаружении атаки детектор посылает фильтру для фильтрации адреса узлов, создающих наибольший трафик. Как только по информации от сенсора детектор решит, что атака прекратилась, цель команды агентов защиты на заданном временном промежутке будет достигнута.

6 Среда моделирования

Для выбора инструментария моделирования сети и процессов передачи информации был проведен анализ пакетов моделирования (имитаторов сетей), включая NS2 [40], OMNeT++ INET Framework [41], SSF Net [42], J-Sim [43] и других. Был выдвинут ряд требований, которые предъявлялись к используемому имитатору, в частности, детальная реализация протоколов, начиная от сетевого уровня (для возможности моделирования основных классов сетевых атак), возможность написания и подключения собственных модулей для реализации агентского подхода, развитый графический интерфейс и др. Было выявлено, что этим требованиям в наибольшей степени удовлетворяет OMNeT++ INET Framework.

Система OMNeT++ представляет собой симулятор дискретных событий [41]. События происходят внутри простых модулей (simple modules). Обмен сообщениями между модулями осуществляется по каналам (модули соединены с ними шлюзами) или непосредственно через шлюзы.

На основе INET Framework разработана среда для многоагентного моделирования механизмов защиты и атак DDoS. Для этого система подверглась нескольким модификациям, в том числе были созданы: таблица фильтрации пакетов на сетевом уровне для моделирования действий стороны защиты; модуль, позволяющий просматривать весь трафик данного узла для ведения статистики, а также для моделирования действий стороны защиты. Подверглись изменению модули, отвечающие за работу Sockets для моделирования механизмов атаки. Ядра агентов были выполнены на основе сопрограмм, так как это удобно для реализации протоколов взаимодействия, на которых основана командная работа агентов. Остальные модули выполнены как обработчики сообщений от ядра и внешней среды.

Пример пользовательского интерфейса среды моделирования показан на рис. 2.

На основном окне визуализации (рис. 2, справа вверху) отображается компьютерная сеть для проведения моделирования. Она представляет собой набор хостов, соединенных каналами связи. Хосты могут нести различную функциональность в зависимости от их параметров или набора внутренних модулей. Внутренние модули отвечают за работу протоколов и приложений на различных уровнях модели OSI. Хосты соединяются между собой каналами связи, параметры которых можно изменять. Приложения (в том числе и агенты) устанавливаются на хосты, подключаясь к соответствующим модулям протоколов.

Окно управления процессом моделирования (внизу посередине рис. 2) позволяет просматривать и менять параметры моделирования. Для отображения текущего состояния команд агентов служат соответствующие окна состояний (сверху посередине рис. 2). Можно открывать различные окна, характеризующие функционирование (статистические данные) отдельных хостов, протоколов и агентов. Например, на рис. 2 внизу слева отображено окно функционирования одного из хостов.

Компьютерная сеть для проведения моделирования состоит из трех подсетей: (1) подсеть защиты, на K узлах которой устанавливаются агенты защиты, и в которой можно выделить защищаемые серверы; (2) промежуточная подсеть, состоящая из N хостов с типовыми клиентами, генерирующими нормальный трафик; (3) подсеть атаки, включающая M узлов с демонами и один узел с мастером. Размеры подсетей задаются соответствующими параметрами моделирования.

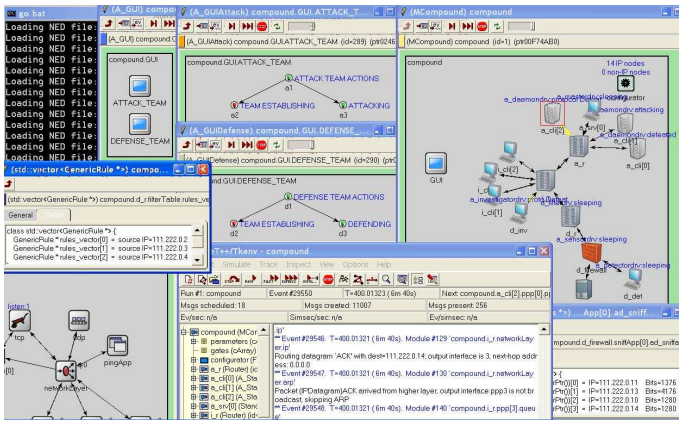


Рис. 2: Пример пользовательского интерфейса среды моделирования

7 Эксперименты

На примере моделирования процессов реализации распределенных атак «отказ в обслуживании» проведен ряд экспериментов.

Рассмотрим пример одного из сценариев моделирования. Сеть для проведения моделирования изображена на рис. 2 (справа вверху). Маршрутизаторы в этой сети соединены между собой волоконно-оптическими каналами связи со скоростью передачи 512 Мбит. Остальные узлы соединены Ethernet 10 Мбит каналами связи.

Через некоторое время после запуска процесса моделирования клиенты начинают посылать запросы серверу, а он на них отвечать. Так происходит генерация нормального трафика.

Через некоторое время после запуска симуляции происходит формирование команды защиты. Агенты расследования, сенсор, фильтр соединяются с детектором и посылают ему сообщения о своей работоспособности. Детектор заносит данные о них в память. Формирование команды атаки происходит аналогичным образом.

После формирования команды защиты, начинаются командные действия. Сенсор начинает сбор данных по трафику (количество переданных бит) для каждого адреса. Детектор опрашивает сенсор (например, каждые 60 сек.), получает от него данные, и определяет, не происходит ли атака. Затем он соединяется с фильтром и агентом расследования и сообщает им IP-адреса подозрительных узлов.

При начале атакующих действий мастер опрашивает всех демонов, выясняя их работоспособность. После того, как все демоны были проверены, оказалось, что они все в рабочем состоянии. Мастер вычисляет индивидуальную интенсивность атаки каждого демона. После этого мастер отсылает каждому демону команду атаки. Демоны приступают к атаке.

Сенсоры посылают детектору список IP-адресов и количество переданных бит за определенное время. Детектор определяет IP-адреса хостов, с которых передается трафик, превышающий максимально допустимый объем. Детектор отсылает фильтру эти адреса для фильтрации трафика, а агенту расследования — для отслеживания агентов атаки и их нейтрализации. После применения фильтром правил запрета на прохождение пакетов от данных адресов, трафик к серверу снижается. Агент расследования пытается обезвредить агентов атаки. В результате агенту расследования удается удалить двух демонов. Оставшийся демон продолжает атаку. Мастер перераспределил на него нагрузку. Однако пакеты атаки не доходят до цели, а фильтруются на входе в защищаемую сеть.

На рис. 3 приведен график зависимости объема трафика, переданного в подсеть сервера, от времени. В промежутке времени (0–300) секунд основной трафик создавался обращениями клиентов к серверу и его ответами. Этот процесс отмечен вертикальными прямыми с низкой интенсивностью. По наступлении атаки (отметка 300 секунд) появился интенсивный трафик — плато от 300 до 400 секунд. Однако примерно на 400 секунде моделирования были применены фильтры, и пакеты атаки стали отбрасываться на входе в сеть сервера. После этого возобновилась нормальная картина.

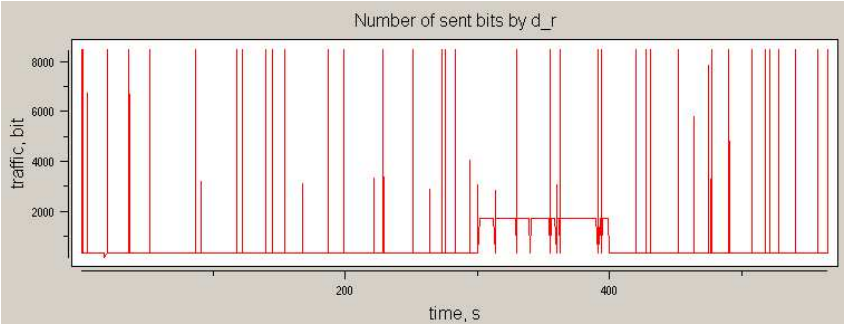


Рис. 3: Зависимость объема трафика от времени

8 Заключение

В данной работе была рассмотрена многоагентная среда моделирования противоборства команд программных агентов злоумышленников и агентов защиты в среде Интернет. Программная среда разработана на базе C++ и OMNET++ INET Framework. Реализованы различные классы атак и механизмов защиты от них. На примере моделирования процессов реализации распределенных атак «отказ в обслуживании» проведен ряд экспериментов. Эксперименты показали эффективность предлагаемого подхода и возможность его использования для моделирования перспективных механизмов защиты и анализа уровня защищенности проектируемых сетей.

В дальнейшем планируется развитие предложенных моделей противоборства, в том числе разработка формальных моделей антагонистического взаимодействия команд агентов защиты и нападения, реализация большего количества механизмов защиты и атак, в том числе дополнительных атак DDoS и механизмов защиты от них, оценка эффективности разработанных механизмов защиты, выработка рекомендаций по построению эффективных механизмов защиты от DDoS-атак, дальнейшее развитие среды моделирования, исследование и совершенствование механизмов внутрикомандного взаимодействия агентов, а также развитие механизмов адаптации и самообучения агентов.

Литература

[1] Котенко И. В. Многоагентные модели противоборства злоумышленников и систем защиты в сети Интернет // Третья Общероссийская Конференция «Математика и безопасность информационных технологий» (МаБИТ-04). М: МГУ, 2004.

[2] Kotenko I. Agent-Based Modeling and Simulation of Cyber-Warfare between Malefactors and Security Agents in Internet // 19th European Simulation Multiconference «Simulation in wider Europe». ESM'05. 2005.

[3] Kotenko I., Ulanov A. Multiagent modeling and simulation of agents' competition for network resources availability // Second International Workshop on Safety and Security in Multiagent Systems. SASEMAS'05. 2005.

[4] Cohen P., Levesque H.J. Teamwork // Nous, No. 35, 1991.

[5] Grosz B., Kraus S. Collaborative Plans for Complex Group Actions // Artificial Intelligence, Vol. 86, 1996.

[6] Tambe M. Towards flexible teamwork // Journal of AI Research, Vol.7, 1997.

[7] Jennings N. R. Controlling cooperative problem solving in industrial multi-agent systems using joint intentions // Artificial Intelligence, Vol.75, No. 2, 1995.

[8] Martin D., Cheyer A., Moran D. The open agent architecture: A framework for building distributed software systems // Applied Artificial Intelligence, Vol.13, No.1–2, 1999.

[9] Yen J., Fan X., Sun S., Wang R., Chen C., Kamali K., Miller M., Volz R.A. On Modeling and Simulating Agent Teamwork in CAST // Proceedings of the Second International Conference on Active Media Technology, 2003.

[10] Giampapa J.A., Sycara K. Team-Oriented Agent Coordination in the RETSINA Multi-Agent System // Tech. report CMU-RI-TR-02-34, Robotics Institute, Carnegie Mellon University, December, 2002.

[11] Stankevich L. A cognitive agent for soccer game // Proceeding of First Workshop of Central and Eastern Europe on Multi-agent Systems (CEEMAS'99). 1999.

[12] Fan X., Yen J. Modeling and Simulating Human Teamwork Behaviors Using Intelligent Agents // Journal of Physics of Life Reviews, Vol.1, No.3, 2004.

[13] Sangpachatanaruk C., Khattab S.M., Znati T., Melhem R., Mosse' D. Design and Analysis of a Replicated Elusive Server Scheme for Mitigating Denial of Service Attacks // Journal of Systems and Software, Vol.73(1), 2004.

[14] Keromytis A., Misra V., Rubenstein D. SOS: Secure Overlay Services // Proceedings of ACM SIGCOMM'02, Pittsburgh, PA, 2002.

[15] Peng T., Leckie C., Kotagiri R. Defending Against Distributed Denial of Service Attacks Using Selective Pushback // 9th IEEE International Conference on Telecommunications, Beijing, China, 2002.

[16] Ioannidis J., Bellovin S.M. Implementing Pushback: Router-Based Defense Against DDoS Attacks // Proceedings of Symposium of Network and Distributed Systems Security (NDSS), San Diego, California, 2002.

[17] Manajan R., Bellovin S.M., Floyd S., Ioannidis J., Paxson V., Shenker S. Controlling High Bandwidth Aggregates in the Network // ICSI Technical Report, July 2001.

[18] Peakflow Platform. Arbor Networks. http://www.arbornetworks.com.

[19] DDoS-Guard. Green Gate Labs. http://www.ddos-guard.com.

[20] Prolexic Solutions. Prolexic. http://www.prolexic.com.

[21] Jin C., Wang H., Shin K.G. Hop-count filtering: An effective defense against spoofed DDoS traffic // Proceedings of the 10th ACM Conference on Computer and Communications Security, 2003.

[22] Law K.T., Lui J.C.S., Yau D.K.Y. You Can Run, But You Can't Hide: An Effective Methodology to Traceback DDoS Attackers // Proceedings of the 10th IEEE International Symposium on Modeling, Analysis, & Simulation of Computer & Telecommunications Systems. MASCOTS'02. 2002.

[23] Snoeren A.C., Partridge C., Sanchez L.A., Jones C.E., Tchakountio F., Schwartz B., Kent S.T., Strayer W.T. Single-Packet IP Traceback // IEEE/ACM Transactions on Networking, Vol.10, No.6, 2002.

[24] Li J., Sung M., Xu J., Li L. Large-scale IP traceback in high-speed Internet: Practical Techniques and theoretical foundation // Proceedings of the IEEE Symposium on Security and Privacy. S&P'04. 2004.

[25] Cabrera J.B.D., Lewis L., Qin X., Lee W., Prasanth R.K., Ravichandran B., Mehra R.K. Proactive detection of distributed denial of service attacks using mib traffic variables — a feasibility study // Proceedings of International Symposium on Integrated Network Management, 2001.

[26] Xuan D., Bettati R., Zhao W. A Gateway-Based Defense System for Distributed DoS Attacks in High Speed Networks // Proceedings of the 2nd IEEE SMC Information Assurance Workshop, West Point, NY, June, 2001.

[27] Mirkovic J., Prier G., Reiher P. Attacking DDoS at the Source // Proceedings of ICNP 2002, Paris, France, 2002.

[28] Kang J., Zhang Z., Ju J. Protect E-Commerce against DDoS Attacks with Improved D-WARD Detection System // Proceedings of 2005 IEEE International Conference on e-Technology, 2005.

[29] Xiang Y., Zhou W. An Active Distributed Defense System to Protect Web Applications from DDOS Attacks // Proceedings of the Sixth International Conference on Information Integration and Web-based Applications Services, iiWAS'2004. Jakarta, Indonesia, 2004.

[30] Gorodetsky V., Karsaev O., Samoilov V., Ulanov A. Asynchronous alert correlation in multi-agent intrusion detection systems // Lecture Notes in Computer Science, Vol.3685, 2005.

[31] Wang X., Reiter M.K. Mitigating bandwidth-exhaustion attacks using congestion puzzles // Proceedings of the 11th ACM Conference on Computer and Communications Security, 2004.

[32] Mankins D., Krishnan R., Boyd C., Zao J., Frentz M. Mitigating Distributed Denial of Service Attacks with Dynamic Resource Pricing // Proceedings of the 17th Annual Computer Security Applications Conference. ACSAC'01. 2001.

[33] Bernet Y., Binder J., Blake S., Carlson M., Carpenter B., Keshav S., Davies E., Ohman B., Verma D., Wang Z., Weiss W. A Framework for Differentiated Services // IETF Internet Draft, 1999.

[34] Wang H., Shin K.G. Transport-aware IP Routers: A Built-in Protection Mechanism to Counter DDoS Attacks // IEEE Transactions on Parallel and Distributed Systems, Vol.14, No.9, 2003.

[35] Wang H., Bose A., El-Gendy M., Shin K.G. IP Easy-pass: Edge Resource Access Control // Proceedings of IEEE INFOCOM'04, Hong Kong, 2004.

[36] Gemberling B.W., Morrow C.L., Greene B.R. ISP Security — Real World Techniques // Presentation, NANOG, October 2001.

[37] Perrig Y.A., Song D. Pi: A path identification mechanism to defend against DDoS attacks // Proceedings of the 2003 IEEE Symposium on Security and Privacy, 2003.

[38] Bellovin S., Leech M., Taylor T. ICMP Traceback Messages // Internet-Draft draft-ietf-itrace-01.txt, October 2001.

[39] Savage S., Wetherall D., Karlin A., Anderson T. Practical network support for ip traceback // Proceedings of the 2000 ACMSIGCOMM Conference, Stockholm, Sweden, August 2000.

[40] NS-2 homepage. http://www.isi.edu/nsnam/ns/.

[41] OMNeT++ homepage. http://www.omnetpp.org.

[42] SSFNet homepage. http://www.ssfnet.org.

[43] J-Sim homepage. http://www.j-sim.org.

Public key infrastructure protection of facilities ad networks

L. Eilebrecht

I would like to start with quick introduction on what system actually does and how it works. I am going to talk about the motivation and the threat model. I am going to explain the concept and what the system is supposed to achieve followed by some implementation details and attack scenarios of how the system works.

As you all probably already know, most solutions use public key or most solutions for encryption and signature use actually some form of public key cryptography but it can only make sense if you trust the public key you are using to encrypt some piece of data. What really matter is the integrity and atomicity of the public key. These systems usually use so called trusted third party as you may know a certification authority (CA) actually issue a public key. So actually have an issued signature and clients verify the issued signature and the authority it indicates the certificate. The problem with that is clients need to trust the trusted party (that where the name comes from). The problem with it is that such trust relationship is not desirable or not even possible in some environments. So what the CA3 finger print system, as it is called, provides a mechanism for clients to indicate the original certificate without relying on direct trust relationship with central authority, with a CA, who issues certificates. So author identification can be performed by client without using or relying on the issued signature of the certificate. So clients are actually enabled to detect malicious changes to certificates.

Before I continue explaining how the system actually works, brief note about related work. We are not aware of any work in direct relation to the system we have developed, of course there are plenty of solutions and have been plenty of research and systems and solutions that try to avoid trust relationship with central authority. Some examples include system policy maker, key note or ASPKI system. The example you may very well know is open PGP which uses the web of trust to indicate public keys. The problem with work of trust is that very often people have to revert to performing a manual verification of a public key. So they have manually verified the fingerprint, the hash of the public key. The fingerprint system is basically based on one way hash function, so we are using hash raining techniques. They are known from time stamping services. But before I explain the details of hash training and the system itself, let's look at the motivation and what the threat actually is.

Certificates signed by a trusted third party are very helpful in limiting threats like a creation of fake certificates because clients can rely on issuer signature. However the problem is the insider who has access to the CA, to the certification authority and especially to the certification authority's private keys and to the related PKI systems, can basically modify a certificate, issue new certificate or revoke a certificate. Such individual can already have an access to the system because he/she may be the system administrator taking care of the service. There is also a threat that somebody has actually gained an access to the authorized service and modified some certificates. So in one way or in other the trusted third party becomes an adversary either deliberately or unintentionally. But changing the certificate doesn't do much harm itself, this is usually using the combination of so-called man in the middle attack, meaning if someone changes a certificate, especially the public key, included in the certificate, and then will be able to trick users into using this fake or misfortuned certificate, would be able with combination of man in middle attack to read victims encrypted communication because he can re-encrypt the data sent to the wrong public key and no one would notice.

So what do we need in order to prevent this or to avoid such a trust relationship? Basically what this system does is not the preventing the initial attack because that is not possible; someone with access to database or certificates storage or issue can always do something manually. What the system does, it makes it impossible for an attacker, for insider to hide such an attack. So we enable the clients to detect such malicious changes and to use the system to indicate certificates without relying on trusted relationship, without relying on the issuer signature of the certificate. In order for a PKI client to do this, he needs to be able to perform certain verifications. This list of verifications requires to actually doing that. First of all the client needs to prove the integrity of the certificate to make sure that it is still the original certificate, issued in the very beginning, that has not been modified. A part from that a client needs to able to verify that the

certificate has not been revoked without consent of the user. In addition the client needs to be able to verify if the certificate exists or not, that the CA can not deny the issue of the certificate. If the PKI operated in a closed environment, where we expect or require every client, every user to have a certificate, we don't need it; otherwise it will also be helpful for client to be able to obtain proof of the non-existence of the certificate. So if somebody simply doesn't have a certificate, it will be helpful for a client to actually obtain the proof that really no certificate exists versus that it has just been removed from the database. In addition the client needs to be able to verify that no duplicate certificates for given entity, for given user exists. And in order to perform these verifications, the clients need specific data to do that, I'll explain later how exactly that works. And of course clients being able to verify and validate the data, so that they are sure that they are using the correct data for verifications. What the system is intended to be is to serve as an add on to any PKI technology that it being used. Of course there are some requirements. First of all we need support on the server and client side. Especially on the client side we have to look at the usability. So we don't want any manual verifications, so the user should not be and must not be required to perform manual verifications as a fingerprint check as you might know from PGP when you verify a public key. So the system must provide procedures and mechanisms to automate all the verifications. In addition the client needs to download the data for doing the verifications. So the amount of data that is required for each client needs to be reasonable. We can not require a client to download megabytes or gigabytes of data just to verify this single certificate. And of course the process itself needs to be reasonable as well in order to ensure usability. In addition the system tries to be usable in for the PKI arbitrary number of users. Basically, let's say, several millions of users in PKI. But of course the system is not just an add on or like a plain solution that you just add to existent PKI, there are some constraints and requirement for PKI. One is that the PKI environment has to ensure that no duplicate certificate for an entity allowed, so there must be only one active certificate at the given time. Of course user can have multiple certificates but they need to have different information, different user ID, different e-mail address or different information about the use of the certificate. In addition, especially if the PKI domain includes multiple CA's, we need a unique identifier for each certificate. In this case it can be just a serial ID of the certificate. And of course the whole system of protecting against the inside attacks makes only sense if the private keys are activated by the client himself. Because otherwise he doesn't need to worry about the inside attacks because private key are created and stored centrally.

I would like to focus on the concept and explain the concept with a simple solution first and later mention some of the implementations details. From the abstract point of view we have client and server side procedures that are required for the system. So basically on the CA, on the server side we have a creation of the identification information for each certification actions performed by the certification authority. This information we will later call the certificate fingerprints. The CA needs the procedure to publish this information. Consequently the client needs this procedure to download identification information, to validate it and, of course, to validate, to indentificate the certificate using this information he has downloaded. So what exactly happens? As I have already mentioned the system is based on one way hash functions. Basically you can use any hash functions as SHA or Whirlpool. So whenever the CA issues a new certificate, renews a certificate or revokes a certificate, performs a certification action, a new fingerprint consisting of multiple hash values and some miter data has to be created by the CA. The values include a hash of the complete certificate, a hash of the certificate's unique identifier, as a SID (serial identifier), for example, hash of the certificate's subject data that can be user's ID, an e-mail address, a real name or a combination of this information. The certificate is actually revoked, we need the information of entry, the fingerprint entry of the relocation or the initial creation of the certificate, and we have in fact to indicate the revocation. Indicate the revocation depending on what kind of PKI technology, what kind of public key system you are using, you might not have a certificate that is issued for the revocation, in that case the hash of the certificate, won't be a hash of the certificate and will just be, for example, in case of X509 systems, a hash of the certificate revocation list entry. In addition to that we have a time stamp, which defines the exact time when the certification action took place, like when the certificate was issued by the CA. With every new fingerprint a new summary hashes is created, calculated over the previous hashes, previous fingerprints and a new fingerprint, a new hash. Together with the summary hash and a fingerprint, that is stored in so-called fingerprint list. This is a hash changing part. Hash changing ensures that you always include previous entries in your new hash calculations, that once you have edit more entries you can not change more entries to the list because otherwise you will invalidate new entries because the hash will no longer match. This data has to make available to the clients, it has to be published, it could be published in a form for intervals or some other form some kind of directory service like an ALDAP service that mostly duplicates just like a flat file. The similar system will be a certificate divulgation list, which just includes entries for certificates.

So once the clients have this information, have this indication information, they can use it to indicate the certificate because if they have a certificate they can calculate various hashes, calculate the fingerprint and compare it to the entry on the fingerprint list. The look up is based on the creation time. Of course that only makes sense if they are sure that they are using the correct fingerprint data. So the fingerprint data, if it provided by the central authority, the clients can not trust the fingerprints, they have to validate the fingerprint data otherwise it doesn't make sense to use fingerprint data for verifications. First of all when they download data they have to do the general verification of the integrity of the list, including the general verifications of the hashes, the data, and the structure of the list. Every client has one or more certificates, so the client is the most authoritative source actually to confirm that the entries actually corresponding to their own certificates is correct because the private key and the public key can calculated the fingerprint. In addition to that it becomes the most important system which is P to P element of the system; it is called cross client verification. Every client includes a summary hash, as I have just mentioned, which is created for each entry, includes summary hashes into communication to other clients. So we assume here that the public key infrastructure is used here to enable clients to communicate securely, it could be an e-mail, instant messaging, voice of IP. So it is every regular message to exchange a P with another user, with another entity of the PKI. A cope of the most count summary hash is included automatically by the client and include the corresponding time. So the receiving client verifies the summary hash against its local copy of the fingerprint data. This doesn't fully indicate the fingerprint data but as it is done over and over again with every new message, the trust to fingerprint system, the integrity and atomicity of the fingerprint data is increased. And this ensures it becomes very difficult to make modifications to the fingerprint system without causing security warnings for the clients because new entries would no longer match. For example, if the client has been compromised and has committed a wrong fingerprint data, the summary hash will no longer match. So the client will detect that something is wrong with his copy of the fingerprint data.

Let me give you a quick summary. Basically at the top we have the certification authority that creates certificates, issues certificates, revokes certificates and then in addition creates a fingerprint. We can have a dedicated fingerprint authority, which just takes care of the fingerprint list; it depends on how it is implemented. It could be the same system actually. So the fingerprint authority creates a list and makes sure that they are available to the clients like a fingerprint directory or some other form that clients are able to download the information. The PKI clients of the users download regularly or on demand the fingerprint data from the central directory and while they communicate they always include summary hash in the e-mail or in instant messaging communication and verify that automatically. So over time we basically have every client verifying that the fingerprint data is actually the same fingerprint data used by every client in a PKI system. Most importantly every client checks his own certificates and thus ensures that these entries are correct in the system. Of course a single fingerprint list and creating summary hash with every certification action would not work in large PKI if you have more than just few users. That will simply not work because client will be required to download a lot of data.

I only briefly talked about the implementation details because covering each art will take too much time but basically what we did is not a single list. We partition the list into smaller lists, basically using interval (it could be 10 minutes, 1 hour, a day) and just for each interval, which includes an arbitrary number of certification actions performed by the CA, we create multiple lists in a tree like structure for each of these intervals with leaves of the tree basically including the fingerprint, the actual fingerprint entries. At the root we have a summary hash, which is then added to a single list, which goes over time and just includes the summary hash, the corresponding time stamp. So we don't have a summary hash, a new entry for each certification actions but just a defined number based on the interval length we are using in our list. So clients can of course still choose to download all data but they can also choose to limit the data they are going to download just to the main list or just to certain information, to specific interval, to verify a particular certificate but it could be implemented on a demand way like client who is about to use the certificate can just download the data, verify the data and indicate the certificate. In extension to that is so-called fingerprint tree. The normal fingerprint list system, which I have just explained, requires the client to have a certificate and a creation time to look at the entry. If we operating the PKI in the closed environment, where every user is required or supposed to have a certificate. We don't need a fingerprint tree. If the client has only user ID, e-mail address or domain name to perform a look up in order to obtain the certificate, it will be helpful if the client is able to prove the response from the certificate directory. For example, if the central server tells the clients there is no certificate, the fingerprint tree can be used by the client to verify that really no certificate exists. Basically what we do is we use the hash of the user ID and use the hash value to create a hash ID, define depth of the tree and the leaves will contain the actual fingerprints. The whole

fingerprint can be included or could be limited to adjust the hash of the user ID and the corresponding time stamp. So these also serve the purpose if have a list with fingerprints that the client can verify, if it contains the duplicate entry for the same user ID, which is not supposed to be allowed, of course, it has to be rebuilt with every new interval. It can be different for every fingerprint, this I have explained earlier, but basically time intervals have to be recreated. And again we have a single list just containing the summary hashes of the tree that basically will be a summary hash of the summary of the each leave lists. This again will be used for cross client verification in order to make sure every client has the same fingerprint tree information.

Let's look at the detection of the inside attacks on some example scenarios how it actually works. The general scenario, as follows, we basically have Allis and Bob, our favorite colleges, who are the users of the same PKI, which uses the CA3 fingerprint system. So Allis and Bob communicate by e-mail and, of course, they would like to do it securely. Eve tries to compromise their communication either by attacking Allis or Bob or actually both. Eve is supposed to be an insider with an access to the CA private keys and access to the central PKI system central service.

So the first scenario. Allis is about to send an e-mail to Bob for the first time. So she has to obtain the certificate. Eve replaces Bob's certificate in the central directory, which stores all the certificates, using a fake one, like a certificate with modified or changed the public key, so Allis's client actually obtained a fake certificate from Bob. Allies can use the creation time of the certificate to locate the corresponding entry in the fingerprint list. Allis can cable the fingerprint hashes and compass them with entry in the fingerprint list. In this case there will be no match because the entry is simply not in the fingerprint data. So Allis's client is supposed to display a warning or actually prevent Allis from sending an e-mail to Bob. Of course Eve is very clever and tries to modify the fingerprint data. What happens? We have the same scenario basically, as I have explained earlier, if the Allis's client has a copy of the fingerprint data, so later the modification will make no change because each client is supposed to download and verify fingerprint data once and then keep it. If Eve changes fingerprint data or try to fix the fingerprint list by fixing all the succeeding hashes in order to make sure the integrity of the fingerprint data is till there. Actually all clients will notice immediately that they have some new data compared to some data that have just been modified. So many many clients will show a warning that something with the fingerprint data is wrong because when downloading a new part form the fingerprint data will not match the fingerprint data they have downloaded earlier. This is considered the very end that Eve is able to feed Allis with specific data, modified data. May be if she is controlling Allis's communication, for example. So Allis's client has wrong data, Allis might end up using the fake certificate because the entry will be in the fingerprint system. However, with every message received from another user, the client will do the cross client verification, verification of the summary hash. So Allis's client will show the warning that something with a fingerprint data is not correct. Basically saying if such a warning appears. There are two possibilities. Either the client has seen the warning with corrupted data or compromised data or the other sender of the message has corrupted data. The client, who has corrupted data, will receive a warning for every message that is being received while the other one will receive one just for e-mail from only a specific user. So both clients know that something is wrong with a fingerprint data.

Another variant would be if Bob is just creating the certificate. Eve would be able before all the fingerprint data is published in the certification interval to actually change the entry in the fingerprint list. In that case the fingerprint would include fingerprint entry for the fake certificate. However the clients are supposed to self verify. So Bob who actually creates the certificate will verify that actually the correct entry appears in the fingerprint list and would also notice in that case that something is wrong, probably before Allis even considers sending an e-mail to Bob.

Let me give you a quick summary of the system. Basically the system ensures that for each certification action information is being made available to the clients. Somehow the clients are enabled to audit the certification actions of the central authority. They are enabled to detect if the certificates have been modified without the consent of the owner of certificate and if something is wrong with a fingerprint data itself as a system has chosen that every client has the same fingerprint data. Thus, the system allows a secure key exchange and allows this to be implemented in automotive way. So we don't have to rely on human doing the verification, doing a manual fingerprint check. So the system can also be used for autonomy systems, devices, where no user is involved, but still needs to ensure that the certificates, the public keys that they are using for encryption and securing communication are actually the correct system.

I would also like to note that the system is not meant to replace the conventional security properties of the certificate. So we still need a signature issuer, we still need a central certification authority. What the system actually does is removing from the authority factor from central systems. So we actually have a combination of hierarchical trust model with a distributed trust model as we are exchanging some of the

hashes between the clients without any central system being involved. So end up with hybrid trust model for the fingerprint system.

Question:
There is multilevel system of key distribution and a tree like system on one certification server. How are the hashes distributed from one leave to another?

Answer:
In every tree like structure we have a master list, which contains summary hashes of the list below that. So every note is actually a list. At the lowest level the leaves contain the fingerprints. We have a summary hash over these fingerprints which were created during that interval. The list is again based on the user ID, so they distributed over the branches of the tree. I am talking about the fingerprint list system now like the tree we create for a specific interval. So we have a summary hash for each of these leave lists, this is added to a hire list, which ends up in a master note, which contains summary hashes. A master of the summary hash is the one, which is used in the single list, which used for keeping summary hashes used form cross client verification.

Question:
What kind of data is used for a single fingerprint?

Answer:
The fingerprint basically consists of the multiple hashes. Most likely we a have a hash of the complete certificate, which allows the client to verify the integrity because if the clients has a certificate, he can calculate a hash and compare it to that value of the fingerprint in the fingerprint list. In addition used for looks up basically we have additional hashes with a serial ID, serial number. We have a hash of the user ID, a hash of an e-mail address. This also enables clients to look up entry in fingerprint tree system because it is based on the hash of the user ID. So when the user has an e-mail address and we would like to fetch a certificate for your e-mail address, I can calculate the hash and look the specific entry. In addition we have some middle data like a time stamp for the creation of the fingerprint.

Question:
When are the new fingerprints added? Is this related to sending a receiving message?

Answer:
For every certification action performed by the certification authority like creation or relocation of the certificate one fingerprint has to be created. At some point, if you do it in a specific interval and to publish a data, the client can download this information. The downloading of the fingerprint information and using the fingerprint data is completely unrelated to issuing the certificate or to actually using the certificate. In the very moment when the certificate is issued, clients don't have the fingerprints because it takes 10 minutes, an hour or a day before it is an actually included in one of the summary hashes and in the fingerprint system. The idea is that these entries are at some point in the fingerprint system. So the client can actually not use the fingerprint system at all and still rely on the normal issuer signature but the client has at some point the data in order to be able to perform an additional identification compared to normal verification of the issuer signature and can, this, use the fingerprint system data to indicate the certificate to make sure that it is an original certificate that was initially issued by the CA.

Question:
Does the system require a lot of resources?

Answer:
It depends, of course, on the number of the certification action you have in the system, on the umber of clients. If you have, let's say, 1000 users of PKI, especially in the closed system, we don't have much data because we don't usually regenerate the certificates every day. Even for the large public certification authority with several million users, the system is designed to require a client, a user to only download, let's say, several kilobytes of data. It will be indicating a particular certificate on average something between 20 or 30 or 40 kilobyte in order to verify the certificate, depending on how much data the client downloads in order to verify the certificate. Of course, if the client downloads all fingerprint list data which is not meant to do, then we have a data in a large PKI. But especially in closed environment, where we actually require every user to have a certificate and we don't have that many users, let's say, a few thousands, several ten thousands user may be, there is actually not that much data because it can be downloaded as a background, it doesn't have to happen when the user is sending a message, it can done regularly or in the background.

Question:
What kind of hash functions can we use within the system?

Answer:
Basically the system is unrelated to what kind of hash functions you would like to use. You can use SHA1, SHA2, and SHA256. Basically any function you would like to use. Basically you can use any hash function or any algorithm you would like to use. The system is solely based on hashes; the implementation also includes the signature from authority to make sure it so not too easy to create wrong fingerprint data.

Question:
Are there any special requirements to the communication channels?

Answer:
Basically it depends on client application, client's use of the communication and client's use for encryption. Basically any custom encryption solutions can be included. Basically what you need is some kind of miter data in a message to include in summary. It can even be some plain text information, en e-mail. If you are using a real time communication as a voice of IP, usually they have a settled base for communication where you can exchange the summary hashes, in some cases it might be even possible to exchange the hashes first to be sure the other person has a right setting before actually sending the data.

Question:
How often has been security violated? If there were cases of security violation, did they mostly occur in state certification centers or only in commercial centers?

Answer:
Most of the attacks are actually performed by the insiders. Some statistics claim it to be 70% or more. As I mentioned in the beginning the system was designed and focuses on environments, where we need a highest possible security, where we simply can not trust the insiders like a system administrator or may be have a fear that somebody with an inside access even an outsider who is obtaining an authorized access to the system. As far as if we have ever seen such examples of attacks. I don't know any examples of the public certification authorities, any specific cases, where certificates have been modified. But there are cases, let's say, when authorities try to use SSL, to actually use the certification authority, which is known to web-browsers and create a fake certificate. So the browser will just verify that based on the trust for the issuer signature and use the certificate. The fingerprint would not be able to detect that it is actually an incorrect fingerprint system. And, of course, in various closed environment there have been cases where people just modify certificates in order to trick user into using wrong certificates.

Аффинная эквивалентность и ее применение при изучении свойств дискретных функций

А. В. Черемушкин

В работе приводится обзор некоторых задач по изучению свойств дискретных функций, при решении которых удается эффективно использовать аффинную эквивалентность и классификационные результаты.

Введение

Многие задачи по изучению свойств дискретных функций близко связаны с аффинной или линейной эквивалентностью. Это может проявляться в различных ситуациях, например, в инвариантности определенных свойств относительно аффинных или линейных преобразований, в существовании более простых описаний или реализаций, полученных после применения таких преобразований, приближении функций аффинными функциями и т.п. Кроме того, аффинная группа позволяет строить классификации для функций, зависящих более чем от пяти переменных. Приведем примеры таких задач.

Обозначения

Пусть $n \geqslant 1$, $V_n(2) = \mathrm{GF}(2)^n$, $\mathcal{F}_n$ — множество двоичных функций от n переменных, $\mathrm{GL}(n,2)$ — полная линейная группа преобразований пространства $V_n(2)$ над полем $\mathrm{GF}(2)$, а $\mathrm{AGL}(n,2) = \mathrm{GL}(n,2)\,\mathrm{H}_n$ — полная аффинная группа, H_n — группа сдвигов пространства $V_n(2)$.

Для каждого целого $s \geqslant 0$ определим подпространства

$$\mathcal{U}_s = \mathrm{RM}(s,n) = \{f : \deg f \leqslant s\} \leqslant \mathcal{F}_n. \tag{1}$$

Так как $\mathcal{U}_0 = \{0,1\} \neq \{0\}$, то имеет смысл при $s < 0$ положить $\mathcal{U}_s = \{0\}$. Пусть также

$$\mathcal{U}_s^{(0)} = \mathrm{RM}_0(s,n) = \{f \in \mathcal{U}_s : f(0) = 0\}.$$

Действие группы G, $G \leqslant \mathrm{AGL}(n,2)$, на факторпространствах $\mathcal{U}_k/\mathcal{U}_s = \{f \oplus \mathcal{U}_s\}$, $-1 \leqslant s < k \leqslant n$, определяется обычным образом: $(f \oplus \mathcal{U}_s)^\alpha = f^\alpha \oplus \mathcal{U}_s$, $f \in \mathcal{U}_k$, $\alpha \in G$. Определим также группы

$$G\mathcal{U}_s = \{(\alpha, h) : \alpha \in G,\ h \in \mathcal{U}_s\}. \tag{2}$$

Операция в этой группе имеет вид $(\alpha,h)\cdot(\beta,f) = (\alpha\beta, h^\beta \oplus f)$, где (α,h), $(\beta,f) \in G\mathcal{U}_s$, а действие на множестве функций $\mathcal{F}_n$ определяется как $f^{(\alpha,h)} = f^\alpha \oplus h$, где $f \in \mathcal{F}_n$ и $(\alpha,h) \in G\mathcal{U}_s$. (Если $G \leqslant \mathrm{GL}(n,2)$, то удобнее рассматривать группы $G\mathcal{U}_s^{(0)}$.) Пусть $(G\mathcal{U}_s)_f$ — группа инерции функции f в группе $G\mathcal{U}_s$

$$(G\mathcal{U}_s)_f = \{(\alpha,h) \in G\mathcal{U}_s : f^{(\alpha,h)} = f\}.$$

Если для всех целых s положить $G_f^{(s)} = \{\alpha \in G : \exists h, (\alpha,h) \in (G\mathcal{U}_s)_f\}$, то, как легко видеть,

$$G_f^{(s)} \cong G_{\{f \oplus \mathcal{U}_s\}} \cong (G\mathcal{U}_s)_f, \tag{3}$$

$$G_f = G_f^{(-1)} \leqslant G_f^{(0)} \leqslant G_f^{(1)} \leqslant \ldots \leqslant G_f^{(\deg f)} = G.$$

Работа выполнена при поддержке гранта Президента РФ НШ № 2358.2003.9.

Асимптотические результаты

Свойство тривиальности групп инерции почти всех функций от n переменных при $n \to \infty$, получившее название эффекта Шеннона, обеспечивает следующую оценку числа N классов эквивалентности функций относительно группы G, $G \leqslant \mathrm{AGL}(n,2)$:

$$\frac{2^{2^n}}{|G|} < N \leqslant \frac{2^{2^n}}{|G|}(1+o(1)).$$

В [6] эффект Шеннона установлен для широкого класса групп, действующих на множестве аргументов функций. В [1] уточнена асимптотика для вероятности этого события. Приведем вид асимптотического разложения числа классов эквивалентности двоичных функций относительно группы $\mathrm{GL}(n,2)$

$$N \sim \frac{2^{2^n}}{|\mathrm{GL}(n,2)|}\left(1+\sum_{t=1}^{\infty} N_t 2^{-2^{n-1}(1-2^{-t})}\right),$$

где

$$N_t = \frac{|\mathrm{GL}(n,2)|}{2^{t(2n-3t)}|\mathrm{GL}(t,2)|\cdot|\mathrm{GL}(n-2t,2)|}.$$

Для группы $\mathrm{AGL}(n,2)$ надо заменить коэффициенты N_t на $N'_t = 2^t N_t$.

В [2] это свойство обобщено на группы $\mathrm{AGL}(n,2)\mathcal{U}_s$ (случай $s=1$ был ранее рассмотрен в [53]). В уточненном виде этот результат можно сформулировать в следующем виде

Теорема 1 ([23]). *Пусть $s = s(n) \leqslant \frac{n}{2}(1-\delta)$, $k = k(n) \geqslant \frac{n}{2}(1+\varepsilon)$, $0 < \delta \leqslant 1$ и $0 < \varepsilon \leqslant 1$. Тогда при $n \to \infty$ почти все формы из факторпространства $\mathcal{U}_k/\mathcal{U}_s$ имеют тривиальную группу инерции в группе* $\mathrm{AGL}(n,2)$.

В этой связи упомянем задачу о нахождении оценок и точных значений величин $n_0(s)$ и $n_1(s)$, которые определяются как минимальное n такое, что существует функция f от n переменных с тривиальной группой инерции $\mathrm{GL}(n,2)_f^{(s)}$ и $\mathrm{AGL}(n,2)_f^{(s)}$ соответственно. Так из предыдущей теоремы сразу получаем

Следствие 1. *При $s \to \infty$ имеет место асимптотическая оценка*

$$n_0(s) \leqslant n_1(s) \leqslant 2s(1+o(1)).$$

В то же время пока лучшей детерминированной оценкой при $s \geqslant 2$ для определенных выше чисел является оценка из [22]:

$$n_0(s) \leqslant n_1(s) \leqslant (s+2)^2 + 1.$$

Точные значения этих величин известны при $s \leqslant 1$ ([13, 56, 22]):

$$n_0(-1) = n_1(-1) = n_0(0) = n_1(0) = 5, \quad n_0(1) = n_1(1) = 6.$$

Отметим также работу [3], в которой введена пороговая функция в эффекте Шеннона, равная минимальному числу значений функции в табличном задании, которое надо исправить для нарушения эффекта Шеннона. Пусть $A_G(f)$ — минимальное число векторов, на которых надо изменить значения функции f, чтобы группа инерции G_f стала нетривиальной.

Теорема 2 ([4]). *Пусть G — одна из групп:* $\mathrm{GL}(n,2)$ *или* $\mathrm{AGL}(n,2)$. *При $n \to \infty$ с вероятностью, стремящейся к единице, выполнено событие*

$$|A_G(f) - 2^{n-3}| \leqslant n2^{\frac{n}{2}-1}.$$

Таблица 1:

n	1	2	3	4	5	6	7
$\mathrm{GL}(n,2)$	4	8	20	92	2 744	950 998 216	$> 2 \cdot 10^{24}$
$\mathrm{AGL}(n,2)$	3	5	10	32	382	15 768 919	$> 10^{22}$
$\mathrm{GL}(n,2)\mathcal{U}_0$	2	4	10	46	1 372	475 499 108	$> 10^{24}$
$\mathrm{AGL}(n,2)\mathcal{U}_0$	2	3	6	18	206	7 888 299	$> 8 \cdot 10^{21}$
$\mathrm{GL}(n,2)\mathcal{U}_1$	1	2	3	14	176	7 880 620	$> 8 \cdot 10^{21}$
$\mathrm{AGL}(n,2)\mathcal{U}_1$	1	2	3	8	48	150 357	$> 6 \cdot 10^{19}$

Задача перечисления функций

Задача *перечисления* (enumeration) состоит в нахождении числа классов эквивалентности функций в данной классификации. В табл. 1 приведены результаты перечисления двоичных функций относительно введенных выше групп при $n \leqslant 7$ и $s \leqslant 1$ ([46, 53, 56, 23]).

Из этой таблицы видно, что линейная и аффинная группы удобны только при $n \leqslant 5$, а при $n \geqslant 6$ надо либо увеличить группу преобразований, действующую на множестве функций, либо — ограничить рассматриваемый класс функций. Группа $\mathrm{AGL}(n,2)$ ($\mathrm{GL}(n,2)$) — максимальна в симметрической (знакопеременной) группе подстановок пространства $V_n(2)$ ($V_n(2) \setminus \{0\}$) (см. [10, 52, 55]), поэтому необходимо ограничить множество функций. При $s = 0$ ситуация существенно не изменяется. Случай $s = 1$ впервые рассмотрен в [53, 54]. Для вычисления числа классов эквивалентности при $s \leqslant 0$ применялась теория перечисления Пойа – Де Брейна, а случай $s = 1$ можно свести к случаю $s = -1$ с увеличением размерности (см. [53, 54, 23]).

В работах [49, 50] предложен общий способ вычисления числа классов эквивалентности пространств $\mathcal{U}_k/\mathcal{U}_s$, $-1 \leqslant s \leqslant n-1$, относительно групп $\mathrm{AGL}(n,2)$ и $\mathrm{GL}(n,2)$ с использованием леммы Бернсайда и рассмотрения их действия на множестве векторов коэффициентов полиномиального задания. Пусть $m_1(n,s,t)$ ($m_0(n,s,t)$) число классов аффинной (линейной) эквивалентности форм из $\mathcal{U}_t/\mathcal{U}_s$ при $-1 \leqslant s \leqslant t \leqslant n-1$ (с учетом симметрии, см. ниже). Значения $m_1(n,s,t)$ из [49] приведены в табл. 2, 3, вычисления значений $m_0(n,s,t)$ в табл. 4, 5 проведены Лакаевым К.С. с использованием метода [49]. Жирным шрифтом отмечены случаи, когда классификация уже известна.

Таблица 2:

$s \setminus t$	0	1	2	3	4	5	6
-1	**2**	**3**	**11**	205	150 357	7 888 299	15 468 919
0		**2**	**7**	120	75 761	3 947 989	
1			**4**	**34**	**2 499**		**150 357**
2				**6**	**34**		

Таблица 3:

$s \setminus t$	0	1	2	3	4	5	6	7
-1	**2**	**3**	**12**	3 486	30 230 045	63 379 147	8 112 499	16 244 999
					341 814	320 777 408	583 888 855	167 506 438
						548	378 066	730 294
0		**2**	**8**	1 890	15 115 039	31 689 573	4 056 249	
					412 866	670 826 699	792 080 063	
						852	701 952	
1			**4**	**179**	118 140 881	247 576 791		
					980	326 613 080		
2				**12**	68 433			

В табл. 6 приведены найденные в [50] числа классов эквивалентности однородных форм от n переменных из $\mathcal{U}_k/\mathcal{U}_{k-1}$ степени k относительно группы $\mathrm{GL}(n,2)$ для $6 \leqslant n \leqslant 11$.

Таблица 4:

$s \setminus t$	0	1	2	3	4	5	6
-1	**2**	**4**	**20**	1 534	7 880 620	475 499 108	950 998 216
0		**2**	**10**	767	3 940 310	237 749 554	
1			**4**	**85**	74 596		
2				**6**	**85**		

Таблица 5:

$s \setminus t$	0	1	2	3	4	5	6	7
-1	**2**	**4**	**22**	161 652	3 868 829	8 112 499	1 038 397	2 076 795
					382 074 516	583 617 583	981 840 994	963 681 989
						352 228	509 577 948	019 155 896
0		**2**	**11**	80 826	1 934 414	4 056 249	519 198 990	
					691 037 258	791 808 791	920 497 254	
						676 114	788 974	
1			**4**	1 596	15 115 005	31 689 573		
					928 948	649 950 738		
						696		
2				**12**	7 384 214			

Таблица 6:

(k, n)	
$(3, 6)$	**6**
$(3, 7)$	**12**
$(3, 8)$	**32**
$(3, 9)$	**349**
$(3, 10)$	3 691 561
$(3, 11)$	60 889 759 853 600
$(4, 8)$	999
$(4, 9)$	121 597 673 132 830
$(4, 10)$	4 490 513 974 418 226 922 710 218 421 015 600
$(4, 11)$	2 847 591 793 161 852 775 156 648 952 439 351 349 039 810 229
	699 431 354 841 358 028
$(5, 10)$	19 749 489 318 110 485 970 697 971 583 208 968 127 316 501 515
$(5, 11)$	15 503 764 406 428 075 099 751 345 714 321 442 971 845 134 712
	815 092 309 403 084 719 632 923 886 700 698 844 470 235 742
	196 625 592 840

Двойственность

Еще в работе [26] было отмечено наличие двойственности факторпространств однородных форм

$$\mathcal{U}_k/\mathcal{U}_{k-1} \cong \mathcal{U}_{n-k}/\mathcal{U}_{n-k-1},$$

где $1 \leqslant k < n$. Соответствие между однородными формами степеней k и $n-k$ устанавливается следующим образом. Одночлену $X = x_{i_1} \dots x_{i_k}$ соответствует одночлен $X^o = x_{j_1} \dots x_{j_{n-k}}$, если $\{j_1, \dots, j_{n-k}\} = \{1, \dots, n\} \backslash \{i_1, \dots, i_k\}$. Однородной форме $f = \sum_s X_s$ теперь поставим в соответствие однородную форму вида $f^o = \sum_s X_s^o$. Для любого линейного преобразования $\beta \in \mathrm{GL}(n,2)$ обозначим через β^* линейное преобразование, матрица которого транспонирована по отношению к обратной матрице линейного преобразования β. Для произвольной подгруппы G группы $\mathrm{GL}(n,2)$ обозначим через G^* группу, состоящую из преобразований, матрицы которых получаются транспонированием и обращением матриц преобразований группы G.

Теорема 3 ([50]). *Пусть $1 \leqslant k < n$. Для любой однородной формы f степени k и любого линейного преобразования $\beta \in \mathrm{GL}(n,2)$ выполнено равенство*

$$(f^\beta)^o \equiv (f^o)^{\beta^*} \pmod{\mathcal{U}_{n-k-1}},$$

где $\beta^* = (\beta^{-1})^t$. В частности,

$$\mathrm{GL}(n,2)_{f^o}^{(n-k-1)} = (\mathrm{GL}(n,2)_f^{(k-1)})^*.$$

Заметим, что хотя для неоднородных форм двойственности для представителей и их групп инерции уже нет, тем не менее, для числа $m(n,s,t)$ классов эквивалентности пространства $\mathcal{U}_t/\mathcal{U}_s$ относительно группы $\mathrm{AGL}(n,2)$ (и, аналогично, для $\mathrm{GL}(n,2)$) имеет место следующее соотношение симметрии.

Теорема 4 ([49]). *Для любого аффинного преобразования $\alpha \in \mathrm{AGL}(n,2)$ и всех $-1 \leqslant s < t \leqslant n$ для числа $N_{(n,t,s)}(\alpha)$ неподвижных элементов пространства $\mathcal{U}_t/\mathcal{U}_s$ относительно α выполняется равенство*

$$N_{(n,t,s)}(\alpha) = N_{(n,n-s-1,n-t-1)}(\alpha),$$

в частности, $m_i(n,s,t) = m_i(n,n-t-1,n-s-1)$, $i = 0,1$.

Задача классификации функций

Классификация функций заключается в нахождении описания орбит (классов эквивалентности) групп преобразований, действующих на множестве функций. Она позволяет построить полную систему инвариантов для данной классификации, с помощью которой легко распознавать принадлежность каждой конкретной функции определенному классу. *Полное перечисление* (complete enumeration) состоит в получении полного списка представителей и нахождении мощностей классов эквивалентности.

Перечислим известные классификации функций от малого числа переменных. Для $n = 4$ линейная и аффинная классификация приведена в [60]. Для $n = 5$ первая классификация была получена для группы $\mathrm{AGL}(5,2)\mathcal{U}_1$ в работе [26]. Затем в [13] была построена классификация для группы $\mathrm{AGL}(5,2)$. Для $n = 6$ в [12] построена аффинная и линейная классификация функций третьей степени с точностью до линейной части (аффинная классификация приведена также в [28]). В [56] (см. также [44]) построена классификация для группы $\mathrm{AGL}(6,2)\mathcal{U}_1$. В [17] найдена аффинная и линейная классификация всех функций от шести переменных с точностью до кубической части. Там же найдена аффинная и линейная классификация функций четвертой степени с точностью до квадратичной части, позднее аффинная классификация была анонсирована в [59]. Для $n = 7$ аффинная классификация функций третьей степени с точностью до квадратичной части получена в [16], а с точностью до линейной части — анонсирована в [59].

Заметим, что задачу классификации функций из подпространства $\mathcal{U}_k$ удобно решать последовательно путем составления сначала классификации элементов факторпространства $\mathcal{U}_k/\mathcal{U}_{k-1}$, затем — из $\mathcal{U}_k/\mathcal{U}_{k-2}$, и т.д. Первым шагом на этом пути является получение классификации однородных форм. Классификация квадратичных форм получена еще в [40]. Классификация однородных кубических

форм для $n = 6$ приведена в [61], при $n = 7$ получена в [16, 50]. В приложении к работе [50] приведен список представителей для $n = 8$. В [18, 19] получено полное описание строения групп инерции для $n = 8$. В [63] найдены порядки групп инерции. В статье [31] объявлено о завершении классификации однородных кубических форм для $n = 9$.

Подробнее о классификации можно посмотреть в [23, 24].

Проверка аффинной эквивалентности преобразований

Представляет интерес изучение классов эквивалентности преобразований пространства $V_n(2)$ при двустороннем действии аффинной (линейной) группой и, в частности, способов проверки равенства

$$F(\alpha(x)) = \beta(G(x)),$$

где $F, G\colon V_n(2) \to V_n(2)$ — преобразования, $\alpha, \beta \in \mathrm{AGL}(n,2)$ ($\mathrm{GL}(n,2)$). Число классов таких преобразований растет с ростом n достаточно быстро ([46, 27], см. табл. 7):

Таблица 7:

Группа $G \times H$ \ n	1	2	3	4	5
$\mathrm{GL}(n,2) \times \mathrm{GL}(n,2)$	2	2	10	52 246	2 631 645 209 645 100 680 644
$\mathrm{AGL}(n,2) \times \mathrm{AGL}(n,2)$	1	1	4	302	2 569 966 041 123 963 092

В работе [27] предложен эффективный алгоритм проверки линейной (аффинной) эквивалентности двух преобразований пространства $V_n(2)$. В основе алгоритма лежит процедура последовательного подбора значений преобразования на линейно независимых векторах, что позволяет легко находить матрицы преобразований. Там же рассматривалась постановка задачи о декомпозиции преобразования в SP-сеть из преобразований меньшей размерности с линейными перемешивающими преобразованиями, и оценивалось минимальное число уровней в такой декомпозиции для произвольного случайного узла.

Пространство существенных переменных

Пусть $V = V_n(2)$ и V^* — сопряженное пространство, состоящее из линейных функций на V. Если $x \in V$ и $a^* \in V^*$, то значение линейной функции a^* на векторе x удобно обозначать в виде (x, a^*). Каждую функцию $f \in \mathcal{F}_n$ можно интерпретировать как функцию на векторном пространстве V, табличное задание которой получается при фиксации единичного базиса $e^1, e^2, \ldots, e^n$. Обозначая через $x_e = (x_1, x_2, \ldots, x_n)$ вектор координат элемента $x \sum\limits_{i=1}^{n} x_i e^i \in V$, имеем

$$f(x) = f_e(x_e) = f_e(x_1, x_2, \ldots, x_n),$$

где $x_i = (x, e^{*i})$, $i = \overline{1,n}$.

Пусть $0 \leqslant t \leqslant n-1$, $1 \leqslant k \leqslant n$. Будем говорить, что переменные $x_{k+1}, \ldots, x_n$ функции $f(x_1, \ldots, x_n)$ являются *несущественными по модулю* $\mathcal{U}_t$, если найдется функция $h_e(x_1, \ldots, x_k)$ такая, что $f \oplus h \in \mathcal{U}_t$. Нетрудно видеть, что переменное x_n является несущественным для функции f по модулю $\mathcal{U}_t$, если и только если $D_{e_n} f \in \mathcal{U}_{t-1}$ или, что то же самое, если

$$\binom{x}{x \oplus e_n} \in (\mathrm{H}_n)_f^{(t-1)},$$

где $e_n = (0, \ldots, 0, 1)$. Пусть теперь функция f зависит по модулю $\mathcal{U}_t$ существенно лишь от k переменных, $1 \leqslant k < n$, то есть

$$f(x) = f_e(x_e) \equiv h_e(x_1, \ldots, x_k) \pmod{\mathcal{U}_t},$$

причем k — минимальное с этим свойством по всем линейным заменам переменных (или, что то же самое, по всем базисам). Тогда с этой функцией однозначно связаны два подпространства:

$V_1^* =< e^{*1}, \ldots, e^{*k} >\subseteq V^*$ — *подпространство существенных переменных по модулю* $\mathcal{U}_t$ (в том смысле, что каждый ненулевой вектор e^* этого подпространства можно дополнить до базиса так, что соответствующее переменное $x_1 = (x, e^*)$ будет существенным) и двойственное ему подпространство $(V_1^*)^\perp = \{x : (x, e^*) = 0, e^* \in V_1^*\} \subseteq V$ векторов, сдвиги по которым лежат в группе инерции $(\mathrm{H}_n)_f^{(t-1)}$. Поэтому в этом случае можно использовать запись $f = f(V_1^*)$.

Заметим, что множество векторов

$$\left\{a \in V_n(2) : \begin{pmatrix} x \\ x \oplus a \end{pmatrix} \in (\mathrm{H}_n)_f^{(0)} \setminus (\mathrm{H}_n)_f\right\}$$

называют *линейной структурой* функции f. Функции, обладающие линейной структурой, при некоторой линейной замене имеют переменные, которые являются существенными по модулю $\mathcal{U}_0$, но не являются существенными по модулю $\mathcal{U}_1$, то есть являются независимыми линейными слагаемыми.

Линейная декомпозиция

При построении линейной классификации двоичных функций наиболее сложной задачей является описание их групп инерции. Вместе с тем, в случаях, когда при некоторой линейной замене переменных функция допускает бесповторную декомпозицию, задача описания групп инерции такой функции значительно упрощается. Рассмотрим несколько простейших случаев представления функций в виде бесповторной суперпозиции. Более подробно эти вопросы рассмотрены в [23].

Теорема 5. *Пусть* $t \geqslant 0$ *следующие условия равносильны:*

a) $f = f(x_1, \ldots, x_n)$ *имеет тривиальную группу инерции* $(\mathrm{H}_n^{(t-1)})_f$;

b) $\deg D_a f \geqslant t$ *для всех векторов* $0 \neq a \in V_n(2)$;

c) *пространство существенных переменных функции* f *по модулю* $\mathcal{U}_t$ *совпадает со всем пространством* V^*.

Будем говорить, что функция $f = f(V^*)$ *линейно разложима в бесповторную сумму по модулю* $\mathcal{U}_s$, если найдется нетривиальное разложение пространства V^* в прямую сумму подпространств $V^* = V_1^* + V_2^*$, при котором функция имеет вид

$$f \equiv f_1(V_1^*) \oplus f_2(V_2^*) \pmod{\mathcal{U}_s}.$$

Если степень нелинейности одного из слагаемых равна единице, то изучение такой функции сводится к случаю $n-1$ переменного.

Теорема 6. *Если в базисе* $e^1, \ldots, e^n$ *имеет место равенство* $f_e(x_1, \ldots, x_n) = h_e(x_1, \ldots, x_{n-1}) \oplus x_n$ *и* $|(\mathrm{H}_n)_f| = 1$, *то* $|(\mathrm{H}_n)_h^{(0)}| = 1$ *и справедливы изоморфизмы:*

$$\begin{aligned} \mathrm{GL}(n,2)_f &\cong \mathrm{GL}(n-1,2)_h^{(1)}; \\ \mathrm{AGL}(n,2)_f &\cong \mathrm{AGL}(n-1,2)_h^{(1)}; \\ \mathrm{AGL}(n,2)_f^{(0)} &\cong \mathrm{AGL}(n-1,2)_h^{(1)} \times \mathrm{H}_1\,. \end{aligned}$$

Заметим, что для слагаемых второй степени ни о каком сведении к группам инерции слагаемых в принципе не может быть и речи, так как полученные функции могут иметь неприводимые группы инерции, в качестве которых выступают классические линейные группы. В то же время, для слагаемых степени три и выше, как правило, такое сведение уже имеет место, но при ограничениях на число существенных переменных по модулю $s \geqslant 2$.

Лемма 1. *Пусть имеется два разложения пространства* V^* *в прямую сумму ненулевых подпространств* $V^* = V_1^* + V_2^* = U_1^* + U_2^*$. *Если при* $s \geqslant 2$ *функция* $f = f(x_1, \ldots, x_n) = f(V^*)$ *имеет тривиальную группу инерции* $(\mathrm{H}_n^{(s-1)})_f$ *и выполняется сравнение*

$$f \equiv f_1(V_1^*) \oplus f_2(V_2^*) \equiv h_1(U_1^*) \oplus h_2(U_2^*) \pmod{\mathcal{U}_s},$$

то функция f допускает разложение

$$f \equiv d_1(W_{11}^*) \oplus d_2(W_{12}^*) \oplus d_3(W_{21}^*) \oplus d_4(W_{22}^*) \pmod{\mathcal{U}_s},$$

где $W_{ij}^* = V_i^* \bigcap U_j^*$, $i,j = 1,2$.

Теорема 7 ([21]). *Если при $s \geqslant 2$ функция $f = f(x_1,\ldots,x_n)$ имеет тривиальную группу инерции $(\mathrm{H}_n)_f^{(s-1)}$, и линейно разложима в бесповторную сумму по модулю $\mathcal{U}_s$, то для этой функции найдется однозначно определенное линейное разложение по модулю $\mathcal{U}_s$ в бесповторную сумму линейно неразложимых (в бесповторную сумму) слагаемых в том смысле, что любое другое такое разложение соответствует тому же самому разложению пространства в сумму подпространств, а соответствующие функции сравнимы по модулю $\mathcal{U}_s$.*

Следствие 2. *Если множество линейно неразложимых в бесповторную сумму по модулю $\mathcal{U}_s$ функций $\{f_1,\ldots,f_m\}$, у каждой из которых размерность пространства существенных переменных по модулю $\mathcal{U}_s$, $s \geqslant 2$, совпадает с числом переменных, разбивается на классы аффинной эквивалентности по модулю $\mathcal{U}_s$: $\{f_{\mu_1},\ldots,f_{\mu_p}\},\ldots,\{f_{\nu_1},\ldots,f_{\nu_q}\}$, то для группы инерции бесповторной суммы этих функций справедлив изоморфизм*

$$\mathrm{AGL}(n,2)^{(s)}_{f_1\oplus\cdots\oplus f_m} \cong [\mathrm{AGL}(n_{\mu_1},2)^{(s)}_{f_{\mu_1}}]\mathrm{S}_p \times \cdots \times [\mathrm{AGL}(n_{\nu_1},2)^{(s)}_{f_{\nu_1}}]\mathrm{S}_q.$$

Аналогичное описание справедливо для группы $\mathrm{GL}(n,2)$.

Рассмотрим теперь случай разложения в произведение. Пусть $-1 \leqslant s \leqslant n-1$. Будем говорить, что функция f *имеет линейные (инверсные линейные) сомножители по модулю* $\mathcal{U}_s$, если найдутся такие функция $l = (x,a^*)$ ($l = (x,a^*)\oplus 1$), $x \in V_n(2)$, $a^* \in V_n^*$, отличная от константы, и функция h, что $f \overset{s}{\equiv} l\cdot h$. Будем также для удобства говорить, что функция f *имеет аффинные сомножители модулю* $\mathcal{U}_s$, если она имеет линейные или инверсные линейные сомножители по модулю $\mathcal{U}_s$. Если функция f имеет $k \geqslant 1$, аффинных сомножителей по модулю $\mathcal{U}_s$ $l_i(x) = (x,a^{*i}) \oplus b_i$, где $a^{*i} \in V_n^*$, $x \in V_n(2)$, $b_i \in \{0,1\}$, $i \in \overline{1,k}$, таких, что векторы a^{*i}, $i \in \overline{1,k}$, линейно независимы, но не имеет $k+1$ таких сомножителей, то будем говорить, что она *имеет ровно k аффинных сомножителей по модулю* $\mathcal{U}_s$. Заметим, что в этом случае подпространство $< a^{*1}, a^{*2},\ldots,a^{*k} >$ определяется по функции однозначно. Следующие утверждения очевидны ([19]).

Лемма 2. *Пусть аффинная функция $l(x) = (x,a^*) \oplus b$ отлична от константы. Следующие условия равносильны:*

a) *функция f имеет аффинный сомножитель l по модулю $\mathcal{U}_s$;*

b) $l \cdot f \equiv f \pmod{\mathcal{U}_s}$;

c) $\overline{l} \cdot f \in \mathcal{U}_{s+1}$.

Теорема 8. *Пусть $k \geqslant 1$ и $s \leqslant \deg f - 1$. Тогда следующие условия эквивалентны:*

a) *функция f имеет ровно k аффинных сомножителей по модулю $\mathcal{U}_s$;*

b) *при некоторой линейной замене переменных с матрицей A функция f удовлетворяет условию*

$$f(xA) \equiv x_1^{b_1} \cdot \ldots \cdot x_k^{b_k} h(x_{k+1},\ldots,x_n) \pmod{\mathcal{U}_s},$$

где $b_i \in \{0,1\}$, $i \in \overline{1,k}$ и функция h не имеет линейных сомножителей по модулю $\mathcal{U}_{s-k}$.

Пусть $k \in \{0,\ldots,n\}$. Обозначим через $\mathcal{F}_n(k)$ множество всех двоичных функций, имеющих ровно k аффинных сомножителей. Функцию $f \equiv 0$ не включаем ни в одно из множеств $\mathcal{F}_n(k)$, $k = \overline{1,n}$. Легко видеть, что выполняется равенство

$$\mathcal{F}_n = \bigcup_{k=0}^{n} \mathcal{F}_n(k) \cup \{0\}.$$

При $n \geqslant 1$ числа

$$\begin{bmatrix} n \\ k \end{bmatrix}_2 = \begin{cases} \prod\limits_{i=0}^{k-1} \frac{2^n-2^i}{2^k-2^i}, & \text{если } k = \overline{1,n}, \\ 1, & \text{если } k = 0, \end{cases}$$

называются коэффициентами Гаусса.

Теорема 9 ([23]). *При $1 \leqslant k \leqslant n$ справедливы равенства*

$$|\mathcal{F}_n(k)| = 2^k \cdot \begin{bmatrix} n \\ k \end{bmatrix}_2 \cdot |\mathcal{F}_{n-k}(0)|,$$

$$|\mathcal{F}_n(0)| = \sum_{k=0}^{n} (-1)^k 2^{\frac{k(k+1)}{2}} \begin{bmatrix} n \\ k \end{bmatrix}_2 (2^{2^{n-k}} - 1) \cdot 2^k.$$

Теорема 10 ([20]). *Если функция $f = f(x_1, \ldots, x_n)$ имеет тривиальную группу инерции $(\mathrm{H}_n)_f$, не имеет аффинных сомножителей и линейно разложима в бесповторное произведение, то для этой функции найдется однозначно определенное линейное разложение в бесповторное произведение линейно неразложимых (в бесповторное произведение) сомножителей в том смысле, что любое другое такое разложение соответствует тому же самому разложению пространства в сумму подпространств, а соответствующие функции на подпространствах совпадают.*

Приведем еще один результат ([21]).

Теорема 11. *Если функция $f = f(x_1, \ldots, x_n)$ имеет тривиальную группу инерции $(\mathrm{H}_n)_f$ и линейно разложима в бесповторное произведение, то она не может быть линейно разложимой в бесповторную сумму.*

Следствие 3. *Если функция $f = f(x_1, \ldots, x_n)$ имеет тривиальную группу инерции $(\mathrm{H}_n)_f$, то она может быть линейно бесповторно разложимой только относительно одной из операций $* \in \{\&, \vee, \oplus\}$.*

Разложение Фурье

Для двоичной функции — это разложение вида

$$f(x) = \frac{1}{2^n} \sum_{a^* \in V_n^*(2)} \widehat{f}(a^*) \cdot (-1)^{(x,a^*)}. \tag{4}$$

$$\widehat{f}(a^*) = \sum_{x \in V_n(2)} f(x) \cdot (-1)^{(x,a^*)}. \tag{5}$$

Набор коэффициентов $\widehat{f}(a^*)$, $a^* \in V_n^*(2)$, задает функцию $\widehat{f}$ на сопряженном пространстве. При этом для нее имеет место следующее очевидное

Утверждение 1. *Если f — инвариант группы G, $G \subseteq \mathrm{GL}(n,2)$, то функция $\widehat{f}$ будет инвариантом группы*

$$G^* = \{(\beta^{-1})^t : \beta \in G\} \subseteq \mathrm{GL}(n,2),$$

где t — транспонирование. В частности, группа G будет совпадать с группой инерции функции f в группе $\mathrm{GL}(n,2)$, в том и только в том случае, когда группа G^* будет совпадать с группой инерции функции $\widehat{f}$ в группе $\mathrm{GL}(n,2)$.

Для k-значной функции f под разложением Фурье понимается разложение по характерам абелевой группы, заданной на множестве аргументов. Заметим, что для k-значной функции f удобнее рассматривать разложения Фурье не самой функции f, а функции $\chi(f)$, где $\chi(x)$ — некоторый характер абелевой группы, определенной на области значений функции. В частности, для двоичной функции рассматривают характер $\chi_f = \chi(f) = (-1)^f$, а само разложение называют разложением Уолша (коэффициенты разложения Уолша обозначают $W_f(a^*) = \widehat{\chi_f}(a^*)$, $a^* \in V_n^*(2)$).

Приближения/аппроксимации аффинными функциями

Для двоичной функции нелинейность определяется как минимальное расстояние по Хеммингу до класса аффинных функций

$$\mathrm{NL}_f = \min_{g \in \mathcal{U}_1} d_H(f,g) = d_H(f, \mathcal{U}_1).$$

Для отображений $F : V_n(2) \to V_m(2)$

$$\mathrm{NL}_F = \min_{0 \neq u \in V_m^*(2)} \left(\min_{g \in \mathcal{U}_1} d_H((F, u^*), g) \right).$$

Нелинейность выражается через коэффициенты преобразования Уолша

$$\mathrm{NL}_f = 2^{n-1} - \max_{a \in V_n^*(2)} |W_f(a^*)|.$$

$$\mathrm{NL}_F = 2^{n-1} - \max_{0 \neq u \in V_m^*(2)} \max_{a^* \in V_n^*(2)} |W_{(F,u^*)}(a^*)|.$$

Оценки величины NL_f:

— из равенства Парсеваля

$$\mathrm{NL}_F \leqslant 2^{n-1} - 2^{\frac{n}{2}-1};$$

— неравенство Сидельникова–Chabaud–Vaudenay

$$\mathrm{NL}_F \leqslant 2^{n-1} - \frac{1}{2}\sqrt{3 \times 2^n - 2 - 2\frac{(2^n-1)(2^{n-1}-1)}{2^m-1}}. \tag{6}$$

Это неравенство улучшает предыдущую оценку при $m \geqslant n$ и достигается только когда $n = m$ нечетно. Функция F называются *бент* (Bent) функцией, если для нее выполняется равенство $\mathrm{NL}_F = 2^{n-1} - 2^{\frac{n}{2}-1}$. Функция F называются *почти бент* (Almost Bent, AB) функцией ([36]), если для нее достигается равенство в неравенстве (6): $\mathrm{NL}_F = 2^{n-1} - 2^{\frac{n-1}{2}}$.

Приведем для сравнения результат о эффективности приближения произвольной двоичной функции $f \in \mathcal{F}_n$ квадратичными формами из $\mathcal{U}_2$. Пусть

$$\varrho_n(f) = \min_{g \in \mathcal{U}_2} d_H(f, g).$$

Теорема 12 ([11]). $\forall x > 0$

$$\lim_{n \to \infty} P\left(\frac{\varrho_n(f) - a_n}{b_n} \leqslant x \right) = 1 - e^{-e^x},$$

где

$$a_n = 2^{n-1} - 2^{n/2-1}\sqrt{\ln 2}\left\{ n - \frac{1}{2} - \frac{\ln n}{n \ln 2} - \frac{4\ln\ln 2 + 4\ln\pi - 3\ln 2}{8n\ln 2} \right\},$$

$$b_n = \frac{2^{n/2-1}}{n\sqrt{\ln 2}}.$$

Для среднего значения случайной величины

$$d_H(f, \mathcal{U}_1) = \min_{g \in \mathcal{U}_1} d_H(f, g).$$

справедлива оценка ([62])

$$\mathrm{M}\, d_H(f, \mathcal{U}_1) = 2^{n-1} - 2^{n/2-1}\sqrt{2\ln|\mathcal{U}_1|}(1 + o(1)).$$

Для квадратичных форм из теоремы получаем

$$\mathrm{M}\, d_H(f, \mathcal{U}_2) = 2^{n-1} - 2^{n/2-1}\sqrt{2\ln|\mathcal{U}_2|}(1 + o(1)).$$

«Близость» к линейным функциям можно характеризовать и другими способами, например, проверяя, как часто для функции f выполняется тождество линейности: $f(x) + f(y) = f(x + y)$. Степень выполнения этого тождества можно характеризовать *вероятностями аддитивности* (термин предложен М. М. Глуховым)

$$p_k(f) = P\left(\sum_{i=1}^{k} f(x^{(i)}) = f(\sum_{i=1}^{k} x^{(i)}) \right), \quad k = 2, 3, \ldots.$$

Легко проверить, что

$$p_k(f) = \frac{1}{2}\left(1 - \frac{1}{2^{(k+1)n}} \sum_{a^* \in V_n^*(2)} W_f^{k+1}(a^*)\right).$$

Чем ближе к $1/2$ значения этих вероятностей, тем более нелинейная функция.

Для отображений $F : V_n(2) \to V_m(2)$ имеем

$$p_k(F) = P\left(\sum_{i=1}^{k} F(x^{(i)}) = F(\sum_{i=1}^{k} x^{(i)})\right) =$$

$$= \frac{1}{2^{(k+1)n+m}} \sum_{b^* \in V_m^*(2)} \sum_{a^* \in V_n^*(2)} W_{(F,b^*)}^{k+1}(a^*),$$

так как

$$\sum_{b^* \in V_m^*(2)} \sum_{a^* \in V_n^*(2)} W_{(F,b^*)}^{k+1}(a^*) =$$

$$= \sum_{b^* \in V_m^*(2)} \sum_{a^* \in V_n^*(2)} \left(\sum_{x \in V_n(2)} (-1)^{(F(x),b^*)+(x,a^*)}\right)^{k+1} =$$

$$= \sum_{x^{(1)},\dots,x^{(k+1)} \in V_n(2)} \left(\sum_{b^* \in V_m^*(2)} (-1)^{(\sum_{i=1}^{k+1} F(x^{(i)}),b^*)}\right) \left(\sum_{a \in V_n^*(2)} (-1)^{(\sum_{i=1}^{k+1} x^{(i)},a^*)}\right) =$$

$$= 2^{n+m} \left|\left\{(x^{(1)},\dots,x^{(k)}) : \sum_{i=1}^{k} f(x^{(i)}) = f(\sum_{i=1}^{k} x^{(i)})\right\}\right|.$$

Понятие нелинейности и вероятности 3-аддитивности смыкаются друг с другом. Так при доказательстве неравенства (6) используется следующий прием:

$$\max_{0 \neq b^* \in V_m^*(2)} \max_{a \in V_n^*(2)} |W_{(F,b^*)}^2(a^*)| \geqslant \frac{\sum\limits_{0 \neq b^* \in V_m^*(2)} \sum\limits_{a^* \in V_n^*(2)} W_{(F,b^*)}^4(a^*)}{\sum\limits_{0 \neq b^* \in V_m^*(2)} \sum\limits_{a^* \in V_n^*(2)} W_{(F,b^*)}^2(a^*)}.$$

В силу равенства Парсеваля знаменатель является константой, не зависящей от функции F.

Используя равенство для вероятности 3-аддитивности, получаем

$$\sum_{b^* \in V_m^*(2)} \sum_{a^* \in V_n^*(2)} W_{(F,b^*)}^4(a^*) =$$

$$= 2^{n+m} |\{(x,y,z) : f(x) + f(y) + f(z) = f(x+u+z)\}| \geqslant$$

$$\geqslant |\{(x,y,z) : x = y \vee x = z \vee y = z\}|.$$

Заметим, что если два из трех векторов совпадают, то равенство из условия 3-аддитивности выполнено для любых функций. Поэтому представляет интерес изучение множества функций, для которого последнее неравенство превращается в равенство. Это значит, что для таких функций условие 3-аддитивности выполнено только для тривиальных наборов векторов. Функция называется *почти совершенно нелинейной* (Almost Perfect Nonlinear, APN), если для нее выполнено условие

$$F(x) + F(y) + F(z) + F(x+y+z) = 0 \iff (x = y \vee x = z \vee y = z).$$

Класс AB функций содержится в классе APN функций. Следуя [36], приведем примеры и сравнительные свойства классов бент, AB и APN функций (см. табл. 8).

Известные классы бент функций $F\colon V_n \to V_m$, $n = 2k$, $x, y \in \mathrm{GF}(2^k)$:

(i) $m \leqslant k$, $F(x,y) = L(x \times \pi(y) + g(y))$, π — подстановка, g — произвольная функция, $L : V_k \to V_m$ — линейное отображение (Nyberg);

Таблица 8:

	Бент функции	АВ функции	APN функции
n	четное	нечетное	? нечетное
m	$m \leqslant n/2$	$m = n$	$m = n$
$\forall a \neq 0\ D_a f(x) = b$	имеет 1 решение	имеет 1 решение	$\leqslant 2$ решений
$\forall u^* \neq 0\ (F, u^*)$	бент	платовидная	платовидная

(ii) $F(x, y) = G(\frac{x}{y})$ ($\frac{x}{y} = 0$, если $y = 0$), G — уравновешенная $(n/2, m)$-функция.

Одним из наиболее удобных способов построения АВ и APN функций является рассмотрение класса степенных функций вида $\tau(x) = x^d$ в поле $\mathrm{GF}(2^n)$, позволяющих обеспечить высокую нелинейность координатных функций и их линейных комбинаций. Действительно, для систем функций на основе подстановки $\tau(x) = x^d$ на пространстве $V_n(2)$, где $(d, 2^n - 1) = 1$, линейные комбинации координатных функции линейно эквивалентны между собой. Это легко следует из того факта, что для каждой линейной комбинации $(\tau(x), u^*)$, $u^* \in V_n^*(2)$, координатных функций, линейную функцию $l(x) = (x, u^*)$ можно выразить через функцию след $\mathrm{tr} : \mathrm{GF}(2^n) \to \mathrm{GF}(2)$ в виде $l(x) = \mathrm{tr}(ax)$ при некотором $a \in \mathrm{GF}(2^n)$. Отсюда

$$(\tau(x), u^*) = l(\tau(x)) = \mathrm{tr}(a\tau(x)) = \mathrm{tr}(ax^d) = \mathrm{tr}((a^{1/d}x)^d),$$

где $a^{1/d}x = Ax$ — линейное преобразование над полем $\mathrm{GF}(2)$. Поэтому все функции вида $l(\tau(x))$, где l — линейная функция, имеют вид $\mathrm{tr}(\tau(Ax))$ при некотором линейном преобразовании $A \in \mathrm{GL}(n, 2)$. Из этого представления вытекает, что линейные комбинации координатных функций линейно эквивалентны, и поэтому имеют одинаковую нелинейность для всех линейных комбинаций.

Известные классы АВ функций $F\colon V_n \to V_n, F(x) = x^d$:

(i) $d = 2^k + 1$, $\gcd(n, k) = 1$ (Gold, 1968);

(ii) $d = 2^{2k} - 2^k + 1$, $\gcd(n, k) = 1$ (Kasami, 1971) ;

(iii) $d = 2^k + 3$, $n = 2k + 1$ (Canteaut, Charpin, Dobbertin, 2000) ;

(iv) $d = 2^k + 2^{k/2} - 1$, если k четное, $d = 2^k + 2^{(3k+1)/2} - 1$, если k нечетное, где $n = 2k + 1$ (Hollman, Xiang, 2001).

Известные классы APN функций $F : V_n \to V_n, F(x) = x^d$:

(i) $d = 2^k + 1$, $\gcd(n, k) = 1$ (Gold, 1968);

(ii) $d = 2^{2k} - 2^k + 1$, $\gcd(n, k) = 1$ (Kasami, 1971 ; Janwa, Wilson, 1993) ;

(iii) $d = 2^k + 3$, $n = 2k + 1$ (Dobbertin, 1999);

(iv) $d = 2^k + 2^{k/2} - 1$, если k четное; $d = 2^k + 2^{(3k+1)/2} - 1$, если k нечетное, где $n = 2k + 1$ (Dobbertin, 1999);

(v) $d = 2^{2k} - 1$, $n = 2k + 1$ (Beth, Ding, 1994);

(vi) $d = 2^{4k} + 2^{3k} + 2^{2k} + 2^k - 1$, $n = 5k$ (Dobbertin, 2000).

Приведенные выше степенные APN функции не являются подстановками при четных n, и вопрос о существовании APN подстановок остается открытым.

В [36] приведены примеры APN и АВ функций, не эквивалентных степенным функциям.

Степень нелинейности

Приведем еще один параметр, характеризующий «нелинейность» функции. *Степень нелинейности* p^m-значной функции F (обозначается $\mathrm{dl}\, F$) называется минимальное натуральное число m такое, что

$$D_{a_1} \dots D_{a_{m+1}} F(x) = 0$$

при всех $a_1, \dots, a_{m+1} \in \Omega$. С учетом свойства 2 получаем, что $\mathrm{dl}\, F$ — это максимальное m такое, что при некоторых $a_1, \dots, a_m \in \Omega$

$$D_{a_1} \dots D_{a_m} F(0) \neq 0.$$

Для степени нелинейности выполняются следующие очевидные свойства.

1. $\mathrm{dl}\, D_a F \leqslant \mathrm{dl}\, F - 1$ *при всех* $0 \neq a \in \Omega$, *причем всегда найдется такой элемент* $0 \neq a \in \Omega$, *что* $\mathrm{dl}\, D_a F = \mathrm{dl}\, F - 1$.
2. $\mathrm{dl}(F_1 + F_2) \leqslant \max\{\mathrm{dl}\, F_1, \mathrm{dl}\, F_2\}$.
3. $\mathrm{dl}(F_1 \cdot F_2) \leqslant \mathrm{dl}\, F_1 + \mathrm{dl}\, F_2$. *Если функции* F_1 *и* F_2 *зависят от непересекающихся множеств переменных, то* $\mathrm{dl}(F_1 \cdot F_2) = \mathrm{dl}\, F_1 + \mathrm{dl}\, F_2$.

Это определение интересно тем, что в нем не участвует операция умножения. Помимо «аддитивного» определения степени нелинейности имеется «мультипликативное» определение: степенью нелинейности многочлена p^m-значной функции F называется максимальное значение величины

$$\|b_1\| + \dots + \|b_n\|$$

для всех входящих в него одночленов $x_1^{b_1} \cdot \dots \cdot x_n^{b_n}$, где $\|b_i\|$ — сумма цифр в p-ичной записи числа $b_i \in \{0, 1, \dots, p^m - 1\}$, $1 \leqslant i \leqslant n$. Для функций, определенных над конечным полем $\mathrm{GF}(p^m)$, эти определения равносильны ([23]), причем степень нелинейности совпадает с максимальной степенью многочленов координатных функций в p-ичном представлении функции F системой из m функций над полем $\mathrm{GF}(p)$. Заметим, что для двоичных функций степень нелинейности и алгебраическая степень многочленов совпадают.

Носители, покрывающие последовательности

Носителем двоичной функции f называется множество векторов, на которых функция принимает ненулевые значения $supp(f) = \{a \in V_n(2) : f(a) \neq 0\}$. Носитель преобразования Уолша $\widehat{\chi_f}$ функции f будем будем обозначать как

$$S_f = supp(\widehat{\chi_f}) = \{b^* \in V_n^*(2) : \widehat{\chi_f}(b^*) \neq 0\}.$$

Перечислим простейшие свойства этих множеств.

Если $g(x) = f(x) \oplus (x, a^*)$, то $S_f = a^* + S_g$.

Если $g(x) = f(xA)$, $A \in \mathrm{GL}(n, 2)$ то $S_g = A^*(S_f)$, $A^* = A^{-t}$.

Если $h(x, y) = f(x) \oplus g(y)$, то $S_h = S_f \times S_g$.

Если f — нечетная функция, то $S_f = V_n(2)$.

Если $f(x) = (x, a^*) \oplus b$ — аффинная, то $S_f = \{a^*\}$.

Если f — квадратичная форма, то S_f — многообразие четной размерности.

Если $f(x) = g(xA) \oplus (x, a^*) \oplus b$ частичная бент функция (то есть g — бент функция от $2t$ переменных), то S_f — многообразие размерности $2t$.

В работе [35] строятся примеры уравновешенных и бент функций, у которых носителями совпадают со всем пространством, либо являются многообразиями, в том числе нечетной размерности. Для этого устанавливается связь между носителями и покрывающими последовательностями, введенными в [34].

Покрывающая последовательность двоичной функции f — это любая числовая функция $\lambda : V_n(2) \to \mathbf{C}$ такая, что

$$\sum_{a \in V_n(2)} \lambda_a D_a f(x) = \rho$$

— константная функция.

Теорема 13 ([34]). *f уравновешенна тогда и только тогда, когда она имеет нетривиальную покрывающую последовательность. f имеет покрывающую последовательность λ в том и только в том случае, когда $\widehat{\lambda}$ принимает постоянные значения на носителе S_f.*

Для уравновешенной функции можно подобрать покрывающую последовательность, носителем которой является функция $\lambda = 1$. Поэтому представляет интерес случай, когда λ принимает значения 0, 1, то есть функция λ является носителем для некоторого подмножества. Рассмотрим случай, когда носитель $S = supp(\lambda)$ покрывающей последовательности λ содержится в подпространстве $E \leqslant V_n(2)$. В этом случае числовая функция

$$\sum_{a \in E} \lambda_a D_a f(x)$$

будет константой ρ в том и только в том случае, когда ее ограничение на любой смежный класс $a+E$, $a \in V_n(2)$, будет константой ρ. Отсюда следует

Утверждение 2 ([35]). *Пусть E — подпространство в $V_n(2)$. Тогда функция f имеет нетривиальную покрывающую последовательность λ с носителем $S \subset E$ в том и только в том случае, когда ограничение функции f на каждое многообразие $a+E$ (рассматриваемое как функция на E), имеет ту же покрывающую последовательность.*

Заметим, что если $D_a f(x) = 1$ при некотором $a \in V_n(2)$, то функция f линейно эквивалентна функции, у которой одно из переменных является независимым слагаемым. Поэтому далее такой случай исключаем.

Утверждение 3 ([35]). *Пусть E — подпространство в $V_n(2)$ и $a+E$ — смежный класс. Пусть функция f не имеет единичных производных. Тогда f имеет в качестве покрывающей последовательности индикатор множества $a+E$ в том и только в том случае, когда $S_f \bigcap E^{\perp} = 0$, где*

$$E^{\perp} = \{b^* \in V_n^*(2) : (a, b^*) = 0, \forall a \in E\}.$$

Это эквивалентно тому, что ограничение f на любой смежный класс $a+E$, $a \in V_n(2)$, будет уравновешенной функцией. Более общо, любая функция λ с условием $\lambda_{a+u} = \lambda_u$ для всех $a, u \in V_n(2)$ будет также покрывающей последовательностью.

Рассмотрим строение носителя спектра платовидных функций. Если коэффициенты Уолша принимают значения $0, \pm 2^{n-h}$, то в силу равенства Парсеваля мощность спектра равна 4^h, при этом нетрудно показать, что $\deg f \leqslant h+1$. Под аффинным рангом множества S понимается минимальная размерность подпространства E такого, что $S \subseteq a+E$ при некотором $a \in V_n(2)$ или, что то же самое, размерность пространства существенных переменных функции f по модулю $\mathcal{U}_1$. Из мощностных соображений всегда $k \geqslant 2h$.

Теорема 14 ([15]). *Если $|S_f| = 4^h$, и k — аффинный ранг спектра платовидной функции, то $2h \leqslant k \leqslant 2^{2h-1} - 2^h + h$.*

Теорема 15 ([15]). *Для любого натурального k в интервале $2h \leqslant k \leqslant 2^{h+1} - 2$ существует платовидная функция с носителем спектра мощности 4^h и аффинным рангом k.*

Теорема 16 ([15]). *Если $h = 2$, то аффинный ранг спектра платовидной функции может принимать только значения 4, 5 или 6.*

Алгебраическая иммунность и следствия малой степени

Функция g называется аннулирующей для f, если $f \cdot g = 0$. Нетрудно показать, что множество $\mathrm{Ann}(f)$ всех аннулирующих функций для f образует главный идеал, порожденный функцией $f \oplus 1$

$$\mathrm{Ann}(f) = < f \oplus 1 >_{\mathcal{F}_n} = \{(f \oplus 1) \cdot g : g \in \mathcal{F}_n\}.$$

Алгебраическая иммунность ($\mathrm{AI}(f)$) функции f определяется как минимальная степень d ненулевой аннулирующей функции для f или $f \oplus 1$, то есть

$$\left(< f >_{\mathcal{F}_n} \bigcup < f \oplus 1 >_{\mathcal{F}_n} \right) \bigcap \mathcal{U}_d \neq (0). \tag{7}$$

Данное определение интересно в связи с тем, что, если $g \in \mathrm{Ann}(f \oplus a)$, $a = 0, 1$, то уравнение $g(x) = 0$ будет следствием для уравнения $f(x) = a \oplus 1$.

Приведем другие эквивалентные определения (см. [58]):

Утверждение 4. *Следующие утверждения эквивалентны:*

(i) $\mathrm{AI}(f) = d$;

(ii) d — *минимальная степень функции* $(f \oplus a) \cdot g \neq 0$ *по всем функциям* $g \in \mathcal{F}_n$ *и* $a \in \{0, 1\}$;

(iii) d — *минимальное число, для которого существуют функции* $g, h \in \mathcal{U}_d$, *не равные одновременно нулевой функции и такие, что* $f \cdot g = h$.

Доказательство. Равносильность (i) $\Leftrightarrow$ (ii) непосредственно вытекает из равенства (7). (i) $\Rightarrow$ (iii): Если $\mathrm{AI}(f) = d$, $0 \neq g \in \mathcal{U}_d$ и $fg = 0$, то можно положить $h = 0$. Если $(f \oplus 1)g = 0$, то $h = d$. (iii) $\Rightarrow$ (i): Если при некоторых функциях $g, h \in \mathcal{U}_d$ выполнено $f \cdot g = h$, то при $h \neq 0$ имеем $f \cdot h = f \cdot (f \cdot g) = h$ и $h \in \mathrm{Ann}(f \oplus 1)$. Если $h = 0$, то $g \in \mathrm{Ann}(f)$. □

Заметим, что при аффинных преобразованиях класс функций с $\mathrm{AI}(f) = d$ очевидно инвариантен относительно аффинных преобразований из $\mathrm{AGL}(n, 2)$. В то же время при действии преобразований $\alpha \in \mathrm{AGL}(n, 2)\mathcal{U}_1$ получаем:

$$\mathrm{AI}(f) = d \quad \Rightarrow \quad \mathrm{AI}(f^\alpha) \leqslant d + 1.$$

Теорема 17 ([38]). *Для любой двоичной функции от* n *переменных* $\mathrm{AI}(f) \leqslant \lceil \frac{n}{2} \rceil$.

Доказательство. Покажем, что для любой функции f найдется функция $0 \neq g \in \mathcal{U}_{\lceil \frac{n}{2} \rceil}$ такая, что $f \cdot g = h \in \mathcal{U}_{\lfloor \frac{n}{2} \rfloor}$. Рассмотрим множество функций A, состоящее из одночленов степени не выше $\lfloor \frac{n}{2} \rfloor$, и множество функций B, состоящее из всевозможных произведений функции f на одночлены степени не выше $\lceil \frac{n}{2} \rceil$. Так как

$$|A| + |B| = \sum_{i=0}^{\lfloor \frac{n}{2} \rfloor} \binom{n}{i} + \sum_{i=0}^{\lceil \frac{n}{2} \rceil} \binom{n}{i} = \sum_{i=0}^{n} \binom{n}{i} + \binom{n}{\lfloor \frac{n}{2} \rfloor} > 2^n,$$

то найдется линейное соотношение, связывающее функции из множества $A \bigcup B$. Оно и даст искомое равенство $f \cdot g = h$. □

Поэтому для классов $\mathrm{AI}_d = \{f \in \mathcal{F}_n : \mathrm{AI}(f) \leqslant d\}$ получаем включения:

$$\mathrm{AI}_1 \subset \mathrm{AI}_2 \subset \cdots \subset \mathrm{AI}_d \subset \mathrm{AI}_{d+1} \subset \cdots \subset \mathrm{AI}_{\lceil \frac{n}{2} \rceil} = \mathcal{F}_n.$$

Оценку мощности класса AI_d получается из следующей теоремы

Теорема 18 ([58]). *Вероятность того, что для случайной двоичной функции от* n *переменных выполнено включение* $f \in \mathrm{AI}(d)$, *оценивается сверху выражением*

$$P(\mathrm{AI}(f) \leqslant d) \leqslant \frac{2(2^{1+n+\cdots+\binom{n}{d}} - 1)\binom{2^n - 2^{n-d}}{2^{n-1} - 2^{n-d}}}{\binom{2^n}{2^{n-1}}}.$$

При небольших значений d эта вероятность стремится к нулю:

Теорема 19 ([58]). *Если* $d = d(n) \leqslant \mu n$, *где* $\mu = \frac{1}{2}\left(1 + \frac{\ln n}{2} - \sqrt{(1 + \frac{\ln n}{2})^2 - 1}\right) \approx 0,22$, *то* $P(\mathrm{AI}(f) \leqslant d) \to 0, n \to \infty$.

Нормальные и k-нормальные функции

Двоичная функция называется *k-нормальной*, если найдется многообразие $a+U$, $a \in V_n(2)$, $U \leqslant V_n(2)$, размерности $dimU = k$, на котором функция принимает постоянное значение. Двоичная функция называется *нормальной*, если она $\lfloor \frac{n}{2} \rfloor$-нормальна.

Например, функция константа $f(x) = c$ — n-нормальна, а аффинная функция $f(x) = (x, a^*) \oplus c$ при $a^* \neq 0$ — $(n-1)$-нормальна, так как она постоянна на многообразиях $\{x : (x, a^*) = 0\}$ и $\{x : (x, a^*) = 1\}$.

Понятие нормальности было введено в работе [41]. Оно применялось для построения классов уравновешенных функций с высокой нелинейностью. В дальнейшем оно оказалось полезным для характеризации многих классов бент функций. Так впервые бент функции с нарушением условия нормальности появляются только при $n \geqslant 14$ ([33]).

Проверка нелинейности тривиальным переборным алгоритмом требует проверки всех смежных классов по всем подпространствам данной размерности, что безусловно является очень нетривиальной задачей, так как число подпространств возрастает экспоненциально с ростом размерности. Модификация перебора с использованием процедуры последовательного поиска многообразий размерности $k+1$ по многообразиям размерности k, на которых функция принимает постоянное значение, позволяет немного улучшить время работы алгоритма. В работе [29] предлагается эффективный вероятностный алгоритм, основанный на случайном поиске, причем доказано, что он с наперед заданной вероятностью успеха строит искомое многообразие максимальной размерности. Для известных классов бент функций найдены все многообразия, на которых функция принимает постоянное значение, и подсчитано их число.

Построение классов функций с заданными свойствами

Заметим, что наличие аффинной классификации позволяет проводить полное описание классов функций, удовлетворяющих тем или иным свойствам. Из определений видно, что многие классы тесно связаны с линейными и аффинными преобразованиями на множестве аргументов функций. Например, множества бент-, платовидных и алгебраически-иммунных порядка t функций замкнуты относительно группы аффинных преобразований. Поэтому задача их описания сводится к построению представителей классов эквивалентности в аффинной классификации. Задача описания линейной структуры полностью сводится к задаче описания группы инерции функции в обобщенной группе сдвигов $(H_n)_f^{(0)}$.

К числу свойств, не инвариантных относительно аффинной группы относятся такие свойства, как (подробнее см. [14, 9]):

- *корреляционно-иммунные порядка t* ($CI(t)$);
- *устойчивые порядка t* (уравновешенные $CI(t)$);
- удовлетворяющие *строгому лавинному критерию (SAC)* (то есть функции f, у которых $\|f(x) \oplus f(x \oplus a)\| = 2^{n-1}$ при всех $a \in V_n(2)$ таких, что $\|a\| = 1$);
- удовлетворяющие *критерию SAC порядка t* ($SAC(t)$) (то есть функции, у которых все подфункции, полученные фиксацией не более t переменных, удовлетворяют критерию SAC);
- удовлетворяющие *критерию распространения* степени p ($PC(p)$) (то есть функции, у которых $\|f(x) \oplus f(x \oplus a)\| = 2^{n-1}$ при всех $a \in V_n(2)$ таких, что $1 \leqslant \|a\| \leqslant p$);
- удовлетворяющие *критерию распространения $PC(p)$* порядка t (то есть функции, у которых все подфункции, полученные фиксацией не более t переменных, удовлетворяют критерию $PC(p)$); и др.

В то же время, следует заметить, что, например, задача построения классов функций, удовлетворяющих критериям $CI(1)$ и $SAC = PC(1)$ может решаться путем поиска различных систем линейно независимых векторов, составляющих базис всего пространства, среди множества векторов, соответствующих нулевому значению коэффициента преобразования Уолша и функции автокорреляции соответственно. Поэтому наличие классификации, существенно облегчает задачу описания таких классов. Так, в работе [28] данный подход применяется для подсчета числа функций, принадлежащих классам

$CI(t)$ и $PC(p)$ и их пересечениям, в классах эквивалентности функций третьей степени нелинейности от шести переменных относительно группы $\mathrm{AGL}(6,2)\mathcal{U}_1$. В работе [51] на основе классификации кубических форм приводится описание кубических бент функций, а в работах [37, 30] приведена классификация возможных типов кубических $(n-4)$-устойчивых функций. Показано, что размерность пространства существенных переменных по модулю $\mathcal{U}_1$ не превосходит 6 и имеется ровно семь классов эквивалентности относительно группы $\mathrm{GL}(n,2))\mathcal{U}_1$, содержащих такие функции. Заметим, что в [5] найдено точное значение для общего числа $(n-4)$-устойчивых функций при $n \geqslant 10$, представляющее собой многочлен десятой степени от n.

Литература

[1] Амбросимов А. С., Шаров Н. Н. Некоторые асимптотические разложения для числа функций с нетривиально группой инерции. // Проблемы кибернетики. – 1979. – Вып. 36. – С.65–86.

[2] Денев И., Тончев В. О числе классов эквивалентности булевых функций относительно некоторых групп преобразований. // Матем. и матем. образование. Научн. съобщ. на 9-та практ. конф. на съюза на мат. в България, 1980, София. – 1980. – С. 41–43.

[3] Денисов О. В. Пороговая функция в эффекте Шеннона для булевых функций относительно симметрической группы.// Дискретная математика. – Т. 5. – Вып. 3. – 1993. – С. 64–75.

[4] Денисов О. В. Двоичные коды, образованные функциями с нетривиальной группой инерции. // Математические вопросы кибернетики. / Под ред. О.Б.Лупанова. – Вып. 11. –М.: Физматлит, 2002. – С.91–148.

[5] Кириенко Д. П. О числе корреляционно иммунных и устойчивых функций порядка $n-4$. // Материалы VIII Международного семинара «Дискретная математика и ее приложения» (Москва, 2 – 6 февраля 2004 г.). – М.: Изд-во механико-матем. ф-та МГУ. – 2004. – С. 421-424.

[6] Клосс Б. М., Нечипорук Э. Н. О классификации функций многозначной логики. // Проблемы кибернетики. – М.:Физматгиз, 1963. – Вып. 9. – С. 27–36.

[7] Кузнецов Ю. В., Шкарин С. А. Коды Рида – Маллера (обзор публикаций) // Математические вопросы кибернетики. – М.: Наука - Физматлит, 1996. – Вып. № 6. – С. 5–50.

[8] Логачев О. А., Ященко В. В., Сальников А. А. Об одном свойстве ассоциированных представлений группы $\mathrm{GL}(n,k)$. // Дискретная математика. – 2000. – Т.12. – Вып.2. – С.154–159.

[9] Логачев О. А., Ященко В. В., Сальников А. А. Булевы функции в теории кодирования и криптологии. М.: МЦНМО, 2004. – 469 с.

[10] Погорелов Б. А. О максимальных подгруппах симметрических групп, заданных на проективных пространствах над конечным полем. // Матем. заметки. – 1974. – Т.16. – № 1. С. 91–100.

[11] Рязанов Б. В., Чечета С. И. О приближении случайной булевой функции множеством квадратичных форм. // дискретная математика. – 1995. – Т. 7. – Вып. 3. – С. 130–145.

[12] Семенов А. С., Черемушкин А. В. Классификация функций степени нелинейности не выше трех от шести переменных. // Вопросы радиоэлектроники. Серия ЭВТ. – 1988. – Вып. 11. – С. 132–140.

[13] Страздинь И. Э. Аффинная классификация булевых функций пяти переменных. // Автоматика и вычислительная техника. – 1975. – № 1. – С. 1–9.

[14] Тарапников Ю. В. О корреляционно-иммунных и устойчивых булевых функциях. // Математические вопросы кибернетики. / Под ред. О.Б.Лупанова. – Вып. 11. –М.: Физматлит, 2002. – С.91–148.

[15] Тарапников Ю. В. О значениях аффинного ранга носителя спектра платовидных функций. // Математика и безопасность информационных технологий: Материалы конференции в МГУ 28-29 октября 2004 г. — М.: МЦНМО, 2005. - С. 226–231.

[16] Черемушкин А. В. Кубические формы от семи переменных. // 4 межгосуд. семинар по дискретной математике и ее прилож. 2–4 февраля 1993 г.: Сб. трудов / Под ред. О.Б.Лупанова. – М.:Изд-во механико-матем.ф-та МГУ, 1998. – С. 145.

[17] Черемушкин А. В. Классификация двоичных функций от шести переменных. // 4 межгосуд. семинар по дискретной математике и ее приложениям, 2–4 февраля 1993 г.: Сб. трудов / Под ред. О.Б.Лупанова. – М.:Изд-во механико-матем.ф-та МГУ, 1998. – С. 143–144.

[18] Черемушкин А. В. Кубические формы от восьми переменных. // Проблемы теоретической кибернетики. Тез. докл. XII Международной конференции (Нижний Новгород, 17–22 мая 1999 г.). Часть II. – М.:МГУ. – 1999. – С. 245.

[19] Черемушкин А. В. Методы аффинной и линейной классификации двоичных функций. // Труды по дискретной математике. Т.4. – М.: Физико-математическая литература. – 2001. –

[20] Черемушкин А. В. Однозначность разложения двоичной функции в бесповторное произведение нелинейных неприводимых сомножителей. // Вестник Московского государственного университета леса – Лесной вестник. – 2004. – № 4(35). – С. 86–190.

[21] Черемушкин А. В. Проблемы декомпозиции и линейной классификации дискретных функций. // Дискретные модели в теории управляющих систем: VI Международная конференция: Москва, 7-11 декабря 2004 г./ Ред. кол. В.Б.Алексеев, В.А.Захаров, Д.С.Романов. — М.: Изд. отдел факультета ВМиК МГУ им. М.В.Ломоносова (лицензия ИД № 05899 от 24.09.2001 г.), 2004. - С. 88–92.С. 273–314.

[22] Черемушкин А. В. О функциях с тривиальной группой инерции в обобщенных аффинных группах. // Вестник ТГУ. Приложение. Материалы международных, всесоюзных и региональных научных конференций, симпозиумов, школ, проводимых в ТГУ. – Томск: Изд. ТГУ. – № 9(I). – Август, 2004. – С. 41–44.

[23] Черемушкин А. В. Декомпозиция и классификация дискретных функций. – М.: ТВП – ОППМ, 2005. (в печати)

[24] Черемушкин А. В. Линейная и аффинная классификация дискретных функций. – М.: Математические вопросы кибернетики. / Под ред. О.Б.Лупанова. – Вып. 14. – М.: Физматлит, 2005. (в печати)

[25] Ashenhurst R. L. The application of counting techniques. // Proceedings of the Association for Computing Machinery, Pitsburg Meeting. – 1952. – pp. 293-305.

[26] Berlekamp E.R. and Welch L.R. Weight Distributions of the Cosets of the (32; 6) Reed-Muller Code. // IEEE Trans. Inform. Theory. – January 1972. – IT-18. – № 1. – pp. 203–207.

[27] Biryukov A., De Canni‘ere C., Braeken A., Preneel B. A Toolbox for Cryptanalysis: Linear and Affine Equivalence Algorithms. // EUROCRYPT'03. – LNCS 2656. – pp. 33-50.

[28] Braeken A., Borissov Y., Nikova S., Preneel B. Classification of Boolean Functions of 6 Variables or Less with Respect to Cryptographic Properties. // url: http://www.iacr.org.

[29] Braeken A., Wolf C., Preneel B. Classification of Highly Nonlinear Boolean Power Functions with a Randomised Algorithm for Checking Normality. //url: http://www.iacr.com/eprint/2004/214.

[30] Braeken A., Borissov Y., Nikova S., Preneel B. Classification of Cubic (n-4)-resilient Boolean Functions. // url: http://www.iacr.com/eprint/2005/332.

[31] Brier E., Langevin P. Classification of Boolean Cubic Forms of Nine Variables. // 2003 Information Theory Workshop (ITW 2003). – IEEE Press, 2003. – pp. 179–182.

[32] de Bruijn N. G. Generalization of Pólia's fundamental theorem in enumerative combinatorial analysis. // Nederl. Acad. Wetensch. Proc., Ser A. – vol.62. – Indag. Math. – 1959. – 21. – pp. 59–69.

[33] Canteaut A., Daum M., Dobbertin H, and Leander G. Normal and non-normal bent functions. // In Augot D., Charpin P., and Kabatianski G, editors. Workshop on Coding and Cryptography 2003. l'Ecole Superieure et d'Appliction des Transmissions, 2003. ISBN 2-7261- 1205-6. – 19 pages.

[34] Carlet C., Tarannikov Y. Covering sequences of Boolean functions and their cryptographic significance. Designes, Codes and Cryptography. – 2002. – Vol. 25. – pp. 263–279.

[35] Carlet C., Mesnager S. On the supports of Walsh transforms of Boolean functions. // url: http://eprint.iacr.org/2004/256.

[36] Carlet C. Vectorial Boolean functions for symmetric cryptography I. // url: www.cimpa-icpam.org/NotesCours/PDF/2005/Carlet05-5.pdf

[37] Carlet C., Charpin P. Cubic Boolean functions with highest resiliency. // IEEE Trans. Information Theory. — 2005. — Vol. 51. — № 2. — 562–571.

[38] Courtois N., Meier W. Algebraic attacks on stream ciphers with linear feedback. // In Advances in Cryptology-EUROCRYPT 2003. Springer-Verlag. – 2003. – vol. LNCS 2656. – pp. 346-359.

[39] Denev J. D., Tonchev V. D. On the number of equivalence classes of Boolean functions under a transformation group. // IEEE Trans. Inform. Theory. – 1980. – v. 26. – № 5. – pp. 625–626.

[40] Dixon L. E. Linear groups with exposition Galois field theory. – Leipzig, 1901. /2-е изд. – Dover Publications, New York, 1958.

[41] Dobbertin H. Construction of Bent functions and balanced Boolean functions with high nonlinearity. // In Fast Software Encryption - FSE 1994, volume 1008 of Lecture Notes in Computer Science, pages 61-74. Bart Preneel, editor, Springer, 1994.

[42] Dobbertin H., Leander G. Cryptographer's Toolkit for Construction of 8-Bit Bent Functions. // url: http://eprint.iacr.org/2005/089.

[43] Fuller J., and Millan W. On Linear Redundancy in the AES S-Box. // FSE 2003. – LNCS 2887. – Springer-Verlag. – pp. 249–266.

[44] J. Fuller. Affine equivalence classes. // url: http://www.isrc.qut.edu.au/people/fuller/.

[45] Harrison M. A. On the number of classes of (n,k)-switching networks. // J. Frankl. Inst. – 1963. – № 4. – p. 313–327.

[46] Harrison M. A. On the classification of Boolean function by the general linear and affine groups. // J. Soc. for Indust. and Appl. Math. – 1964. – v. 12. – № 2. – p. 285–299.

[47] Harrison M. A. Sur la classification des fonctioms logiques á plusyeurs valeurs. – Bull. Math. Soc. Sci. Math. de la R.S. de Roumantic. – 1969. – B (61). – №1. – pp. 41–54.

[48] Hou X.-D. Classification of cosets of the Reed-Muller code R(m-3,m). // Discrete Math. – Vol. 128. – 1994. – pp. 203–224.

[49] Hou X.-D. AGL$(m, 2)$ Acting on R(r,m)/R(s,m). // J.of Algebra. – 1995. – Vol. 171. – № 3. – pp. 921–938.

[50] Hou X.-D. GL$(m, 2)$ Acting on R(r,m)/R(r-1,m). // Discrete Math. – Vol. 149. – 1996. – pp. 99–122.

[51] Hou X.-D. GL$(m, 2)$ Cubic bent functions. // Discrete Math. – Vol. 189. – 1998. – pp. 149–161.

[52] Kantor W. M, McDough T. P. On the maximality of PSL$(d + 1, q)$, $d \geqslant 2$. // J. London Math. Soc. – Vol. 8. – № 3. – p. 426.

[53] Lechner R. J. Affine equivalence of switching functions. – Ph.D.Dissertation, Harvard Univ., Cambridge. Mass., January 1963. / Submitted to Bell Telephone Labs. as "Theory of switching" Harvard Computation Labs., Cambridge. Mass., Rept BL-33.

[54] Lechner R. J. A transform approach to login design. // IEEE Trans. Computers. – 1970. – v. C-19. – № 7. – pp. 627–640.

[55] List R. On permutation groups containing $\mathrm{PSL}_n(q)$ as a subgroup. // Geom. Dedic. – 1975. – Vol. 4. – № 2–4. – p. 373-375.

[56] Maiorana J.A. A Classification of the Cosets of the Reed-Muller code R(1,6). // Mathematics of Computation. – July 1991. – Vol. 57. – № 195. – pp. 403-414.

[57] Masaki S., Yoshiyuki I., Noburu T., Tadao K. Weight distribution of (128,64)–Reed–Muller Code. // IEEE Trans. Inform. Theory. – Vol. IT-17. – Sept., 1971. – pp. 627–628.

[58] Meier W., Pasalic E., Carlet C. Algebraic Attacks and Decomposition of Boolean Functions. // Eurocrypt 2004, LNCS 3027. – 2004. – pp.474–491.

[59] Meng Qing-shu, Yang min, Zhang huan-guo and Liu yu-zhen. Analysis of Affinely Equivalent Boolean Functions. // url: http://eprint.iacr.org/2005/025.

[60] Ninomia I. A study of the structures of boolean functions and its application to the synthesis of switching circuits. // IEEE Trans. Electronic Computers. – 1963. – v. EC-12. – p. 152.

[61] Rothaus O. S. On «bent» functions. // J. Combin. Theory. – 1976. – 20A. – pp. 300–306.

[62] Ryazanov B.V. Probabilistic methods in the theory of approximation of discrete functions. // In: Probabilistic Methods of Discrete Math.: Proc. 3rd Petrozavodsk Conf. TVP/VSP, Moscow/Utrecht, 1993, pp. 403–412.

[63] Sugita T., Kasami T., Fujiwara T. Weight distributions of the third and fifth order Reed-Muller codes of length 512. – Nara Inst. Sci. Tech. Report, Feb. 1996.

Часть III
Секция «Математические проблемы информационной безопасности»

Алгебраические атаки на потоковые шифры и алгебраическая иммунность булевых функций

Ю. В. Таранников

1 Потоковые шифры

Потоковый шифр, говоря немного упрощенно — это устройство с памятью, которое после введения в него «ключа», определяющего начальные значения ячеек памяти, действует автономно и производит псевдослучайную последовательность, которая преобразует исходное сообщение побитово или побайтово в зашифрованное сообщение, например, складываясь с ним побитово по модулю 2.

Главными требованиями к потоковым шифрам являются скорость их работы и надежность. Под надежностью понимается невозможность для противника за разумное время по некоторой имеющейся у него информации, например по схеме шифратора и перехваченным кускам выданной им псевдослучайной последовательности, определить всю псевдослучайную последовательность целиком, или, что равнозначно, раскрыть «ключ», что позволило бы противнику моментально читать все наши сообщения, зашифрованные с помощью этого ключа.

Одной из наиболее часто использующихся составных частей потоковых шифров является Регистр Сдвига с Линейной Обратной Связью (РСЛОС).

РСЛОС просто реализуется как элемент микросхемы и очень быстро работает. Использования одного только РСЛОС недостаточно, потому что существует много атак, позволяющих раскрывать «ключ» РСЛОС (начальные состояния его ячеек) за полиномиальное время относительно N — длины ключа, в то время как в идеале хотелось бы, чтобы противник не имел бы никакого более простого способа, чем перебирать все возможные варианты ключей, которых 2^N, и сравнивать производимые ими псевдослучайные последовательности с перехваченной.

Для того, чтобы избавиться от линейной зависимости выдаваемой РСЛОС псевдослучайной последовательности от начальных состояний ячеек, значения некоторых n ячеек РСЛОС в каждый момент времени подают на нелинейный фильтр, представляющий собой булеву функцию от n переменных. И уже выходное значение булевой функции является очередным элементом псевдослучайной последовательности. Модель поточного шифратора, основанная на РСЛОС и нелинейном фильтре, показана на рис. 1. Существуют, конечно, и другие модели, однако указанная является одной из самых распространенных.

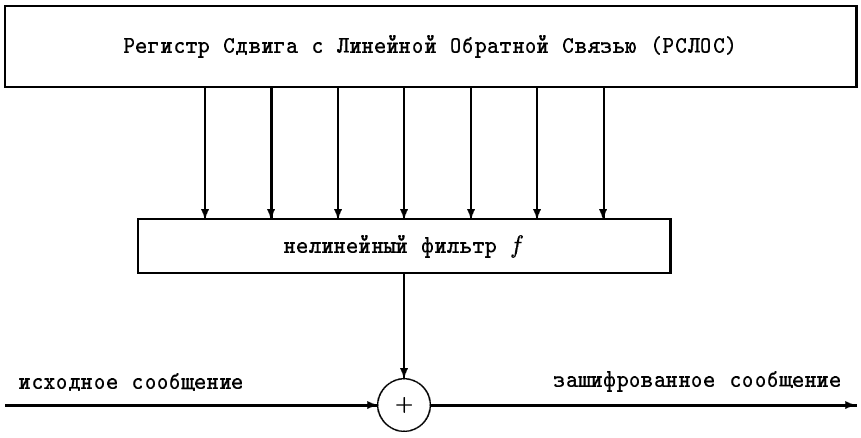


Рис. 1: Потоковый шифр, состоящий из РСЛОС и нелинейного фильтра

В 2002–2003 годах Н. Куртуа [12, 13] и ряд его коллег разработали новый вид криптографической атаки — так называемую алгебраическую атаку, оказавшаяся весьма эффективной против рассматриваемого нами здесь типа потоковых шифров. Применение этой атаки основано на построении точной или приближенной системы нелинейных уравнений невысокой степени, связывающей начальные значения ячеек памяти РСЛОС и значения перехваченной противником последовательности, и решения этой системы путем последующей линеаризации и получения из нее переопределенной системы линейных уравнений.

В связи с разработкой этой атаки Н. Куртуа выдвинул новые требования надежности, которым должна удовлетворять булева функция f, используемая в качестве нелинейного фильтра: функция f не только не должна иметь хорошей аппроксимации функциями невысокой степени, но и не должно существовать функции g невысокой степени, такой что функция $f \cdot g$ тоже невысокой степени или хорошо аппроксимируется функцией невысокой степени. Н. Куртуа показал, что против алгебраической атаки любой шифр с фильтром, у которого не более чем десять входов, заведомо не удовлетворяет требуемым критериям надежности.

2 Алгебраическая атака

Алгебраическая атака — это атака на шифр, заключающаяся в построении системы уравнений по возможности меньшей степени, связывающих элементы ключа, элементы выходной последовательности, а также, возможно, некоторые промежуточные параметры, и последующем решении построенной системы.

Ниже мы рассмотрим наиболее простые и общие виды алгебраических атак на потоковые шифры, использующие РСЛОС и нелинейный фильтр.

Элементы выходной последовательности шифра $\{b_i\}$ есть функции от начальных состояний регистра сдвига:

$$b_i = f(L^i(k_0, \ldots, k_{n-1})),$$

где n — длина регистра, L — линейный оператор переходов, задаваемый регистром, f — фильтрующая булева функция.

Зная некоторые элементы выходной последовательности $\{b_i\}$ (даже не обязательно последовательные), можно получить систему булевых уравнений от n переменных степени $\deg f$, из которой путем линеаризации можно получить систему линейных уравнений от $\sum\limits_{j=0}^{\deg f} \binom{n}{j}$ переменных.

Однако существуют подходы («сценарии»), упрощающие атаку:

- домножить функцию f на функцию g невысокой степени ($\leqslant d$), так чтобы получилась функция h невысокой степени ($\leqslant d$), тогда получается система уравнений степени, не превосходящей d;
- домножить функцию f на функцию g невысокой степени, так чтобы получился тождественный 0, в этом случае тоже получается система уравнений степени, не превосходящей d, только уравнения получаются только для элементов выходной последовательности, равных 1 (для элементов выходной последовательности, равных 0, получаются «уравнения» $0 = 0$).

То же самое вместо функции f можно проделать и с функцией $f + 1$.

Мейер, Пасалич, Карле [19] свели описанные сценарии к одному: для функции f должна найтись ненулевая функция g невысокой степени, такая что либо $fg = 0$, либо $(f + 1)g = 0$.

Соответственно, для того чтобы эффективная алгебраическая атака в таком виде не могла быть проведена, функция f, использующаяся в качестве нелинейного фильтра, должна обладать достаточно высокой *алгебраической иммунностью.*

Алгебраической иммунностью $\mathrm{AI}(f)$ функции f называется минимальная степень такой ненулевой функции g, что либо $fg = 0$, либо $(f + 1)g = 0$.

3 Нахождение для фильтрующей функции аннигилятора невысокой степени

Остановимся вкратце на вопросе нахождения для фильтрующей функции аннигилятора невысокой степени. Заметим, что поскольку схема шифра используется неоднократно, эту работу можно рассматривать как предвычисления. Обычный и наиболее естественный способ поиска аннигилятора заключается в составлении системы уравнений, переменными которой являются неопределенные коэффициенты при слагаемых полинома аннигилятора требуемой алгебраической степени.

Для функций, заданных коротким полиномом, В. В. Баев [1] разработал другой алгоритм построения системы, сложность которого зависит от числа переменных и числа мономов в полиноме функции. Какой из способов эффективнее — зависит от дополнительных условий (от соотношения параметров).

4 Быстрая алгебраическая атака и другие подходы

Составленную систему нелинейных уравнений на коэффициенты аннигилятора можно решать не в лоб, а сводя ее предварительно к системе меньшей степени, используя зависимость между коэффициентами. Этот метод получил название *быстрой алгебраической атаки* [14, 7, 18]. Проблема заключается в нахождении такой зависимости [8]. При быстрой алгебраической атаке могут эффективно использоваться дополнительные технические приемы: дискретное преобразование Фурье [18], базисы Гребнера и другие.

Изложенные здесь алгебраические атаки имеют и вероятностные версии. Они применимы, когда фильтрующая функция имеет очень хорошую аппроксимацию функцией с низкой алгебраической иммунностью.

Рассматриваются и алгебраические атаки на шифры с дополнительными ячейками памяти [6]. Результаты исследований говорят о том, что при перенесении известных методов алгебраических атак на этот случай, их эффективность уменьшается с ростом числа ячеек. Но и доказательство надежности получающихся шифров в этом случае затрудняется.

Часто стараются использовать шифры «с несколькими выходами», у которых за один такт работы генерируется несколько символов выходной последовательности [17]. Это увеличивает быстродействие шифра. Но при этом уменьшается надежность. Результаты исследований показывают, что в этом случае эффективность алгебраической атаки должна быть, вообще говоря, большей, хотя полученные до сих результаты носят большей частью теоретический характер.

Заметим, что строгого математического обоснования того, что системы имеют единственное решение, нет. Практические рекомендации говорят, что уравнений нужно получить «чуть» больше, чем будет переменных. Компьютерные эксперименты для небольших значений параметров подтверждают этот тезис.

5 Алгебраическая иммунность булевых функций

Алгебраическая иммунность булевых функций — это свойство, позволяющее противостоять некоторым видам алгебраических атак.

Напомним, что алгебраической иммунностью $\mathrm{AI}(f)$ функции f называется минимальная степень такой ненулевой функции g, что либо $fg = 0$, либо $(f+1)g = 0$.

Функция g такая, что $fg = 0$, называется *аннигилятором* функции f.

Верхняя оценка алгебраической иммунности любой булевой функции от n переменных оценивается следующим образом [13]:

$$\mathrm{AI}(f) \leqslant \left\lceil \frac{n}{2} \right\rceil .$$

Для разработанных к данному моменту алгебраических атак для того, чтобы алгебраические атаки на шифр с регистром сдвига длины 128 были не более эффективны, чем полный перебор ключей, алгебраическая иммунность фильтрующих булевых функций должна быть не меньше, чем примерно 12.

6 Построение булевых функций с высокой алгебраической иммунностью

В статье Мейера, Пасалича и Карле [19] показано, что алгебраическая иммунность «почти всех» булевых функций от n переменных асимптотически не меньше, чем примерно $0.27n$. Однако проблема состоит в доказательстве нижних оценок алгебраической иммунности конкретных последовательностей функций, причем желательно обладающих хорошими характеристиками и по другим криптографически важным свойствам.

К настоящему моменту применялись следующие способы доказательства нижних оценок алгебраической иммунности конкретных последовательностей булевых функций:

- *индуктивный* — для рекурсивных конструкций, прослеживается сохранение или возрастание совокупностей характеристик на каждом шаге построения;
- *анализ систем линейных уравнений* — при попытке построить аннигилятор меньшей степени с помощью неопределенных коэффициентов его полинома показывается, что получающаяся система линейных булевых уравнений имеет только нулевое решение.

А. А. Ботев в 2004 году [2, 3, 10, 5] для рекурсивной конструкции функций из [20], достигающих оптимального соотношения между нелинейностью и устойчивостью, доказал, что алгебраическая иммунность функций из этой последовательности равна $\Omega(\sqrt{n})$. Метод доказательства был индуктивным. А. А. Ботев построил также [4, 5] новое большое семейство функций с такими же характеристиками.

В 2005 году Далаи, Гупта и Майтра [15] построили рекурсивную последовательность функций, достигающих максимально возможного значения $\mathrm{AI}(f) = \lceil \frac{n}{2} \rceil$. Первоначальный метод доказательства — анализ систем линейных уравнений, однако в финальной версии работы на FSE они заменили его на индуктивное. Нелинейность функций из их работы оценивается как $2^{n-1} - \binom{n}{\lfloor n/2 \rfloor}$, что не очень хорошо, потому что относительная нелинейность в этом случае равна

$$\frac{\mathrm{nl}(f)}{2^n} \sim \frac{1}{2} - \frac{1}{\sqrt{\pi n/2}},$$

что стремится к $\frac{1}{2}$ недостаточно быстро.

В другой работе в 2005 году Далаи, Майтра и С. Саркар [15] построили еще один класс булевых функций, функций, достигающих максимально возможного значения алгебраической иммунности

$$\mathrm{AI}(f) = \left\lceil \frac{n}{2} \right\rceil.$$

Метод доказательства — анализ систем линейных уравнений, но очень простой: для одного за другим неопределенных коэффициентов аннигилятора показывается, что эти коэффициенты равны 0. К построенному классу относится и известная функция голосования. Нелинейность построенных функций оценивается той же величиной:

$$2^{n-1} - \binom{n}{\lfloor n/2 \rfloor}.$$

В статье Брекена и Принеля [11] аналогичные результаты распространены на некоторые виды симметрических функций, однако расширение класса функций, на которых достигается максимально возможная алгебраическая иммунность, довольно незначительно.

М. С. Лобанов в 2005 году доказал, что нелинейность любой булевой функции с алгебраической иммунностью k не меньше, чем $2^{n-1} - \sum_{i=k-1}^{n-k} \binom{n-1}{i} = 2\sum_{i=0}^{k-2} \binom{n-1}{i}$, причем эта оценка достигается при любых возможных значениях параметров n и k. Из этого неравенства, в частности, следует, что функции, построенные в [15], среди всех функций с максимально возможной алгебраической иммунностью имеют наихудшую возможную нелинейность.

Нами замечено, что имея функцию от k переменных с максимально возможной нелинейностью $\lceil \frac{k}{2} \rceil$, можно построить бент функцию (функцию с максимально возможной нелинейностью $2^{n-1} - 2^{n/2-1}$ и относительной нелинейностью $\frac{1}{2} - \frac{2}{2^{n/2+1}}$) от $n = 2k$ переменных с алгебраической иммунностью

примерно $\frac{n}{4}$, а на ее основе устойчивую функцию с теми же алгебраической иммунностью и относительной нелинейностью. Этого достаточно для практических целей. Так, для того чтобы современные методы быстрых алгебраических атак не давали выигрыша по сравнению с простым перебором ключей, достаточно, чтобы алгебраическая иммунность комбинирующих функций была не меньше, чем примерно 12. В соответствии со сделанным выше замечанием для этого достаточно использовать функцию от примерно 50 переменных, просто вычисляющуюся, нелинейность которой достаточна, чтобы противостоять быстрой корреляционной атаке.

Открытыми проблемами являются следующие:

- построение бент функции с максимально возможной алгебраической иммунностью;
- построение устойчивых функций с близкими к оптимальным алгебраической иммунностью и нелинейностью;
- построение более широких классов функций с оптимальным сочетанием требуемых криптографических свойств;
- формулировка критериев стойкости для функций-фильтров против других (более продвинутых) разновидностей алгебраических атак и построение функций, удовлетворяющих этим критериям.

Литература

[1] В. В. Баев, О некоторых алгоритмах построения аннигиляторов низкой степени для булевых функций, Дискретная математика, в печати.

[2] А. А. Ботев, Об алгебраической иммунности рекурсивных конструкций нелинейных фильтров, Математика и безопасность информационных технологий, Материалы конференции в МГУ 28–29 октября 2004, Изд-во МЦНМО, 2005, с. 131–135.

[3] А. А. Ботев, Об алгебраической иммунности одной рекурсивно заданной конструкции корреляционно иммунных функций, Труды XV международного семинара «Синтез и сложность управляющих систем», Новосибирск, 18–23 октября 2004, с. 8–12.

[4] А. А. Ботев, Об алгебраической иммунности новых конструкций фильтров с высокой нелинейностью, Труды VI международной конференции «Дискретные модели в теории управляющих систем», Москва, 7–11 декабря 2004, с. 227–230.

[5] А. А. Ботев, О свойствах корреляционно-иммунных функций с высокой нелинейностью, Диссертация на соискание ученой степени кандидата физико-математических наук, Москва, 2005.

[6] F. Armknecht, M. Krause, Algebraic attacks on combiners with memory, Crypto 2003, Lecture Notes in Computer Science, V. 2729, pp. 162–176, Springer-Verlag, 2003.

[7] F. Armknecht, Improving fast algebraic attacks, Fast Software Encryption, 2004, Lecture Notes in Computer Science, V. 3017, pp. 65–82, Springer-Verlag, 2004.

[8] F. Armknecht, On the existence of low-degree equations for algebraic attacks, available at eprint.iacr.org/2004/185. Also presented at SASC Ecrypt Workshop (State of the art in stream cipher, October 14–15, 2004).

[9] F. Armknecht, Algebraic attacks on stream ciphers, European congress on computational methods in applied sciences and engineering, ECCOMAS 2004, Jyvaskyla, 2004.

[10] A. Botev, Y. Tarannikov, Lower bounds on algebraic immunity for recursive constructions of nonlinear filters, Preprint, 2004.

[11] A. Braeken, B. Preneel, On the algebraic immunity of symmetric Boolean functions, Cryptology ePrint archive (http://eprint.iacr.org/), Report 2005/245.

[12] N. Courtois, Higher order correlation attacks, XL algorithm, and cryptanalysis of Toyocrypt, Proceedings of 5th International Conference on Information Security and Cryptology (ICISC 2002), November 28–29, 2002, Seoul, Korea, Lecture Notes in Computer Science, V. 2587, pp. 182–199, Springer-Verlag, 2002.

[13] N. Courtois, W. Meier, Algebraic attacks on stream ciphers with linear feedback, Advanced in Cryptology: Eurocrypt 2003, Warsaw, Poland, May 4–8, 2003, Proceedings, Lecture Notes in Computer Science, V. 2656, pp. 345–359, Springer-Verlag, 2003.

[14] N. Courtois, Fast algebraic attacks on stream ciphers with linear feedback, Crypto 2003, Lecture Notes in Computer Science, V. 2729, pp. 177–194, Springer-Verlag, 2003.

[15] D. K. Dalai, K. C. Gupta, S. Maitra, Cryptographically significant Boolean functions: construction and analysis in terms of algebraic immunity, Proceedings of FSE 2005, to appear in Lecture Notes in Computer Science.

[16] D. K. Dalai, S. Maitra, S. Sarkar, Basic theory in construction of Boolean functions with maximum possible annihilator immunity, Cryptology ePrint archive (http://eprint.iacr.org/), Report 2005/229.

[17] J. D. Golic, Vectorial Boolean functions and induced algebraic equations, Cryptology ePrint archive (http://eprint.iacr.org/), Report 2004/225.

[18] P. Hawkes, G. Rose, Rewriting variables: the complexity of fast algebraic attacks on stream ciphers, Advances in cryptology — Crypto 2004, Lecture Notes in Computer Science, V. 3152, pp. 390–406, Springer-Verlag, 2004.

[19] W. Meier, E. Pasalic, C. Carlet, Algebraic attacks and decomposition of Boolean functions, Advances in Cryptology — Eurocrypt 2004, Lecture Notes in Computer Science, V. 3027, pp. 474–491, Springer-Verlag, 2004.

[20] Y. Tarannikov, On resilient Boolean functions with maximal possible nonlinearity, Proceedings of Indocrypt 2000, Calcutta, India, December 10–13, 2000, Lecture Notes in Computer Science, V. 1977, pp. 19–30, Springer-Verlag, 2000.

Гомоморфизмы двоичных регистров сдвига

В. И. Солодовников

Гомоморфизмы являются неотъемлемой частью любой категории, в том числе и категории автоматов. В настоящем докладе описываются гомоморфизмы автоматов, называемых двоичными регистрами сдвига с линейной по входной переменной функцией обратной связи, то есть автоматов, вырабатывающих *двоичные линейно управляемые усложненные* ***рекуррентные*** *последовательности.* Оказывается, что всякий гомоморфизм такого регистра разлагается в композицию гомоморфизма на регистр и некоторого гомоморфизма, близкого к изоморфизму. При описании этого разложения основную роль играет некая операция, распространяющая *операцию «произведение многочленов» на множество всех двоичных функций.* Вопрос о гомоморфизмах регистра сводится к вопросу о поиске общих делителей его функции обратной связи и выходной функции.

Излагаемые ниже результаты работы [3] базируются на результатах К. Г. Таболова, В. А. Башева, А. Я. Прососова, внесших значительный вклад в разработку данной тематики.

Далее потребуются следующие обозначения и терминология.

- V_n — множество всех двоичных строк длины n, $n \geqslant 0$.
- F_n — множество всех двоичных функций от n переменных.
- F^*_{n+1} — множество всех линейных по последней (то есть $(n+1)$-й) переменной функций из F_{n+1}.

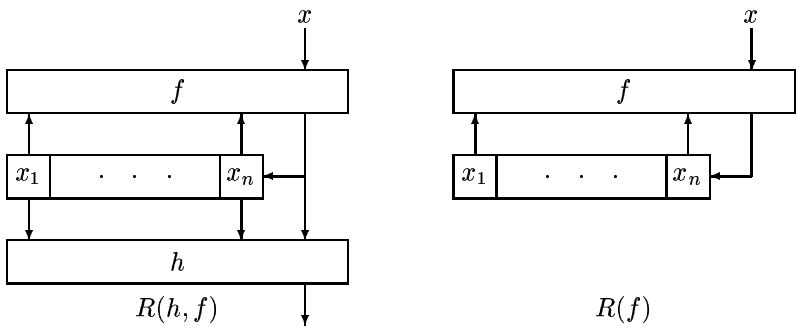
Через $\psi\varphi$ обозначается композиция отображений, при которой φ действует первым.

Автоматом с входным алфавитом X, множеством состояний S, выходным алфавитом Y, функцией переходов $\delta\colon S \times X \to S$ и функцией выходов $\lambda\colon S \times X \to Y$ называют пятерку объектов $A = (X, S, Y, \delta, \lambda)$. При этом подразумевается, что если автомат A находится в состоянии $s \in S$ и на его вход подается входной символ $x \in X$, то на выходе его появляется выходной символ $\lambda(s, x)$ и автомат переходит в состояние $\delta(s, x)$.

Развивая терминологию работы [1], двоичным ***регистром сдвига*** *длины n с функцией обратной связи f и выходной функцией h*, где $f, h \in F_{n+1}$, будем называть автомат $R(h, f) = (V_1, V_n, V_1, \rho, \mu)$, у которого функция переходов ρ и функция выходов μ определяются равенствами

$$\rho((x_1, \ldots, x_n), x) = (x_2, \ldots, x_n, f(x_1, \ldots, x_n, x)),$$
$$\mu((x_1, \ldots, x_n), x) = h(x_1, \ldots, x_n, f(x_1, \ldots, x_n, x))$$

для любых $x_1, \ldots, x_n, x \in V_1$. Регистр сдвига без выхода с функцией обратной связи f будем обозначать через $R(f)$. Эти автоматы изображаются следующим образом:



Гомоморфизмом автомата $A' = (X', S', Y', \delta', \lambda')$ в автомат $A = (X, S, Y, \delta, \lambda)$ называется тройка отображений (α, β, γ), $\alpha\colon X' \to X$, $\beta\colon S' \to S$, $\gamma\colon Y' \to Y$, удовлетворяющих условиям $\beta\delta' = \delta(\beta \times \alpha)$, $\gamma\lambda' = \lambda(\beta \times \alpha)$. Этот факт обозначают следующим образом: $(\alpha, \beta, \gamma)\colon A \to A'$. Если α и γ — тождественные вложения, то гомоморфизм называется *внутренним* и обозначается: $\beta\colon A \to A'$.

Сразу отметим, что, как и для любых автоматов, любой гомоморфизм $(\alpha, \beta, \gamma)\colon R(h,f) \to A$ регистра $R(h,f)$ в автомат $A = (V_1, S, V_1, \delta, \lambda)$ сводится к внутреннему гомоморфизму $\beta\colon R(\gamma h, f) \to A\alpha$, где $A\alpha = (V_1, S, V_1, \delta\alpha, \lambda\alpha)$, $\delta\alpha(s,x) = \delta(s, \alpha(x))$, $\lambda\alpha(s,x) = \lambda(s, \alpha(x))$. Поэтому далее будем рассматривать только внутренние гомоморфизмы.

Для любых $f_i \in F_{n_i+1}$, $i = 1, 2$, определим функцию

$$f_2 \triangleleft f_1 \in F_{n_1+n_2+1}$$

и отображение

$$\pi_{f_1,n_2}\colon V_{n_1+n_2} \to V_{n_2}$$

равенствами

$$\begin{aligned} &f_2 \triangleleft f_1(x_1, \ldots, x_{n_1+n_2+1}) \\ &\quad = f_2(f_1(x_1, \ldots, x_{n_1+1}), f_1(x_2, \ldots, x_{n_1+2}), \ldots, f_1(x_{n_2+1}, \ldots, x_{n_1+n_2+1})), \\ &\pi_{f_1,n_2}(x_1, \ldots, x_{n_1+n_2}) \\ &\quad = (f_1(x_1, \ldots, x_{n_1+1}), f_1(x_2, \ldots, x_{n_1+2}), \ldots, f_1(x_{n_2}, \ldots, x_{n_1+n_2})) \end{aligned}$$

для любых $x_1, \ldots, x_{n_1+n_2+1} \in V_1$, так что $f_2 \triangleleft f_1 = f_2\pi_{f_1,n_2+1}$.

Состояния автомата назовем *совпадающими с задержкой*, если для некоторого k любая входная последовательность длины k переводит их в одинаковые состояния.

Внутренний гомоморфизм автоматов $\beta\colon A' \to A$ назовем *изоморфизмом с задержкой*, если отображение β сюръективно и для любого состояния s автомата A все состояния полного прообраза $\beta^{-1}(s)$ совпадают с задержкой.

Основным результатом является следующая теорема о разложении произвольного гомоморфизма регистра в композицию гомоморфизма на регистр и изоморфизма с задержкой, причем разложении, по существу, однозначном.

Теорема 1. *Пусть* $n \geqslant 0$, $f \in F^*_{n+1}$, $h \in F_{n+1}$, $A = (V_1, S, V_1, \delta, \lambda)$, $\overline{A} = (V_1, S, \delta)$, $\beta\colon R(f) \to \overline{A}$ *— сюръективный гомоморфизм. Тогда существуют* $f_1 \in F^*_{n_1+1}$, $f_2 \in F^*_{n_2+1}$, *где* $n_1 + n_2 = n$, *и изоморфизм с задержкой* $\beta_2\colon R(f_2) \to \overline{A}$ *такие, что* $f = f_2 \triangleleft f_1$, $\beta_1 = \pi_{f_1,n_2}\colon R(f) \to R(f_2)$ *— сюръективный гомоморфизм и выполняются следующие свойства:*

1) $\beta = \beta_1\beta_2$;

2) *если* $\beta = \beta'_1\beta'_2$, *где* $\beta'_1\colon R(f) \to R(f'_2)$ *— гомоморфизм,* $\beta'_2\colon R(f'_2) \to \overline{A}$ *— изоморфизм с задержкой,* $f'_2 \in F^*_{n'_2+1}$, *то* $m = n_2 - n'_2 \geqslant 0$ *и для некоторого* $c \in V_1$ *выполняются равенства*

$$\begin{gathered} \beta'_1 = \pi_{f'_1,n'_2} = \pi_{(c\oplus x_{m+1}),n'_2}\beta_1, \quad \beta_2 = \beta'_2\pi_{(c\oplus x_{m+1}),n'_2}, \\ f'_1 = (c \oplus x_{m+1}) \triangleleft f_1, \quad f_2 = f'_2 \triangleleft (c \oplus x_{m+1}) \end{gathered}$$

и, в частности, если f *существенно зависит от* 1*-й переменной, то* $n_2 = n'_2$ *и* $f'_1 = f_1 \oplus c$, $f'_2 = f_2(x_1 \oplus c, \ldots, x_{n_2+1} \oplus c)$, $\beta'_2 = \beta_2(x_1 \oplus c, \ldots, x_{n_2} \oplus c)$;

3) *если* $\beta\colon R(h,f) \to A$ *— гомоморфизм, то существует* $h_2 \in F_{n_2+1}$ *такая, что* $h = h_2 \triangleleft f_1$ *и* $\beta_1\colon R(h,f) \to R(h_2,f_2)$, $\beta_2\colon R(h_2,f_2) \to A$ *— гомоморфизмы.*

Автомат называют *подстановочным*, если все его частичные функции переходов являются подстановками множества состояний. Очевидно, что для двоичного регистра сдвига ненулевой длины подстановочность равносильна *линейности функции обратной связи по* 1*-й переменной*, а в случае линейности функции обратной связи по последней переменной равносильна еще и отсутствию различных совпадающих с задержкой состояний. Следовательно, не подстановочный регистр сдвига без выхода длины большей 1 с линейной по последней переменной функцией обратной связи всегда имеет нетривиальные изоморфизмы с задержкой. Из теоремы 1 получаем следствие 1, доказанное А. Я. Прососовым для регистров без выхода.

Следствие 1. *Любой гомоморфный образ двоичного регистра сдвига с линейной по* 1*-й и последней переменным функцией обратной связи изоморфен некоторому двоичному регистру сдвига с линейной по* 1*-й и последней переменным функцией обратной связи.*

Следствие 2. *Любой гомоморфный образ двоичного регистра сдвига с линейной по последней переменной функцией обратной связи и линейной по первой и последней переменным выходной функцией изоморфен некоторому двоичному регистру сдвига с линейной по последней переменной функцией обратной связи и линейной по первой и последней переменным выходной функцией.*

Рассмотрим теперь *множество всех двоичных функций* $F = \bigcup_{n=0}^{\infty} F_{n+1}$. Относительно операции $\triangleleft$ оно является моноидом с единичным элементом $x_1 \in F_1$. Следовательно, на множестве F возникает соответствующее транзитивное и рефлексивное бинарное отношение *делимости (справа)* $|$, которое определяется следующим образом: $f_1 \mid f$ *тогда и только тогда, когда* $f = f_2 \triangleleft f_1$ *для некоторой* $f_2 \in F$ *(которую назовем частным от деления* f *на* f_1*, а* f_1 *— делителем* f*).* Это отношение не является симметричным, поскольку $f_1 \mid f$ и $f \mid f_1$ только в случае, когда f и f_1 являются функциями от одинакового количества переменных и $f \oplus f_1 = \mathrm{const}$. Кроме того, частное не всегда определено однозначно. Точнее, справедливо следующее свойство: частное от деления любой функции f на функцию f_1 определено однозначно тогда и только тогда, когда f_1 является *функцией без запретов* (то есть отображение π_{f_1,n_2} является сюръективным для любого $n_2 \geqslant 0$, см. [2]). В частности, все функции из подмоноида $F^* = \bigcup_{n=0}^{\infty} F_{n+1}^*$ моноида F являются функциями без запретов.

Введенное отношение делимости функций является продолжением отношения делимости многочленов в следующем смысле. Обозначим через L^* подмоноид моноида F^*, состоящий из всех линейных функций из F^*, а через $\mathrm{GF}(2)[x]*$ — моноид (относительно умножения) всех ненулевых многочленов над полем $\mathrm{GF}(2)$. Тогда отображение $\varphi\colon \mathrm{GF}(2)[x]* \to L^*$, где

$$\varphi(c_0 \oplus c_1 x \oplus \ldots \oplus c_n x^n) = c_0 x_1 \oplus c_1 x_2 \oplus \ldots \oplus c_n x_{n+1},$$

является изоморфизмом и, следовательно, сохраняет отношение делимости.

Делитель функции f назовем *собственным*, если число его переменных больше 1 и меньше числа переменных функции f. Не имеющие собственных делителей функции назовем *неприводимыми.*

Следствие 3. *Пусть* $n \geqslant 0$, $f \in F_{n+1}^*$ *и* f *линейна по* 1*-й переменной. Тогда следующие утверждения равносильны:*

1) *автомат* $R(f)$ *не имеет гомоморфизмы на автоматы с числом состояний, большем* 1 *и меньшем* 2^n*;*

2) *функция* f *неприводима.*

Наибольшим общим делителем функций $h, f \in F$ назовем любой их общий делитель от наибольшего количества переменных. Множество всех наибольших общих делителей функций h и f обозначим через (h, f). Оно не пусто, поскольку функции $x_1, 1 \oplus x_1 \in F_1$ являются делителями любой функции из F. Функции $h, f \in F$ назовем *взаимно простыми*, если $(h, f) = \{x_1, 1 \oplus x_1\}$.

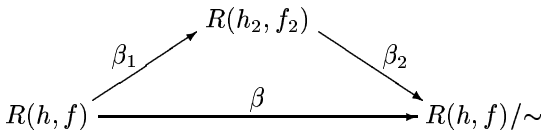
Следующая теорема устанавливает связь между приведенной формой регистра сдвига и наибольшим общим делителем его функций.

Теорема 2. *Пусть* $n \geqslant 0$, $f \in F_{n+1}^*$, $h \in F_{n+1}$. *Тогда* $(h, f) = \{f_1, 1 \oplus f_1\}$, *где* $f_1 \in F_{n_1+1}^*$, $n_1 \leqslant n$, *причем для любой функции* $f_1' \in F$ *следующие утверждения равносильны:*

1) $f_1' \mid h$, $f_1' \mid f$*;*

2) $f_1' \mid f_1$.

При этом существует коммутативная диаграмма гомоморфизмов автоматов



где β *— естественный гомоморфизм автомата* $R(h, f)$ *на свою приведенную форму* $R(h, f)/\sim$, $\beta_1 = \pi_{f_1,n_2}$, $n_2 = n - n_1$, $f_2 \in F_{n_2+1}^*$, $h_2 \in F_{n_2+1}$, $f = f_2 \triangleleft f_1$, $h = h_2 \triangleleft f_1$, β_2 *— изоморфизм с задержкой.*

Следствие 4. *Пусть* $n \geqslant 0$, $f \in F_{n+1}^*$, $h \in F_{n+1}$ *и выполнено хотя бы одно из условий:*

a) *f линейна по 1-й переменной;*

b) *h линейна по 1-й и $(n+1)$-й переменным.*

Тогда следующие утверждения равносильны:

1) *автомат $R(h,f)$ не имеет различных эквивалентных состояний (то есть минимален);*

2) *функции h и f взаимно просты.*

Изложенные результаты частично обобщаются на случай регистра $R(h,f)$, где $f\colon X^{n+1}\to X$, $h\colon X^{n+1}\to Y$, X,Y — произвольные конечные множества, следующим образом.

Функцию $\varphi\colon X^{m+1}\to X$ назовем *биективной по последней* (то есть $(m+1)$-й) *переменной*, если при любой фиксации всех остальных переменных получаемое отображение $X\to X$ является биекцией. В этом случае через φ^{-1} обозначим функцию, *обратную к функции φ по последней переменной*, то есть $\varphi^{-1}\colon X^{m+1}\to X$ и $\varphi^{-1}(x_1,\ldots,x_m,\varphi(x_1,\ldots,x_{m+1}))=x_{m+1}$.

Теорема 3. *Пусть $n,n_2\geqslant 0$, $f\colon X^{n+1}\to X$, $f_2\colon X^{n_2+1}\to X$ — биективные по последней переменной функции, $h\colon X^{n+1}\to X$, $h_2\colon X^{n_2+1}\to X$, $\beta\colon X^n\to X^{n_2}$. Тогда следующие утверждения равносильны:*

1) *$\beta\colon R\left(h,f^{-1}\right)\to R\left(h_2,f_2^{-1}\right)$ — гомоморфизм;*

2) *$n\geqslant n_2$, $\beta=\pi_{f_1,n_2}$, $f=f_2\triangleleft f_1$, $h=h_2\triangleleft f_1$ для некоторой биективной по последней переменной функции $f_1\colon X^{n_1+1}\to X$, где $n_1=n-n_2$.*

Для случая двоичных регистров без выхода импликация (2) $\Longrightarrow$ (1) была получена К. Г. Таболовым, а импликация (1) $\Longrightarrow$ (2) — В. А. Башевым.

Литература

[1] Golomb S. W. Shift register sequences. Holden-Day, San Francisco, 1967.

[2] Сумароков С. Н. Запреты двоичных функций и обратимость для одного класса кодирующих устройств. Обозрение прикладной и промышленной математики, 1994, **1**, № 1, 33–55.

[3] Солодовников В. И. Гомоморфизмы двоичных регистров сдвига. Дискретная математика, 2005, **17**, № 1, 73–88.

Применение сплетений для построения гибких высокоскоростных алгоритмов поточного шифрования с гарантированными свойствами

В. С. Анашин

Напомним следующее определение. Пусть $U\colon Z \to Z$ и $\mathcal{V} = \{(V_z\colon X \to X)\colon z \in Z\}$ — отображения соответствующих множеств. *Сплетением* (в другой терминологии, *косым произведением*, или *косым сдвигом*) называется следующее отображение декартова произведения $Z \times X$ в себя:

$$U \curlywedge \mathcal{V}\colon (z,x) \mapsto (U(z), V_z(x))$$

Сплетения часто используются в криптографии (явно, а чаще неявно): например, т.н. сеть Фейстеля есть ни что иное как композиция сплетения $(z,x) \mapsto (z, x \oplus f(z))$ с последующей перестановкой координат, булевы функции треугольного вида — это композиции сплетений, полупрямое произведение автоматов тоже основано на сплетениях, и т.п.

В сообщении будет изложена математическая теория сплетений для отображений кольца целых 2-адических чисел, с помощью которой можно строить алгоритмы поточного шифрования, показывающие высокие скоростные характеристики независимо от используемой платформы (т.е. от типа процессора и операционной среды), и обладающие рядом *математически доказуемых* криптографических свойств, таких, например, как

- большая длина L периода гаммы;
- отсутствие перекрытий гаммы на разных ключах;
- равномерное распределение гаммы;
- высокий линейный ранг гаммы;
- отсутствие линейных статистических аналогов ранга, меньшего $\frac{L}{2}$, совпадающих с гаммой на числе знаков, превышающем $\frac{L}{2}$. [1]

В качестве иллюстрации этой теории будет рассмотрен алгоритм ABC (версия 2): скорость выработки гаммы соответствующей программой на C (без опций, ориентированных на конкретный тип процессора) составляет 6,91 Гбит/сек на процессоре Пентиум-4 с тактовой частотой 3,2 Мгц.

Отметим, что в иной терминологии можно вместо вышеуказанных сплетений говорить о неавтономных неархимедовых динамических системах. Таким образом, излагаемая теория позволяет использовать аппарат неархимедовой динамики для обоснования криптографических свойств поточных шифраторов.

Литература

[1] V. Anashin. Pseudorandom Number Generation by p-adic Ergodic Transformations. 2004. http://arXiv.org/abs/cs.CR/0401030.

[2] V. Anashin. *Pseudorandom Number Generation by p-adic Ergodic Transformations: an Addendum.* 2004. http://arXiv.org/abs/cs.CR/0402060.

[1] Вопрос о линейных статистических аналогах был задан автору во время его выступления на МаБИТ-04; теперь на него получен ответ.

[3] В. С. Анашин. Равномерно распределенные последовательности целых p-адических чисел, II. Дискретная математика, **14** (2002), № 4, с. 3–64. Препринт на английском доступен на http://arXiv.org/abs/math.NT/0209407.

[4] V. Anashin. Uniformly distributed sequences over p-adic integers. Number theoretic and algebraic methods in computer science. Proceedings of the Int'l Conference (Moscow, June–July, 1993) (A. J. van der Poorten, I. Shparlinsky and H. G. Zimmer, eds.), World Scientific, 1995, 1–18.

[5] В. С. Анашин. Равномерно распределенные последовательности целых p-адических чисел. Мат. заметки, **55** (1994), № 2, с. 3–46.

[6] V. S. Anashin. Uniformly distributed sequences in computer algebra, or how to construct program generators of random numbers. J. Math. Sci. (Plenum Publishing Corp., New York), **89** (1998), No 4, 1355 – 1390.

[7] Vladimir Anashin, Andrey Bogdanov, Ilya Kizhvatov, and Sandeep Kumar. ABC: A new fast flexible stream cipher. Version 2, July 2005. http://crypto.rsuh.ru/papers/abc-v2.pdf.

Совершенная схема множественного разделения секрета на основе линейных преобразований над конечным цепным коммутативным кольцом

А. Н. Алексейчук, А. Л. Волошин, Л. В. Скрыпник

1. Схема множественного разделения секрета (multi-secret sharing scheme; СМРС) [1, 2] представляет собой криптографический протокол, позволяющий «разделять» одновременно несколько секретных ключей (секретов) среди участников схемы таким образом, чтобы только заранее определенные подмножества (коалиции) участников могли восстановить значения определенных ключей при объединении своих компонент (проекций секретов). Если при этом участники каждой коалиции не получают никакой апостериорной информации об остальных ключах (к которым, согласно протоколу, они не имеют права доступа), то соответствующая СМРС называется совершенной [2].

Естественный тривиальный способ задания СМРС состоит в построении нескольких обычных схем разделения секрета (СРС) (см., например, [3], гл. 5), каждая из которых независимо от остальных используется для разделения «своего» секретного ключа из заданной совокупности ключей. Как правило, такое решение оказывается малопрактичным, поскольку участникам СМРС приходится хранить большой объем секретной информации.

Различные способы построения СРС на основе линейных преобразований (кодов) над конечными полями или векторными пространствами изучались в [3, 4, 5, 6] и ряде других работ. В [7] предложена «векторная» конструкция, в общем случае, несовершенной, СРС над кольцом Галуа и получено частичное описание иерархии доступа [8] на множестве участников такой схемы разделения секрета.

Ниже предлагается метод построения совершенных СМРС, которые отличаются от представленных в [1, 2] и являются прямым обобщением «векторных» схем разделения секрета над конечными полями [4] и кольцами Галуа [7]. Охарактеризованы иерархии доступа предложенных СМРС и получены необходимые и достаточные условия существования СМРС данного типа для произвольной заранее определенной иерархии доступа. Предложен алгоритм построения схемы множественного разделения секрета для заданной иерархии доступа, обобщающий известный ранее алгоритм построения «векторной» СРС над конечным полем [6].

2. Пусть R - конечное локальное коммутативное кольцо главных идеалов (коммутативное цепное кольцо) с радикалом J и полем вычетов $\bar{R} = R/J = \mathbf{GF}(q)$ [9]. Обозначим d индекс нильпотентности радикала J. Зафиксируем элемент $a \in J \backslash J^2$ и произвольное отображение $\gamma : \bar{R} \to R$ такое, что $\gamma(r + J) \equiv r (\mathrm{mod} J)$ для любого $r \in R$. Согласно [9], идеалы кольца R образуют цепь: $R \supset J \supset J^2 \supset \ldots \supset J^d = 0$. При этом для любого $k \in \overline{0, d}$ выполняются равенства $J^k = Ra^k$, $|J^k| = q^{d-k}$, и каждый элемент $r \in R$ может быть однозначно представлен в виде

$$r = r[0] + r[1]a + \ldots + r[d-1]a^{d-1}, \tag{1}$$

где $r[j] \in \gamma(\bar{R})$, $j \in \overline{0, d-1}$.

Опишем комбинаторную модель предлагаемой схемы множественного разделения секрета.

Пусть $S = \bar{R}^d$ - множество всех упорядоченных наборов d секретных ключей (каждый из которых является элементом поля $\bar{R}$), подлежащих распределению между участниками из множества $P = \{1, 2, \ldots, n\}$, $n \geqslant 2$. Зафиксируем $k \times (n+1)$-матрицу

$$G = \left(\begin{array}{c|c} 1 & \\ 0 & \\ \vdots & G' \\ 0 & \end{array} \right), \tag{2}$$

над кольцом R, где $k \geqslant 2$, столбцы которой занумеруем слева направо числами $0, 1, \ldots, n$. Матрице (2) поставим в соответствие СМРС $\sigma(G)$, которую определим следующим образом.

Пусть $(s_j : j \in \overline{0,d-1})$ - произвольный элемент множества S. Тогда для нахождения проекций ключей s_j, $j \in \overline{0,d-1}$, дилер СМРС

(а) вычисляет элемент

$$s = \sum_{j=0}^{d-1} \gamma(s_j) a^j \in R; \tag{3}$$

(б) независимо, случайно и равновероятно выбирает элементы $a_1, \ldots, a_{k-1} \in R$ и находит вектор $(s, \pi_1, \ldots, \pi_n) = (s, a_1, \ldots, a_{k-1})G$, последние n координат которого объявляются проекциями упорядоченного набора ключей $(s_j : j \in \overline{0,d-1})$. При этом элемент $\pi_i \in R$ доставляется i-му участнику СМРС, $i \in \overline{1,n}$.

Для любого $j \in \overline{0,d}$ обозначим символом $\tilde{\Sigma}_j$ совокупность всех множеств A участников СМРС $\sigma(G)$, которые могут однозначно восстановить по имеющимся у них проекциям секретные ключи $s_0, s_1, \ldots, s_{d-j-1}$ и только их. Отметим, что

$$\tilde{\Sigma}_{j_1} \cap \tilde{\Sigma}_{j_2} = \varnothing, j_1, j_2 \in \overline{0,d}, j_1 \neq j_2, \bigcup_{j=0}^{d} \tilde{\Sigma}_j = 2^P,$$

где 2^P - совокупность всех подмножеств множества P. Следуя терминологии [8], назовем семейство множеств $\tilde{\Sigma} = \{\tilde{\Sigma}_j : j \in \overline{0,d}\}$ иерархией доступа СМРС $\sigma(G)$.

Как показывает следующая теорема, участники, принадлежащие произвольной коалиции $A \in \tilde{\Sigma}_j$ $(j \in \overline{0,d})$, не могут получить из имеющихся у них проекций никакой апостериорной информации о секретных ключах s_l с номерами $l \in \overline{d-j,d-1}$. Таким образом, СМРС $\sigma(G)$ является совершенной (комбинаторной) схемой множественного разделения секрета.

Теорема 1. *Пусть M - модуль над кольцом R, порожденный строками матрицы G вида (2). Для любого $U \subseteq P_0 \stackrel{def}{=} P \cup 0$ обозначим $\|M_U\|$ число различных векторов, содержащихся в столбцах с номерами из множества U таблицы (размера $|M| \times (n+1)$), составленной из элементов модуля M. Пусть, далее, $j \in \overline{0,d}$ и $\tilde{\Sigma}_j \neq \varnothing$. Тогда для любого $A \in \tilde{\Sigma}_j$ выполняется равенство $\|M_{A \cup 0}\| = q^j \|M_A\|$.*

Назовем построенную СМРС $\sigma(G)$ линейной схемой множественного разделения секрета над кольцом R. Отметим, что в частном случае $d = 1$ эта СМРС представляет собой обычную «векторную» СРС над полем $\mathbf{GF}(q)$ [4]. Если R является кольцом Галуа, то пункт (б) изложенного выше алгоритма аналогичен процедуре вычисления проекций (одного) секретного ключа $s \in R$ в несовершенной СРС, описанной в [7].

Обозначим G_A подматрицу матрицы G, содержащуюся в ее столбцах с номерами из множества $A \subseteq P$. Символом $G_i^{\downarrow}$ обозначим i-й столбец матрицы G, $i \in \overline{0,n}$; в частности, $G_0^{\downarrow} = (1, 0, \ldots, 0)^T \in R^{(k)}$.

Следующая теорема, обобщающая один из результатов статьи [7], содержит полное описание иерархии доступа СМРС $\sigma(G)$.

Теорема 2. *Пусть $A \subseteq P$, $j \in \overline{0,d}$. Тогда $A \in \tilde{\Sigma}_j$ в том и только том случае, когда j является наименьшим целым числом от 0 до d, для которого совместна система линейных уравнений (СЛУ)*

$$G_A x^{\downarrow} = a^j G_0^{\downarrow} \tag{4}$$

над кольцом R.

Отметим, что утверждение теоремы 2 по существу содержит в себе алгоритм восстановления секретных ключей s_l, $l \in \overline{0,d-j-1}$ участниками произвольной коалиции $A \in \tilde{\Sigma}_j$, $j \in \overline{0,d-1}$, основанный на соотношениях (2) - (4) и однозначности представления элементов кольца R в виде (1).

3. Пусть теперь задано семейство $\Sigma = \{\Sigma(i) : i \in \overline{0,d}\}$ попарно не пересекающихся подмножеств множества 2^P (случай $\Sigma(i) = \varnothing$ не исключается), таких, что $\bigcup_{j=0}^{d} \Sigma(i) = 2^P$.

Требуется установить необходимые и достаточные условия, при которых семейство Σ является иерархией доступа некоторой линейной СМРС над кольцом R и (в случае существования) построить в явном виде матрицу G вида (2), задающую такую СМРС.

Обозначим $A_{j,1}, \ldots, A_{j,r_j}$ все минимальные (относительно включения) элементы совокупности множеств $\Sigma(j)$, $j \in \overline{0,d}$.

Теорема 3. *Тогда и только тогда существует линейная над кольцом R СМРС с иерархией доступа Σ, когда выполняются следующие условия:*

(1) $\varnothing \in \Sigma(d)$;

(2) для любого $j \in \overline{0,d}$ класс множеств $\Delta(j) \stackrel{def}{=} \bigcup_{l=0}^{j} \Sigma(l)$ является монотонным (то есть из условий $A \in \Delta(j)$, $B \in 2^P$, $A \subseteq B$ следует, что $B \in \Delta(j)$);

(3) существует матрица C над кольцом R, состоящая из $r = r_0 + \ldots + r_{d-1}$ строк вида $\overrightarrow{c_{j,l}} = (a^j, \overrightarrow{f_{j,l}})$, где $\overrightarrow{f_{j,l}} \in R^n$, $j \in \overline{0,d-1}$, $l \in \overline{1,r_j}$, такая, что

(а) для любых $j \in \overline{0,d-1}$, $l \in \overline{1,r_j}$ множество номеров ненулевых координат вектора $\overrightarrow{f_{j,l}}$ равно $A_{j,l}$;

(б) для любого $j \in \overline{0,d-1}$ и произвольного максимального (относительно включения) элемента X класса $2^P \backslash \Delta(j)$ совместна СЛУ

$$C_{\overline{X}} x^{\downarrow} = a^{d-(j+1)} C_0^{\downarrow}$$

над кольцом R, где $C_{\overline{X}}$ - подматрица матрицы C, состоящая из ее столбцов с номерами из множества $\overline{X} \stackrel{def}{=} P \backslash X$, $C_0^{\downarrow}$ - столбец матрицы C с номером, равным 0.

При выполнении условий (1) - (3), в качестве строк матрицы G, задающей СМРС с иерархией доступа Σ, можно взять элементы подходящей системы образующих модуля решений СЛУ $Cx^{\downarrow} = 0^{\downarrow}$ над кольцом R.

Отметим, что в частном случае $d = 1$ справедливость теоремы 3 вытекает из результатов, изложенных в [6].

Авторами предложен алгоритм, позволяющий проверять существование и (в случае положительного результата проверки) строить искомую матрицу G, задающую линейную СМРС с иерархией доступа Σ. Указанный алгоритм состоит из этапа предварительных вычислений и процедуры последовательного формирования строк матрицы C, удовлетворяющей условиям (а), (б) теоремы 3, по методу поиска с возвращением. Временная сложность последней составляет

$$T = O\left(n^2 d^n \sum_{j=0}^{d-1} |\overline{\Delta}_j^1| q^{\left(\sum_{j=0}^{d-1} \sum_{l=1}^{r_j} |A_{j,l}| - n\right)} \right)$$

арифметических операций в кольце R, где $\overline{\Delta}_j^1$ - совокупность всех максимальных элементов класса множеств $2^P \backslash \Delta(j)$, $j \in \overline{0,d-1}$.

4. Важной, с практической точки зрения, является задача нахождения необходимых и достаточных условий, при которых предложенная схема множественного разделения секрета является оптимальной (среди всех СМРС, реализующих данную иерархию доступа) по критерию минимума наибольшей из длин проекций секретных ключей, хранящихся у ее участников. В настоящее время полное решение этой задачи авторам не известно, однако построены отдельные семейства линейных схем множественного разделения секрета, обладающих указанным свойством оптимальности.

Литература

[1] Jackson W.-A., Martin K.M., O'Keefe C.M. Multisecret threshold schemes // Advances in Cryptology - CRYPTO'93. - Lecture Notes in Computer Science. - V. 773. - P. 126 - 135.

[2] Blundo C., De Santis A., Di Crescenzo G., Gaggia A.G., Vaccaro U. Multi-secret sharing schemes // Advances in Cryptology - CRYPTO'95. - Lecture Notes in Computer Science. - V. 832. - P. 150 - 163.

[3] Введение в криптографию / Под общ. ред. В. В. Ященко. - М.: МЦНМО-ЧеРо. - 1999. - 272 с.

[4] Brickell E.F. Some ideal secret sharing schemes // J. Combin. Math. and Combin. Comput. - 1989. - №9. - P. 105 - 113.

[5] Blakley G.R., Kabatianski G.A. Linear algebra approach to secret sharing schemes // Preproc. of Workshop on Information Protection.: Moscow, 1993.

[6] van Dijk M. A linear construction of perfect secret sharing schemes // Advances in Cryptology - EUROCRYPT'94. - Lecture Notes in Comput. Science. - V. 950. - P. 23 - 34.

[7] Ashikhmin A., Barg A. Minimal vectors in linear codes // IEEE Trans. on Inform. Theory. - 1998. - V. 5. - P. 2010 - 2018.

[8] Kurosawa K., Okada K., Sakano K., Ogata W., Tsujii S. Nonperfect secret sharing schemes and matroids // Advances in Cryptology - EUROCRYPT'93. Lecture Notes in Comput. Science. - V. 765. - P. 126 - 141.

[9] Нечаев А.А. Конечные кольца главных идеалов // Мат. сб. - 1973. - Т. 91. - №3. - С. 350 - 366.

О некоторых комбинаторно-групповых задачах в криптографии

М. В. Шеблаев

Одним из актуальных направлений современной криптографии является криптография с открытым ключом. Наряду с классическими схемами, использующими задачи теории чисел, в последнее время были предложены новые схемы и протоколы, основанные на комбинаторной теории групп.

Определение 1. Группа кос B_n — бесконечная некоммутативная группа, которая задается следующим образом:

$$B_n = \{\sigma_1, \sigma_2, \ldots, \sigma_{n-1} \,|\, \sigma_i\sigma_j = \sigma_j\sigma_i, |i-j| > 1; \sigma_i\sigma_j\sigma_i = \sigma_j\sigma_i\sigma_j, |i-j| = 1\}.$$

В работах [1], [2] были предложены протоколы обмена ключей, использующие вычислительную сложность некоторых модификаций задачи поиска сопрягающих элементов в группе B_n.

В работе [3] показано, что на самом деле для криптоанализа этих протоколов существенно важными является следующая задача:

Определение 2. Проблема принадлежности подгруппе: для заданных $x \in B_n, H \subset B_n$ установить, верно ли что $x \in H$.

Основными результатами доклада являются следующие теоремы:

Теорема 1. *Пусть $H = \langle y \rangle$ — циклическая подгруппа группы B_n, порожденная элементом $y \in B_n$. Существует алгоритм, проверяющий $x \in H = \langle y \rangle$, $x, y \in B_n$ за $O(n^2 \log n \max(|x|, |y|))$*

Теорема 2. *Для любой подгруппы $H \in B_k$, $k \leqslant 4$, существует алгоритм, дающий ответ на вопрос: разрешима ли проблема сопряженности в этой подгруппе.*

Литература

[1] I. Anshel, M. Anshel, B. Fisher, D. Goldfield. New Key Agreement Protocol in Braid Group Cryptography, Lecture Notes in Computer Science, v.2020, Springer Berlin, 2001

[2] K.H. Ko, S.J. Lee, J. H. Cheon, J. W. Han. New Public Key Cryptosystem Using Braid Groups, Proc. ASIACRYPT, 2000

[3] V. Shpilrain, A. Ushakov. The conjugacy search problem in public key cryptography: unnecessary and insufficient. Applicable Algebra in Engineering, Communication and Computing, to appear.

Градиентная статистическая атака на блоковые шифры

Б. Я. Рябко, В. А. Монарев, А. Н. Фионов, Ю. И. Шокин

Аннотация

Предлагается атака на блоковые шифры, основанная на выявлении отклонений от случайности в преобразованиях, проводимых на каждом цикле (раунде) шифрования. Показана возможность осуществления этой атаки по отношению к шифрам, для которых не известно более эффективных атак, чем перебор ключей. Представлены результаты экспериментов с шифром RC5. При практической реализации атака базируется на новых статистических тестах, недавно предложенных авторами.

1 Введение

Криптоанализу блоковых шифров посвящено множество исследований, причем новые результаты в этой области часто используются для улучшения конструкции шифров. Иногда сложность новой атаки (измеряемая объемом памяти и количеством операций, необходимыми для реализации атаки) может быть слишком большой для ее практического осуществления. Однако, если достигается даже относительно небольшое уменьшение сложности атаки по сравнению с известными методами, возникает мотивация для дальнейшего развития методов построения шифров. Так, линейный криптоанализ шифра DES (см. [1]) требует 2^{43} известных пар текст–шифротекст и обычно считается неосуществимым на практике. Тем не менее, он оказал существенное воздействие на принципы построения современных блоковых шифров, которые теперь устойчивы к такому виду атак. В данной работе мы предлагаем новую атаку на блоковые шифры, названную «градиентной статистической атакой». Мы показываем возможность проведения этой атаки на шифры, для которых не известно более эффективных атак, чем прямой перебор ключей.

Рассмотрим блоковый шифр с длиной блока n, длиной ключа s и функцией шифрования $E(x, K)$, где $x \in \{0,1\}^n$ обозначает блок текста, а $K \in \{0,1\}^s$ — секретный ключ. Обычно для современных блоковых шифров $n = 64$ или 128, $s = 128$ бит. Большинство блоковых шифров являются итерационными, т.е. содержат много раундов (циклов) преобразований, обычно обрамленных некоторым «прологом» и «эпилогом». Каждый из этих этапов, в свою очередь, может быть поделен на некоторое число более простых шагов. Вследствие итерационной структуры шифра секретный ключ K преобразуется в последовательность подключей (или раундовых ключей) $k_1, k_2, \ldots, k_t$, где t — число «простых шагов» в блоковом шифре. Обозначим через x_0 исходное состояние блока x, а через x_i — состояние после i-го шага. Таким образом, полное шифрование представляется как $x_t = E(x_0, K)$ и может быть записано в виде

$$x_1 = E_1(x_0, k_1), \quad \ldots, \quad x_t = E_t(x_{t-1}, k_t), \tag{1}$$

где E_i обозначает шифрующее преобразование на i-м шаге.

Пример. Рассмотрим шифр RC5 [2] с длиной блока 64 и количеством раундов r. Процесс шифрования

Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований, номер проекта 03-01-00495

сопоставляется с (1) следующим образом:

вход: (a, b)	$x_0 = (a, b)$
пролог:	
$a \leftarrow a + k_1$	$x_1 = E_1(x_0, k_1)$
$b \leftarrow b + k_2$	$x_2 = E_2(x_1, k_2)$
раунд 1:	
$a \leftarrow ((a \oplus b) \hookleftarrow b) + k_3$	$x_3 = E_3(x_2, k_3)$
$b \leftarrow ((b \oplus a) \hookleftarrow a) + k_4$	$x_4 = E_4(x_3, k_4)$
...	...
раунд r:	
$a \leftarrow ((a \oplus b) \hookleftarrow b) + k_{2r+1}$	
$b \leftarrow ((b \oplus a) \hookleftarrow a) + k_{2r+2}$	$x_t = E_t(x_{t-1}, k_t)$
выход: (a, b)	$x_t = (a, b)$

Здесь $t = 2r + 2$, длина каждого подключа 32 бита. Многие другие шифры, включая RC6 и AES, также могут быть описаны (1) с относительно небольшими, например, 32-битовыми подключами.

Мы предлагаем атаку по выбранному тексту для шифра, который может быть представлен схемой (1) с относительно небольшими подключами. Обозначим длины секретного ключа и каждого подключа соответственно через $|K|$ и $|k|$. Прямой перебор ключей требует $O(2^{|K|})$ операций (расшифровываем с $K = 0, 1, \ldots$ пока не получим известный x). Предлагаемая атака требует $O(mt2^{|k|})$ операций, где m — количество блоков шифротекста, достаточное для статистического анализа. Атака завершается нахождением правильных подключей (вместо самого K) при условии, что статистический тест способен обнаружить отклонение от случайности в последовательности из m блоков. Существенным моментом является то, что мы используем новые эффективные статистические тесты, недавно предложенные в [3, 4].

Экспериментальные исследования, проведенные с шифром RC5, демонстрируют практическую осуществимость предлагаемой атаки. В частности, шифр RC5, имеющий 8 раундов, может быть взломан при использовании 2^{33} пар текст–шифротекст.

2 Описание атаки

Предлагаемая атака относится к классу атак по выбранному тексту. В этих атаках криптоаналитик может подавать любую информацию на вход шифра и наблюдать соответствующий выход. Его цель — раскрыть секретный ключ или раундовые ключи. Предполагается, что блоковые шифры должны быть устойчивы по отношению к такого рода атакам.

Мы рассматриваем блоковый шифр, который может быть описан схемой (1). Заметим, что соответствующая (1) последовательность действий при дешифровании выглядит как

$$x_{t-1} = D_t(x_t, k_t), \quad \ldots, \quad x_0 = D_1(x_1, k_1), \tag{2}$$

где D_i обозначает дешифрующее преобразование, обратное по отношению к E_i.

Одно из требований к блоковым шифрам состоит в том, что имея на входе последовательность различных блоков, шифр должен выдавать на выходе последовательность бит, которая выглядит случайной. Истинно случайная последовательность может быть определена как последовательность, порожденная бернуллиевским источником с равными вероятностями нулей и единиц. Мы будем неформально называть последовательности «более случайными» или «менее случайными» в зависимости от того, как сильно они отличаются от истинно случайных последовательностей. Один способ измерения случайности состоит в использовании некоторой статистики, вычисляемой на основе последовательности и имеющей такое свойство, что менее случайные последовательности имеют большее значение статистики (с учетом некоторой вероятности ошибки в суждении). Это может быть хорошо известная статистика x^2, подчиняющаяся распределению χ^2. Обозначим такую статистику через $\gamma(x)$, где x — битовая последовательность.

Обозначим через $\alpha_1, \alpha_2, \ldots, \alpha_m$ последовательность входных блоков. Пусть все блоки заведомо неслучайны и попарно различны. Возможный пример может быть $\alpha_1 = 1, \alpha_2 = 2, \ldots, \alpha_m = m$, где числа записываются с помощью n-битовых слов. Применим один шаг шифрования к входной последовательности, обозначив результат через $\beta_1, \beta_2, \ldots, \beta_m$:

$$\beta_1 = E_1(\alpha_1, k_1), \quad \ldots, \quad \beta_m = E_1(\alpha_m, k_1).$$

Мы можем предположить, что последовательность β более случайна, чем α, т.е. $\gamma(\beta) < \gamma(\alpha)$. После второго шага шифрования последовательность

$$E_2(\beta_1, k_2), \quad E_2(\beta_2, k_2), \quad \ldots, \quad E_2(\beta_m, k_2)$$

более случайна, чем β и т.д. То есть каждый последующий шаг шифрования увеличивает степень случайности.

Отметим очевидное следствие: при дешифровании в соответствии с (2) случайность данных от шага к шагу уменьшается. Например, последовательность

$$D_1(\beta_1, k_1), \quad D_1(\beta_2, k_1), \quad \ldots, \quad D_1(\beta_m, k_1),$$

которая есть α, менее случайна, чем β. Но важно следующее: если подключ не верен, обозначим его через k_1', то последовательность

$$\alpha_1' = D_1(\beta_1, k_1'), \quad \ldots, \quad \alpha_m' = D_1(\beta_m, k_1')$$

будет *более* случайна, чем β, $\gamma(\alpha') < \gamma(\beta)$. Это происходит потому, что дешифрование с другим ключом соответствует дополнительному шифрованию с этим ключом, что составляет суть известного принципа многократного шифрования. Вообще говоря, дешифрование с неправильным раундовым ключом увеличивает случайность, в то время как дешифрование с правильным ключом уменьшает случайность. Эта разница может быть выявлена статистическим тестом.

Предлагаемая градиентная статистическая атака осуществляется следующим образом. Вначале шифруем последовательность $\alpha_1, \alpha_2, \ldots, \alpha_m$, определенную выше. Обозначим выходную последовательность через ω,

$$\omega_1 = E(\alpha_1, K), \quad \ldots, \quad \omega_m = E(\alpha_m, K).$$

(Напомним, что шифр состоит из t раундов или шагов и длина подключа на каждом шаге равна $|k|$.)

Теперь начинаем основную процедуру поиска ключа. Для всех $u \in \{0,1\}^{|k|}$ вычисляем последовательность

$$\Gamma_t(u) = D_t(\omega_1, u), \quad D_t(\omega_2, u), \quad \ldots, \quad D_t(\omega_m, u)$$

и оцениваем степень ее случайности, т.е. вычисляем $\gamma(\Gamma_t(u))$. Находим такое u^*, для которого $\gamma(\Gamma_t(u^*))$ максимальна. Полагаем, что неизвестный подключ $k_t = u^*$. Заметим, что количество операций на этой стадии пропорционально $m2^{|k|}$.

После этого на основе последовательности $\Gamma_t(k_t)$ повторяем аналогичные вычисления и находим подключ k_{t-1}. Используя $\Gamma_{t-1}(k_{t-1})$, находим k_{t-2} и т.д. до k_1. Общее число операций для раскрытия всех подключей пропорционально $mt2^{|k|}$.

3 Эксперименты с RC5

Экспериментальное исследование предложенной атаки проводилось следующим образом. Во-первых, была проанализирована степень случайности зашифрованных последовательностей как функция числа шагов шифрования. Цель была найти максимальное количество шагов, на которых тесты могли отличить зашифрованную последовательность от истинно случайной. Во-вторых, было проверено предположение, на котором базируется атака, а именно, увеличивает ли случайность последовательности дешифрование с неверным подключом, точнее, различимы ли с помощью тестов последовательности полученные при дешифровании с верным и неверными ключами. В-третьих, на основе полученных результатов была реализована непосредственно атака на шифр. Эксперименты проводились на многопроцессорной системе, содержащей 10 1-ГГц процессоров Alpha с 1 Гбайт памяти в каждом.

Таблица 1: Число последовательностей, признанных неслучайными

t	m	Число ключей	Число неслучайных выходов
10	2^{28}	30	30
11	2^{29}	22	10
12	2^{31}	6	6
13	2^{32}	6	6
14	2^{32}	6	5
15	2^{33}	3	3

Чтобы протестировать статистические свойства RC5, использовалась последовательность $\alpha_1\alpha_2\ldots\alpha_m$ при достаточно больших m с несколькими случайно выбранными ключами (табл. 1). Мы видим, что зашифрованная последовательность стабильно отличается от истинно случайной вплоть до 15-го шага (при уровне значимости 0.01), что соответствует 8-му раунду RC5.

Чтобы проверить различимость последовательностей, дешифрованных с верным и неверными подключами, мы использовали исходную последовательность длины $m = 2^{24}$, зашифрованную за 8 или 9 шагов при случайно выбранном ключе K и соответствующих подключах k_8 и k_9. В каждом случае выходная последовательность дешифровалась назад на один шаг с верным и пятью случайно выбранными неверными подключами u_1, ..., u_5. Все вычисления повторялись 10 раз. В табл. 2 показано количество случаев, в которых последовательность признавалась неслучайной. Мы видим, что результаты дешифрования с верным и неверными ключами надежно различаются (10 против 4 и 5 против 0).

Таблица 2: Число последовательностей, признанных неслучайными

	Верный ключ	u_1	u_2	u_3	u_4	u_5
$t = 8$	10	4	4	4	3	3
$t = 9$	5	0	0	0	0	0

Эксперименты подтвердили наши предположения о принципиальной возможности предлагаемой градиентной статистической атаки. На сегодняшний день посредством предложенной атаки взломан шифр RC5 с пятью раундами.

Литература

[1] Menzes A., van Oorschot P., Vanstone S. Handbook of Applied Cryptography. CRC Press, 1996.

[2] Rivest R. L. The RC5 encryption algorithm // B. Preneel, editor. Fast Software Encryption. Second International Workshop (LNCS 1008), P. 86–96. Springer-Verlag, 1995.

[3] Ryabko B. Ya., Stognienko V. S., Shokin Yu. I. A new test for randomness and its application to some cryptographic problems // Journal of Statistical Planning and Inference. V. 123, N. 2. 2003. P. 365–376.

[4] B. Ya. Ryabko, V. A. Monarev. Using information theory approach to randomness testing // Journal of Statistical Planning and Inference. V. 133, N 1. 2005. P. 95–110.

Верхние оценки средних вероятностей дифференциальных аппроксимаций булевых отображений

Л. В. Ковальчук

Вступление

В большинстве работ, посвященных исследованию стойкости блочных шифров относительно методов линейного и дифференциального криптоанализа, изучаются SPN-шифры или шифры Фейстеля, единственными нелинейными преобразованиями в которых являются s-блоки, а ключевой сумматор реализует операцию побитового булевого сложения двоичных векторов. В роботах [1]–[5] и других разработан и развит математический аппарат для оценки стойкости таких шифров к указанным методам криптоанализа.

Вместе с тем, некоторые современные шифры (например [6],[7]), имеют другой принцип построения; в частности, ключевой сумматор реализует операции сложения по модулям 2^{16} или 2^{32}. Известные методы оценки стойкости классических блочных шифров ([1]–[5]) оказываются, вообще говоря, не применимыми к анализу стойкости шифров, описанных в [6],[7].

В [8] введены новые числовые параметры s-блоков шифров Фейстеля типа ГОСТ 28147-89, в терминах которых получены аналитические выражения верхних оценок средних вероятностей дифференциальных и линейных характеристик шифра.

В данной работе получен ряд новых верхних границ средних вероятностей дифференциальных аппроксимаций отображений на множестве $\{0,1\}^m$, представляющих собой композицию ключевого сумматора, реализующего сложение по модулю 2^m, и блока подстановок (для различных вариантов задания групповых операций на области определения и множестве значений таких отображений).

1 Оценки средних вероятностей дифференциальных аппроксимаций для композиции сумматора по модулю 2^m и блока подстановки

Далее будем использовать обозначения:

$$V_m = \{0,1\}^m,\ m \in \mathbb{N};$$

$$f_k(x) = \varphi(x+k),\ x,k \in V_m, \tag{1}$$

где под операцией сложения понимается сложение по модулю 2^m, а функция $\varphi : V_m \to V_m$ обладает следующим свойством:

$$\varphi(x_1, x_2) = 2^t \varphi_2(x_2) + \varphi_1(x_1), \tag{2}$$

где $x_2 \in V_t,\ x_1 \in V_{m-t},\ \varphi_1 : V_{m-t} \to V_{m-t},\ \varphi_2 : V_t \to V_t$ — биекции; сложение выполняется по модулю 2^m.

Введем в рассмотрение следующие величины:

$$d_{f_k}(\alpha,\beta) = 2^{-m} \sum_{x \in V_m} \delta\left(f_k(x \circ \alpha) \bullet f_k^*(x), \beta\right); \tag{3}$$

$$D_f(\alpha,\beta) = 2^{-m} \sum_{k \in V_m} d_{f_r}(\alpha,\beta), \tag{4}$$

где символ δ является символом Кронекера, под операциями «$\circ$» и «$\bullet$» понимаются некоторые групповые операции, определенные на V_m; $f_k^*(x)$ означает элемент, обратный к $f_k(x)$ относительно операции «$\bullet$».

Также будем использовать обозначения «$+$» и «$\oplus$», означающие, соответственно, операции сложения по модулю 2^l, где значение l будет ясно из контекста, и побитовое сложение по модулю 2.

Далее мы будем строить верхние оценки для $D_f(\alpha,\beta)$ и $\max\limits_{\alpha,\beta\neq 0} D_f(\alpha,\beta)$ при различном выборе операций «$\circ$» и «$\bullet$».

Теорема 1. *В наших обозначениях справедливы следующие неравенства:*

1) $D_f(\alpha,\beta) \leqslant W^{\varphi_2}(\alpha_2,\beta_2)\, D_{\varphi_1}(\alpha_1,\beta_1)$, *где «$\circ$» и «$\bullet$» – операции сложения по модулю* 2^m,

$$W^{\varphi_2}(\alpha_2,\beta_2) = 2^{-t} \max_{\eta,\nu\in V_1} \left\{ \sum_{x_2\in V_t} \delta\left(\varphi_2(x_2+\alpha_2+\nu) - \varphi_2(x_2) - \eta, \beta_2\right) \right\},$$
$$\alpha = (\alpha_2,\alpha_1),\ \beta = (\beta_2,\beta_1),\ \alpha_1,\beta_1 \in V_{m-t},\ \alpha_2,\beta_2 \in V_t;$$

2) $D_f(\alpha,\beta) \leqslant U^{\varphi_2}(\alpha_2,\beta_2)\, D_{\varphi_1}(\alpha_1,\beta_1)$, *где «$\circ$» – операция сложения по модулю* 2^m, *«$\bullet$» – операция сложения по модулю* 2*;*

$$U^{\varphi_2}(\alpha_2,\beta_2) = 2^{-t} \max_{\nu\in V_1} \left\{ \sum_{x_2\in V_t} \delta\left(\varphi_2(x_2+\alpha_2+\nu) oplus \varphi_2(x_2), \beta_2\right) \right\},$$
$$\alpha = (\alpha_2,\alpha_1),\ \beta = (\beta_2,\beta_1),\ \alpha_1,\beta_1 \in V_{m-t},\ \alpha_2,\beta_2 \in V_t;$$

3) $D_f(\alpha,\beta) \leqslant V^{\varphi_2}(\alpha_2,\beta_2)\, D_{\varphi_1}(\alpha_1,\beta_1)$, *где «$\circ$» – операция сложения по модулю* 2, *«$\bullet$» – операция сложения по модулю* 2^m*;*

$$V^{\varphi_2}(\alpha_2,\beta_2) = 2^{-2t} \max_{\eta,\mu,\nu\in V_1} \left\{ \sum_{x_2,k_2\in V_t} \delta(\varphi_2((x_2\oplus\alpha_2)+k_2+\mu) - \varphi_2(x_2+k_2+\nu) - \eta, \beta_2) \right\},$$
$$\alpha = (\alpha_2,\alpha_1),\ \beta = (\beta_2,\beta_1),\ \alpha_1,\beta_1 \in V_{m-t},\ \alpha_2,\beta_2 \in V_t;$$

4) $D_f(\alpha,\beta) \leqslant Y^{\varphi_2}(\alpha_2,\beta_2)\, D_{\varphi_1}(\alpha_1,\beta_1)$, *где «$\circ$» и «$\bullet$» – операции сложения по модулю* 2*;*

$$Y^{\varphi_2}(\alpha_2,\beta_2) = 2^{-2t} \max_{\eta,\nu\in V_1} \left\{ \sum_{x_2,k_2\in V_t} \delta(\varphi_2((x_2\oplus\alpha_2)+k_2+\nu) \oplus \varphi_2(x_2+k_2+\eta), \beta_2) \right\},$$
$$\alpha = (\alpha_2,\alpha_1),\ \beta = (\beta_2,\beta_1),\ \alpha_1,\beta_1 \in V_{m-t},\ \alpha_2,\beta_2 \in V_t.$$

Следствие 1. *Пусть* $m = pt$, $\alpha = (\alpha_p,\ldots,\alpha_1)$, $\beta = (\beta_p,\ldots,\beta_1)$, $\alpha_i,\beta_i \in V_t$,

$$\varphi(\alpha) = \sum_{i=1}^{p} 2^{(i-1)t} \varphi_i(\alpha_i), \tag{5}$$

где $\varphi_i : V_t \to V_t$ *– биекции,* $i = \overline{1,p}$, *сложение в (5) выполняется по модулю* 2^m. *Тогда в условиях п.п. 1–4 теоремы 1, выполнено, соответственно:*

1) $D_f(\alpha,\beta) \leqslant \prod\limits_{i=2}^{p} W^{\varphi_i}(\alpha_i,\beta_i)\, D_{\varphi_1}(\alpha_1,\beta_1)$,

2) $D_f(\alpha,\beta) \leqslant \prod_{i=2}^{p} U^{\varphi_i}(\alpha_i,\beta_i)\, D_{\varphi_1}(\alpha_1,\beta_1)$,

3) $D_f(\alpha,\beta) \leqslant \prod_{i=2}^{p} V^{\varphi_i}(\alpha_i,\beta_i)\, D_{\varphi_1}(\alpha_1,\beta_1)$,

4) $D_f(\alpha,\beta) \leqslant \prod_{i=2}^{p} Y^{\varphi_i}(\alpha_i,\beta_i)\, D_{\varphi_1}(\alpha_1,\beta_1)$.

В таблице 1 приведены результаты статистических оценок распределений вероятностей параметров $W^\varphi, U^\varphi, V^\varphi, Y^\varphi$ для $\varphi : V_4 \to V_4$ (как функций равновероятной подстановки φ на V_4).

Таблица 1: Результаты статистической оценки распределения параметров $W^\varphi, U^\varphi, V^\varphi, Y^\varphi$ (для 10^4 подстановок φ на V_4)

Интервал для значения параметра	Количество подстановок для W^φ	Количество подстановок для U^φ	Количество подстановок для V^φ	Количество подстановок для Y^φ
0.00–0.05	0	0	0	0
0.05–0.10	0	0	0	0
0.10–0.15	24	0	784	0
0.15–0.20	3899	225	7075	325
0.20–0.25	4650	5627	1851	5998
0.25–0.30	0	0	12	0
0.30–0.35	1196	1360	245	1065
0.35–0.40	200	2423	29	2274
0.40–0.45	28	20	3	5
0.45–0.50	3	310	1	301
0.50–0.55	0	0	0	0
0.55–0.60	0	0	0	0
0.60–0.65	0	30	0	28
0.65–0.70	0	0	0	0
0.70–0.75	0	5	0	4
0.75–0.80	0	0	0	0
0.80–0.85	0	0	0	0
0.85–0.90	0	0	0	0
0.90–0.95	0	0	0	0
0.95–1.00	0	0	0	0

2 Оценки средних вероятностей дифференциальных аппроксимаций в схеме Фейстеля

Пусть $f_k(x,y) = (y, x \oplus \varphi(y+k))$, где $x, y, k \in V_m$, φ обладает свойством (2). На множестве V_{2m} введем следующие операции:

$$v \circ u = (v^L \oplus u^L, v^R + u^R),$$

$$v \bullet u = (v^L + u^L, v^R \oplus u^R),$$

где $v = (v^L, v^R)$, $u = (u^L, u^R)$, $v^L, v^R, u^L, u^R \in V_m$, «+» означает сложение по модулю 2^m, «$\oplus$» — сложение по модулю 2.

Как и в предыдущем разделе, будем рассматривать величины

$$d_{f_k}(\alpha,\beta) = 2^{-2m} \sum_{x \in V_{2m}} \delta\left(f_k(x \circ \alpha) \bullet f_k^*(x), \beta\right); \tag{6}$$

$$D_f(\alpha,\beta) = 2^{-m} \sum_{k \in V_m} d_{f_k}(\alpha,\beta). \tag{7}$$

Лемма 1. $D_f(\alpha,\beta) = d_{f_k}(\alpha,\beta) = \delta(\alpha^R,\beta^L)\, d_\varphi(\alpha^R,\beta^R - \alpha^L)$, $\forall k \in V_m$, *где* $d_\varphi(a,b) = 2^{-m} \sum\limits_{x\in V_m} \delta\left(\varphi(x+a)-\varphi(x),b\right)$ *и не зависит от* k, $a,b \in V_m$.

Теорема 2. $d_\varphi(a,b) \leqslant \Delta^{\varphi_2}(a_2,b_2)\, d_{\varphi_1}(a_1,b_1)$, *где*

$$\Delta^{\varphi_2}(a_2,b_2) = 2^{-t} \max_{\eta,\nu\in V_1} \left\{ \sum_{x_2\in V_t} \delta\left(\varphi_2(x_2+a_2+\nu)-\varphi_2(x_2)-\eta,b_2\right)\right\},$$

$a=(a_2,a_1)$, $b=(b_2,b_1)$, $a_1,b_1 \in V_{m-t}$, $a_2,b_2 \in V_t$.

Литература

[1] Biham E, Shamir A. Differential cryptanalysis of DES-like cryptosystems // Journal of Cryptology. — 1991. — V. 4. — № 1. — P. 3 — 72.

[2] Matsui M. Linear cryptanalysis methods for DES cipher // Advances in Cryptology — EUROCRYPT'93, Proceedings. — Springer Verlag, 1994. — P. 386 — 397.

[3] Knudsen L.R. Practically secure Feistel cipher // Fast Software Encryption. — FSE'94, Proceedings. — Springer Verlag, 1994. — P. 211 — 221.

[4] Kanda M. Practical security evaluation against differential and linear cryptanalysis for Feistel ciphers with SPN round function // Selected Areas in Cryptography. — SAC 2000, Proceedings. — Springer Verlag, 2001. — P. 324 — 338.

[5] Vaudenay S. Decorrelation: a theory for block cipher security // J. of Cryptology. — 2003. — V. 16. — № 4. — P. 249 — 286.

[6] Gosudarstvennyi Standart 28147-89. Cryptographic Protection for Data Processing Systems. Government Committee of the USSR for Standarts, 1989.

[7] Lai X., Massey J.L., Murphy S. Markov ciphers and differential cryptanalysis // Advances in Cryptology — EUROCRYPT'91, Proceedings. — Springer Verlag, 1991. — P. 17 — 38.

[8] А. Алексейчук, Л. Ковальчук. Линейный и дифференциальный криптоанализ шифров, содержащих сумматор по модулю 2^m // Международная конференция «Современные проблемы и новые течения в теории вероятности», Черновцы, 19 - 26 июня 2005 г., с. 9-10.

О конструкциях эндоморфных совершенных шифров

С. С. Коновалова, С. С. Титов

По теореме Шеннона совершенные шифры – это шифры обобщенного гаммирования со случайной равновероятной гаммой, и только они [1, 2]. Поэтому актуальна задача изучения конструкций, порождающих семейства таких абсолютно стойких шифров.

Работа посвящена решению проблем построения эндоморфных совершенных шифров, как классических, так и имитостойких, обобщающих теорему Шеннона для других видов криптоатак [1, 2, 3, 4, 5, 6]. О классических линейных эндоморфных совершенных шифрах, описанных в 1987 году западными криптологами [5], в книге [4] были поставлены следующие задачи (определения см. ниже и в [4]):

1. Является ли шифр конструкции 1 мультипликативным шифром?
2. Является ли любой совершенный билинейный шифр мультипликативным шифром?
3. Является ли любой совершенный линейный шифр билинейным шифром?

В [7] эти задачи решены, а именно дан положительный ответ на первый и отрицательный на второй и третий вопросы. Ниже, в первой части работы, приводится краткое изложение этих результатов на основе теории конечных плоскостей.

Во второй части работы исследуются имитостойкие совершенные шифры, а именно – $U(L)$ и $O(L)$-стойкие шифры [4]. Как показано в [4], основную проблему представляет построение эндоморфных $U(L)$ и $O(L)$-стойких шифров. Представлены конструкции $U(2)$, $U(3)$- и $O(2)$, $O(3)$-стойких шифров на основе конечных плоскостей и аналогов дробно-линейных функций и показана связь между ними.

Будем придерживаться понятий, терминологии и методов книги [4]. Отметим только, что матрица зашифрования в конструкции 1 – ганкелева матрица: см. [6], с. 218 и [3, 8].

Введем некоторые понятия: правило зашифрования совершенного по Шеннону шифра задается уравнением $y = x * k$, где y — зашифрованный текст, x — открытый текст, k — ключ зашифрования, $*$ — умножение в соответствующей квазигруппе. Множества X открытых текстов и Y закрытых текстов (шифрвеличин) рассматриваются в этой работе как подмножества векторных пространств над конечным полем F, F^r — пространство векторов-строк длины $r \in \mathbb{N}$ над полем F.

Естественно было бы считать шифр линейным над полем F, если $X = F^m$, $Y = F^n$ $(m, n \in \mathbb{N})$ и для каждого k операция зашифрования линейна по x. Однако линейных в таком понимании совершенных шифров не существует (см. стр. 66-68 в [4]), но можно построить линейный над F совершенный шифр, изменив в определении условия: $X = F^m \setminus \{0\}$, $Y = F^n \setminus \{0\}$ и для каждого $k \neq 0$ операция зашифрования линейна по x. Далее под линейным шифром будем понимать шифр, удовлетворяющий указанным условиям. Для такого шифра правило зашифрования можно задать матрицей M_k размеров $m \times n$. Линейный над F шифр является сильно совершенным [3] тогда и только тогда, когда выполняются условия:

1. Для любых x, $y \in F^m \setminus \{0\}$ существует (и единственный) ключ $k \in K$, удовлетворяющий условию $y = xM_k$
2. Распределение ключа $P(K)$ равномерно.

Линейный шифр назовем билинейным над F, если $X = F^m \setminus \{0\}$, $Y = F^n \setminus \{0\}$, $K = F^s \setminus \{0\}$ для некоторых m, n, $s \in \mathbb{N}$; $x \in X$ и каждый элемент матрицы M_k линеен по k.

Часто рассматривают три конструкции [4], позволяющие строить совершенные шифры. Пусть $\overrightarrow{x} = (x_1, \ldots, x_m)$, $\overrightarrow{k} = (k_1, \ldots, k_m)$ — ненулевые элементы поля $GF(q^m)$, представленные в координатной форме.

Конструкция 1. Шаг 1. Пусть $\overrightarrow{k} = (k_1, \ldots, k_m) \neq 0$ — начальный вектор линейной рекуррентной последовательности максимального периода над полем $F = GF(q)$.

Шаг 2. Пользуясь законом рекурсии, выразим каждый из следующих $m-1$ знаков $k_{m+1}, k_{m+2}, \ldots, k_{2m-1}$ ЛРП в виде линейных комбинаций переменных $k_1, \ldots, k_m$.

Шаг 3. В качестве i-й строки матрицы M_k возьмем вектор $(k_i, \ldots, k_{i+m-1})$, каждая координата k_j, $j > m$ которого записана в виде (полученном на этапе 2) линейной комбинации переменных $k_1, \ldots, k_m$.

Конструкция 2. Определим правило зашифрования в соответствии с соотношением $y = x \cdot k$ в поле $\mathrm{GF}(q^m)$.

Конструкция 3. Определим правило зашифрования в соответствии с соотношением $y' = x' \cdot k'$ в поле $GF(q^m)$. Здесь $x' = xA$, $k' = kB$, $y' = yC$, а A, B, C — невырожденные матрицы $m \times m$ над $GF(q)$.

Определение 1. Конструкция 3 дает билинейный над F минимальный сильно совершенный *мультипликативный* шифр.

1 Решение трех задач о трех конструкциях совершенных шифров

Задача 1. Докажем, что сильно совершенный билинейный шифр, построенный с помощью конструкции 1, является мультипликативным.

Пусть уравнение зашифрования билинейного шифра представлено в виде $y = xM_k$, где $\overrightarrow{y} = (y_0, y_1, \ldots, y_{n-1})$, $\overrightarrow{x} = (x_0, x_1, \ldots, x_{n-1})$ — векторы, $M = M_k$ — квадратная матрица $n \times n$, построенная при помощи конструкции 1.

В соответствии с [8] и шестой главой в [3], введём сопровождающую матрицу S многочлена $f(x) = a_0 + a_1 x^1 + \ldots + a_{n-1}x^{n-1} + x^n$ над полем F в виде

$$S = \begin{pmatrix} 0 & \ldots & 0 & -a_0 \\ 1 & \ldots & 0 & -a_1 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \ldots & 1 & -a_{n-1} \end{pmatrix}$$

Получаем, что любой ключевой вектор $(k_{0+i}, \ldots, k_{n-1+i})$ связан с вектором $(k_{1+i}, \ldots, k_{n+i})$ равенством $(k_{1+i}, \ldots, k_{n+i}) = (k_{0+i}, \ldots, k_{n-1+i})S$. Поэтому матрица M_k составлена из строк вида k, kS, kS^2,..., kS^{n-1} где k — ключевая вектор-строка $\overrightarrow{k} = (k_0, k_1, \ldots, k_{n-1})$.

Отсюда вытекает, что матрица M_k невырождена для любого $k \neq 0$. Поскольку многочлен f примитивен, то матрица S порождает поле $GF(2^n)$ матричных многочленов от S, а её степени порождают мультипликативную группу этого поля, которая является циклической. Следовательно, для любого $k \neq 0$ можно определить такую степень m, что $k = k_o S^m$, где $k_0 = \overrightarrow{k_o} = (1, 0, \ldots, 0)$ — «первоначальная» ключевая строка, как это отмечено в [4] вслед за [9]. Поэтому матрица M_k составлена из строк вида $k_0 S^m$, $k_0 S^{m+1}$, $k_0 S^{m+2}$,..., где $m = m(k)$ в силу этой таблицы соответствия степеней m и ключевых строк k. Векторное пространство всех n-битовых строк x можно изоморфно вложить в векторное пространство степеней S по первой строке: $\overrightarrow{x} = \overrightarrow{e} S^\ell$, где $\overrightarrow{e} = \overrightarrow{k_0}$. Такое же вложение можно произвести и для векторного пространства векторов k и y: $\overrightarrow{k} = \overrightarrow{e} S^m \Leftrightarrow M(\overrightarrow{k}) = M_e S^m$, $\overrightarrow{y} = \overrightarrow{e} S^s$, $\varphi(S^s) = \overrightarrow{y}$, $\varphi(S^m) = \overrightarrow{k}$, где M_e – это матрица M_k с вектором $\overrightarrow{e}$ в первой строке. Если $y = x * k$, то $\overrightarrow{y} = \overrightarrow{x} M(\overrightarrow{k})$ при $\overrightarrow{x} = \overrightarrow{e} S^\ell$ и $M_k = M_e S^m$, $\overrightarrow{y} = \overrightarrow{e} S^s$, т.е. при $\overrightarrow{x} = \varphi(S^\ell)$ и $\overrightarrow{k} = \varphi(S^m)$, получаем $\overrightarrow{y} = \overrightarrow{e} S^\ell M_e S^m$, $\overrightarrow{y} = \overrightarrow{e} S^{t+m}$, и, поскольку φ – изоморфизм пространств, найдется такая степень t, что $\overrightarrow{e} S^\ell M_e = \overrightarrow{e} S^t$. Положив $\overrightarrow{u} = \overrightarrow{e} S^t$, получим $\varphi(S^t) = \overrightarrow{u} \Leftrightarrow \overrightarrow{u} = \overrightarrow{x} M_e$, и если рассматривать представление элементов поля $GF(2^n)$ как векторов – первых строк степеней матрицы S, то получается, что $y = u \cdot k$ в $GF(2^m)$, т.е. $S^s = S^t S^m$, где $u = xM_e$, что доказывает мультипликативность шифра, поскольку умножение на M_e – линейное преобразование A степеней S^ℓ в пространство степеней S^t, и $x * k = \varphi(S^\ell) * \varphi(S^m) = \varphi(S^t \cdot S^m) = \varphi((S^\ell)A \cdot S^m)$. Итак, доказана

Теорема 1. *Конструкция 1 задает мультипликативный шифр вида* $y = u \cdot k$, $u = xM_e$.

Удобно рассмотреть рекурренту $k_{i+3} = k_{i+2} + k_i$ [4] в качестве примера. Пусть ЛРП задается примитивным многочленом $f(x) = x^3 + x^2 + 1$. Матрица зашифрования имеет следующий вид:

$$M_k = \begin{pmatrix} k_1 & k_2 & k_3 \\ k_2 & k_3 & k_1 + k_3 \\ k_3 & k_1 + k_3 & k_1 + k_2 + k_3 \end{pmatrix}$$

Сопровождающая матрица и связь ее степеней с векторами $\overrightarrow{k}$ и $\overrightarrow{x}$:

$$S = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix} \quad M_k = \begin{pmatrix} kE \\ kS \\ kS^2 \end{pmatrix} \quad k_0 = (1,0,0) \quad M_k = \begin{pmatrix} k_0 S^m \\ k_0 S^{m+1} \\ k_0 S^{m+2} \end{pmatrix}$$

$$S^2 = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix} \quad S^3 = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{pmatrix} \quad S^4 = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}$$

$$S^5 = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \end{pmatrix} \quad S^6 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix} \quad S^7 = E = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$\overrightarrow{e} S^0 M_e = \overrightarrow{e} E M_e = \overrightarrow{e} M_e = \overrightarrow{e} = \overrightarrow{e} S^0$, так что $(S^0)A = S^0$; $\overrightarrow{e} S^6 M_e = \overrightarrow{e} S^1$, и поэтому $(S^6)A = S^1$; $\overrightarrow{e} S^1 M_e = \overrightarrow{e} S^2$, так что $(S^1)A = S^2$, а поскольку разложение по базису – это $S^2 = S^6 + S^1$ (за базис взят S^0, S^6, S^1), то $(S^1)A = S^6 + S^1$; поэтому матрица A линейного преобразования $S^\ell \longmapsto S^t$ (т.е. $\overrightarrow{x} \longmapsto \overrightarrow{u}$) имеет вид

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} = M_e$$

k	m
001	1
010	6
011	2
100	0
101	5
110	4
111	3

x	ℓ
001	1
010	6
011	2
100	0
101	5
110	4
111	3

x	u
001	011
010	001
011	010
100	100
101	111
110	101
111	110

Легко проверить, что таблица Кэли и таблица умножения в поле степеней S совпадают.

Замечание. Построенная изотопия приводит конструкцию 1 к мультипликативному шифру не только для примитивного, но и для любого неприводимого многочлена.

Действительно, из анализа вышеприведённого доказательства ясно, что примитивность многочлена не столь существенна, важно лишь наличие таблицы перекодирующего изоморфизма, а это очевидно имеет место для любого неприводимого многочлена.

В качестве примера удобно рассмотреть многочлен $f(x) = x^4 + x^3 + x^2 + x + 1$.

В силу его непримитивности сопровождающая матрица многочлена будет иметь всего пять, а не $2^4 - 1 = 15$ различных ненулевых степеней, поэтому с их помощью нельзя получить все 15 ключевых вектора $\overrightarrow{k}$ и $\overrightarrow{x}$. Но векторное пространство всех 4-битовых строк x можно изоморфно вложить в линейное пространство матриц $4 * 4$, задавая вложение формулой $\varphi(x) = a + bS + cS^2 + dS^3$, где a, b, c, d – элементы поля $\mathbb{Z}_2$. Тогда первой строкой этих матриц будет вектор $\overrightarrow{x}$. Ниже представлена таблица, в которой приведено соответствие комбинаций элементов a, b, c, d с первой строкой (вектором $\overrightarrow{x}$) соответствующей матрицы $\varphi(x)$:

abcd	x
0001	0110
0010	0011
0011	0101

0100	0001
0101	0111
0110	0010
0111	0100

1000	1000
1001	1110
1010	1011
1011	1101

1100	1001
1101	1111
1110	1010
1111	1100

Итак, действительно, каждый из 15-ти векторов $\vec{x}$ кодируется одним набором чисел a, b, c, d, причем таблицы умножения в поле и в квазигруппе соответствуют друг другу в силу нашей изотопии, задаваемой матрицей

$$M_e = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

Задача 2. Для решения воспользуемся простым наблюдением и теоремой Алберта.

Наблюдение 1. Набор матриц M_k приводит к совершенному шифру тогда и только тогда, когда $\det[M(k') - M(k'')] \neq 0$ для любых различных ненулевых k' и k''.

Матрица M_k является изоморфным образом вектора k. Если этот определитель будет равен нулю, то, хотя $k' \neq k''$, будет существовать такой ненулевой вектор $\vec{x}$, что $\vec{x}M(k') = \vec{x}M(k'')$, а это не соответствует определению совершенного шифра.

Теорема (Алберт). *Если квазигруппа с единицей изотопна группе, то она ей изоморфна (и, следовательно, ассоциативна). См., например, [12, 15].*

По результатам наблюдения 1 линейный совершенный шифр является частным случаем системы Веблена-Веддербёрна [11], и поэтому решением второй задачи является любая такая система, обладающая свойством двусторонней дистрибутивности и не сводящаяся к полю. Например, это полуполе Алберта и полуполе Дональда Кнута (см. [11]).

Возьмем пятимерное полуполе Дональда Кнута (для него характерна двусторонняя дистрибутивность), которое задается неприводимым многочленом $f(x) = x^5 + x^2 + 1$, так что $x_0 = 1$, x, x^2, x^3, x^4 — базис $GF(2^5)$, и следующей таблицей умножения базисных элементов:

1	x	x^2	x^3	x^4
x	x^2	x^3	x^4	$x+1$
x^2	x^3	x^4	x^2+1	$x^4+x^3+x^2+x$
x^3	x^4	x^2+1	x^3+x	x^4+x^2+x
x^4	$x+1$	$x^4+x^3+x^2+x$	x^4+x^2+x	x^3+x^2

Матрица зашифрования:

$$M_k = \begin{pmatrix} k_1 & k_2 & k_3 & k_4 & k_5 \\ k_2 & k_3 & k_4 & k_5 & k_3+k_1 \\ k_3 & k_4 & k_5 & k_3+k_1 & k_4+k_2 \\ k_4 & k_5 & k_3+k_1 & k_4+k_2 & k_5+k_3 \\ k_5 & k_3+k_1 & k_4+k_2 & k_5+k_3 & k_4+k_3+k_1 \end{pmatrix}$$

По теореме Алберта эти полуполя дают примеры билинейных немультипликативных шифров ввиду их неассоциативности: $A = x(x^2x^2) = x+1, B = (xx^2)x^2 = x^2+1 \Rightarrow A \neq B$ [11, 7]. Мультипликативный же шифр задается квазигруппой, изотопной мультипликативной группе поля (изотопия задается линейными преобразованиями A, B, C [4]). Системы Веблена-Веддербёрна, не сводящиеся к полям, дают недезарговы конечные плоскости [11]. Такие системы существуют, примеры см. выше; значит, доказана

Теорема 2. *Существуют билинейные немультипликативные совершенные шифры.*

Задача 3. Докажем, что не любой совершенный линейный шифр является билинейным шифром. Для этого достаточно подвергнуть ключ k нелинейному обратимому преобразованию, которых над конечным полем гораздо больше, чем линейных [10].

Например $y = x \cdot k^h$, $h \neq 1$, $h \in \mathbb{Z}$. При $h = -1$ происходит не шифрование, а расшифровывание текста (явление парастрофии). Над полем характеристики два можно брать только нечетные h. Очевидно, что при условии взаимной простоты h с $2^m - 1$ возможно построение шифра, который будет линейным, но не билинейным [7]. Итак, справедлива

Теорема 3. *Любая нелинейная по ключу k изотопия мультипликативной группы поля дает пример линейного, но не билинейного совершенного шифра.*

Решением третьей задачи является и система Холла [11], преимущество которой перед рассмотренной выше (где отсутствует левая единица) в том, что в ней есть двусторонняя единица. Пусть $f(x) = (x^2 - rx - s)$ – многочлен второго порядка, неприводимый над полем F. Матрица зашифрования, соответствующая системе Холла, имеет вид [7]:

$$M_k = \begin{pmatrix} k_0 & k_1 \\ \frac{-k_0^2 + rk_0 + s}{k_1} & -k_0 + r \end{pmatrix} \text{ для } k_1 \neq 0, \quad M(k_0, 0) = k_0 E \text{ для } k_1 = 0,$$

где ключ $k = \overrightarrow{k} = (k_0, k_1)$.

Характеристическим многочленом такой матрицы является исходный $\det(\lambda E - M) = \chi_m(\lambda) = f(\lambda) = \lambda^2 - r\lambda - s \neq 0$ т.к. $f(x)$ неприводим. Проверим теперь в соответствии с наблюдением 1 определитель разности недиагональных матриц для разных k:

$$M(x, y) = \begin{pmatrix} x & y \\ \frac{-x^2 + rx + s}{y} & -x + r \end{pmatrix}, \qquad M(u, v) = \begin{pmatrix} u & v \\ \frac{-u^2 + ru + s}{v} & -u + r \end{pmatrix},$$

$$\det[M(x, y) - M(u, v)] = \det \begin{pmatrix} x - u & y - v \\ \frac{-x^2 + rx + s}{y} - \frac{-u^2 + ru + s}{v} & -x + u \end{pmatrix}$$
$$= \frac{-(xv - yu)^2 + r[(xv - yu)(v - y)] + s(v - y)^2}{yv} \neq 0,$$

т.к. иначе $z = \dfrac{xv - yu}{v - y}$ был бы корнем уравнения $z^2 - rz - s = 0$, принадлежащим полю $GF(q)$, однако по условию многочлен $f(z) = z^2 - rz - s$ неприводим над этим полем. Значит доказана

Теорема 4. *Система Холла дает пример двумерного над данным полем линейного, но не билинейного совершенного шифра на основе квазигруппы с двусторонней единицей.*

Воспользуемся теперь выявленной [7] связью совершенных шифров [4] с конечными плоскостями [10, 11] для изучения имитостойких совершенных шифров [4].

2 Конструкции совершенных шифров, стойких к другим видам криптоатак

Перейдем теперь к рассмотрению $U(L)$ и $O(L)$-стойких шифров, а также перпендикулярных $PA_1(L, \lambda, \mu)$ и циклических перпендикулярных массивов $CPA_1(L, \lambda, \mu)$, поскольку построение таких шифров эквивалентно построению перпендикулярных массивов [4].

Определение 2. $U(L)$-стойкий шифр – это шифр, стойкий к атакам на основе неупорядоченной L-кратной совокупности шифртекстов, полученных на одном ключе.

Задача построения $U(L)$-стойких шифров эквивалентна задаче построения перпендикулярных массивов специального вида.

Определение 3. Перпендикулярный массив $PA_\omega(t, \lambda, \mu)$ – это матрица A размеров $\omega \cdot C_\mu^t \times \lambda$ с элементами из множества Y мощности μ, каждая строка которой состоит из λ различных элементов, и любые t различных элементов множества Y содержатся точно в ω строках подматрицы, составленной любыми t столбцами матрицы A.

Далее будем рассматривать только минимальные эндоморфные шифры, т.е. шифры, строящиеся на основе массивов с $\lambda = \mu$, т.к. по теореме 4.2.4 в [4] изучение $U(L)$-стойких шифров с $\lambda < \mu$ и с минимальным числом ключей π сводится к случаю, когда $\lambda = \mu$ (см. об $O(L)$ с. 152 в [4]).

Теорема (А.Ю.Зубов [4], теорема 4.2.8)). *Если существует перпендикулярный массив* $PA_\omega(t,\lambda,\mu)$, *то есть и* $U(t)$*-стойкий шифр с параметрами* $|X|=\lambda$, $|Y|=\mu$, $|K|=\omega\cdot C_\mu^t$.

Определение 4. Циклический перпендикулярный массив $CPA_\omega(t,\lambda,\mu)$ – это перпендикулярный массив $PA_\omega(t,\lambda,\mu)$, который с каждой строкой содержит в качестве строк все ее циклические сдвиги [4].

Построение циклического $U(2)$-стойкого шифра сводится к задаче расстановки ферзей, не угрожающих друг другу на цилиндрической шахматной доске $n\times n$ [14]. Из результатов [14] вытекает, в частности, несуществование $CPA_1(2,9,9)$. Рассмотрением массивов, где зашифрование дается формулой $y=i\cdot x+j$ с ключом $k=(i,j)$, показано [4], что существует массив $CPA_1(2,5,5)=CPA_1(3,5,5)$.

Определение 5. $O(L)$-стойкий шифр – это шифр, стойкий к атакам на основе упорядоченной L-кратной совокупности шифртекстов, полученных на одном ключе. В этом случае в таблице зашифрования $A_\omega(L,\lambda,\mu)$ при $\omega=1$ любая строка должна содержать только по одному вектору из любых L ее элементов, содержащихся в любых L столбцах. Как и в случае $U(L)$-стойких шифров, для создания $O(L)$-стойких шифров строится матрица с $\lambda=\mu$ (см. стр. 152 в [4]). Для минимальных значений параметров эндоморфного $O(L)$-стойкого шифра имеем: $\lambda=\mu$, $\pi=(\lambda!)/((\lambda-L)!)$ [4]. Отметим, что построение $O(L)$-стойкого шифра приводит к схеме разделения секрета $(L,2L-1)$ [6]. Больше возможностей для создания $U(L)$ и $O(L)$-стойких шифров предоставляют нециклические массивы.

В следующих рассуждениях устанавливается связь с первой частью данной работы:

Наблюдение 2. Построение эндоморфного $O(2)$-стойкого шифра сводится к построению конечной (аффинной) плоскости: любым двум x будут однозначно соответствовать пара y, что определяет на плоскости две точки, через которые можно провести единственную прямую, и наоборот – любые две непараллельные прямые пересекаются в единственной точке.

Более подробное обоснование представляется излишним и фактически повторяет вводные выкладки в соответствующих разделах классических текстов [10, 11, 13].

Итак, с каждым линейным совершенным шифром можно связать эндоморфный $O(2)$-стойкий шифр по формуле $\vec{y}=\vec{x}\,M_{\vec{k}}+\vec{\ell}$, где $M_{\vec{k}}$ – матрица зашифрования линейного шифра на ключе $\vec{k}$. Такой шифр естественно тоже назвать линейным.

Массивы, построенные по системам Веблена-Веддербёрна, не сводящимся к полям (см. выше), дают примеры линейных $O(2)$-стойких шифров с интересными свойствами.

Наблюдение 3. Набор матриц M_k приводит к $U(2)$-стойкому шифру тогда и только тогда, когда $\det[M(k')\pm M(k'')]\neq 0$ для любых различных ненулевых k' и k''.

Если в наборе матриц будут существовать матрицы $M(k'')$ и $M(k')$, причем $M(k'')=-M(k')$, то $\det[M(k')+(-M(k'))]=0$, что противоречит наблюдению 1.

Связь между $U(2)$-стойкими и $O(2)$-стойкими шифрами достаточно проста:

Теорема 5. *Эндоморфный* $U(2)$*-стойкий шифр с уравнением зашифрования* $\vec{y}=\vec{x}\,M_{\vec{k}}+\vec{\ell}$, *где* $M_{\vec{k}}$ *– набор матриц, может быть дополнен до линейного* $O(2)$*-стойкого шифра; обратно, если с каждой матрицей* M_k *в* $O(2)$*-стойком шифре содержится и матрица* $-M_k$, *то одну матрицу из любой такой пары (любую!) можно убрать и получить линейный* $U(2)$*-стойкий шифр.*

Действительно, для построения $U(2)$-шифра необходим набор матриц M_k. Если к нему добавить набор матриц $-M_k$, то получившийся массив будет являться $O(2)$-стойким шифром, т.к. $\det[\pm M(k')-(\pm M(k''))]=\det[\pm(M(k')\pm M(k''))]=\pm\det[M(k')\pm M(k'')]\neq 0$ (исходя из наблюдения 3). Обратный вывод фактически повторяет рассуждения книги [4] на с. 92-96.

Отметим, что ограничение в обратном утверждении является существенным. Контрпример – аффинная плоскость системы Холла для примитивного над полем $GF(3)$ многочлена $f(x)=x^2+x+2$. Нетрудно убедиться, что из соответствующего линейного, но не билинейного $O(2)$-стойкого шифра невозможно выделить $U(2)$-стойкий шифр, и здесь не для каждой матрицы зашифрования M_k наличествует матрица зашифрования $(-M_k)$.

В [4] на с. 91-92 упоминается о существовании $PA_1(3,8,8)$. Его можно реализовать в аффинной плоскости по формуле $\vec{y}=\vec{x}\cdot\vec{k}+\vec{\ell}$, где умножение – в поле $GF(8)$ по таблице степеней многочлена $f(x)=x^3+x+1$ или $f(x)=x^3+x^2+1$. Искомый массив является как $U(3)$-, так и $O(2)$-стойким шифром.

Вопрос о возможности выделения $U(3)$-стойкого шифра из $O(3)$-стойкого шифра требует отдельного детального изучения.

Семейства $O(3)$-стойких шифров можно строить как массивы зашифрования на основе дробно-линейных функций вида $f(x) = (ax + b)/(cx + d)$, добавляя к ее элементам ∞ и определяя значения функции для следующих значений x: $f(\infty) = a/c$, $f(-d/c) = \infty$. В книгах [4] (с. 149), [9] указано, что проективная линейная группа $\mathrm{PGL}(2, \lambda)$ точно 3-транзитивна, и поэтому может быть использована для построения $O(3)$-стойкого шифра. Представление этой группы в виде дробно-линейных преобразований кажется нам достаточно естественным и геометрически ясным [10].

Для получения новых $O(3)$-стойких шифров следует отказаться от групповой структуры множества ключей (см. рекомендацию на с.151 в [4]). Кроме дробно-линейных преобразований конечного поля, что равносильно использованию точно 3-транзитивной группы $\mathrm{PGL}(2, \lambda)$, можно предложить обобщённые дробно-линейные функции, не образующие группу, определённые через тернарную операцию произвольной конечной аффинной плоскости (см. [11, 13]): $y = x \cdot m \circ b$, если точка (x,y) лежит на прямой из F_m (семейство параллельных прямых), проходящей через точку $(0,b)$. Эта операция определена для любых x, m и b, выбранных из множества $\lambda = \mu$ элементов. Умножение xm и сложение $x + b$ определяются как частные случаи тернарной операции следующим образом: $xm = x \cdot m \circ 0$, $x + b = x \cdot 1 \circ b$. Для тернарной операции постулируются пять свойств [11, 13], равносильных, как известно, заданию конечной плоскости. Дробно-линейную функцию по данной тернарной операции естественно задать следующим образом: $f(x) = (x \cdot a \circ b)/(x \cdot c \circ d)$. Деление происходит в квазигруппе ненулевых элементов с операцией умножения: для $c = 0$ $f(x) = (x \cdot a \circ b)/d = x \cdot a' \circ b'$, $f(\infty) = \infty$; для $c \neq 0$ достаточно взять $c = 1$ и $f(x) = (x \cdot a' \circ b')/(x \cdot 1 \circ d') = (x \cdot a' \circ b')/(x + d')$, $f(\infty) = a'$, $f(x_0) = \infty$ при $x_0 \cdot 1 \circ d' = x_0 + d' = 0$. Этот элемент x_0 – единственный. В самом деле, легко удостовериться, используя постулаты тернарной операции, что операция сложения определяет квазигруппу на множестве всех элементов, а операция умножения определяет квазигруппу на множестве ненулевых элементов, поэтому и правое и левое деление однозначно выполнимы.

Однако описанная выше конструкция не всегда приводит к $O(3)$-стойкому шифру; например, не всегда приводит к ней система Холла. Свойство быть $O(3)$-шифром геометрически означает наличие в плоскости проекций специального вида [10]. Необходимо, чтобы график линейной или дробно-линейной функции проходил через любые три данные точки. Проективная линейная группа $\mathrm{PGL}(2, \lambda)$ обладает такими проекциями в силу конфигурации Паппа, наличие которой равносильно коммутативности умножения в поле $GF(\lambda)$ [10]. Тем не менее эта конструкция даёт результат не только для конечных полей:

Теорема 6. *В коммутативном полуполе перечень линейных и дробно-линейных функций образует таблицу зашифрования эндоморфного $O(3)$-стойкого шифра.*

Эта теорема доказывается путем прямых выкладок, причем они оказываются справедливыми не только для поля, но и для коммутативного полуполя. Схема и идея доказательства: если не существует линейной функции, проходящей через три данные точки, то для коэффициентов проходящей через них дробно-линейной функции получается система линейных уравнений, определитель которой имеет геометрический смысл «площади» этого треугольника и не равен нулю в силу вычисления, которое можно провести, не используя ассоциативность, т.к. один из его столбцов состоит из единиц этого полуполя.

3 Заключение

Привлечение подходов геометрической алгебры позволило решить ряд задач конструирования совершенных шифров, как классических, так и имитостойких, и расширить многообразие известных шифров. Развитие таких алгебраических методов позволит перейти к исследованию более имитостойких совершенных шифров.

Авторы благодарят В.В. Ященко, М.М. Глухова и А.А. Махнёва за внимание к работе.

Литература

[1] Алферов А.П., Зубов А.Ю., Кузьмин А.С., Черемушкин А.В. Основы криптографии. – М.: Гелиос АРВ, 2001. 480 с.

[2] Шеннон К. Работы по теории информации и кибернетике. – М.: ИЛ, 1963. 830 с.

[3] Бабаш А.В., Шанкин Г.П. Криптография. (Серия «Аспекты защиты») Под ред. Шерстюка В.П., Применко Э.А. – М.: СОЛОН-Р, 2002. – 512 с.

[4] Зубов А.Ю. Совершенные шифры. — М.: Гелиос АРВ, 2003. – 160 с.

[5] Massey J., Maurer U., Wang M. Non-expanding, key minimal, robustly-perfect, linear and bilinear ciphers. – Proceedings of Crypto'87; Advances in Cryptology, 1987. – p. 237-247.

[6] Харин Ю.С., Берник В.И., Матвеев Г.В., Агиевич С.В. Математические и компьютерные основы криптологии. – Мн.: Новое знание, 2003. 382 с.

[7] Коновалова С.С., Титов С.С. Три задачи о трех конструкциях совершенных шифров. – Проблемы теоретической и прикладной математики. Труды 36-й Региональной молодежной конференции. – Екатеринбург: УрО РАН, 2005. – С. 37-41.

[8] Гантмахер Ф.Р. Теория матриц. – М.: Наука, 1967. – 576 с.

[9] Глухов М.М., Елизаров В.П., Нечаев А.А. Алгебра (учебник). – М.: в/ч 33965, 1990.

[10] Артин Э. Геометрическая алгебра. – М.: Наука, 1969. – 284 с.

[11] Холл М. Комбинаторика. – М.: Мир, 1970. – 424 с.

[12] Белоусов В.Д. Основы теории квазигрупп и луп. М.: Наука, 1967. 223 с.

[13] Скорняков Л.А. Проективные плоскости. УМН. 6:6 (46), 1951. С. 112-154.

[14] Гребенщикова Н.В., Корепанова Н.В., Русина И.С., Титов С.С. Варианты расстановки ферзей на цилиндрической доске. – Молодые ученые – транспорту: Труды IV научно-технической конференции. – Екатеринбург: УрГУПС, 2003. – С. 359-363.

[15] Белоусов В.Д., Белявская Г.Б. Латинские квадраты, квазигруппы и их приложения. Кишинев: Штиница, 1989. 80 с.

Тестирование генераторов псевдослучайных последовательностей на основе MTD-моделей

Ю. С. Харин, А. Н. Ярмола

1 Введение

Одной из важных проблем криптографической защиты информации является статистическое тестирование псевдослучайных последовательностей $x_t \in \mathcal{A} = \{0, 1, \ldots, N-1\}$, $t \in \mathbf{N}$ [1, 2]. Статистический тест – это решающее правило, позволяющее по наблюдаемой реализации $x_1, \ldots, x_T \in \mathcal{A}$ длительностью T с заданной точностью принять гипотезу H_0 ($\{x_t\}$ – равномерно распределенная случайная последовательность (РРСП), т.е. символы $x_1, x_2, \ldots$ независимы в совокупности и равномерно распределены на $\mathcal{A}$) или принять альтернативу H_1. Проведенный в [3] обзор существующих статистических тестов показывает: 1) многие из известных тестов ориентированы на проверку лишь одного из вероятностных свойств, характеризующих РРСП; 2) многие тесты построены «эвристически» и не фиксируют семейство альтернатив; 3) многие тесты не имеют оценок мощности. Поэтому актуальными являются задачи разработки адекватных вероятностных моделей для описания отклонений H_1 от модели РРСП и построения алгоритмов статистического анализа для обнаружения и оценивания таких отклонений.

Настоящий доклад посвящен решению этих задач применительно к отклонениям H_1 от модели РРСП, характеризующимся наличием стохастических зависимостей высокого порядка в $\{x_t\}$. Для этой цели используется малопараметрическая модель временных рядов – MTD-модель Рафтери [4].

2 MTD-модели и их свойства

Пусть $\{x_t \in \mathcal{A} : t \in \mathbf{N}\}$ – однородная цепь Маркова s-ого порядка, $s < \infty$, с пространством состояний $\mathcal{A} = \{0, \ldots, N-1\}$ мощности $2 \leqslant N < \infty$, определенная на вероятностном пространстве $(\Omega, \mathcal{F}, \mathbf{P})$, с некоторой $(s+1)$-мерной матрицей вероятностей переходов $P = (p_{i_0,\ldots,i_s})$, $p_{i_0,\ldots,i_s} = \mathbf{P}\{x_t = i_s \mid x_{t-1} = i_{s-1}, \ldots, x_{t-s} = i_0\}$, $i_0, \ldots, i_s \in \mathcal{A}$, $t > s$. MTD-модель, предложенная А. Рафтери [4] в 1985 году для «малопараметрического» описания цепей Маркова высокого порядка с дискретным временем, задает специальный вид матрицы P:

$$p_{i_0,\ldots,i_s} = \sum_{j=0}^{s-1} \lambda_j q_{i_j i_s}, \; i_0, \ldots, i_s \in \mathcal{A}, \tag{1}$$

где $Q = (q_{ik})$ - стохастическая $(N \times N)$-матрица, $i, k \in \mathcal{A}$; $\lambda = (\lambda_0, \ldots, \lambda_{s-1})'$ – s-вектор, $\lambda_0 + \ldots + \lambda_{s-1} = 1$, $\lambda_0 > 0$, $\lambda_j \geqslant 0$, $j = 1, \ldots, s-1$. Важным обобщением MTD-модели является MTDg-модель, в которой для каждого из s прошлых моментов времени используется «своя» матрица вероятностей переходов [5]:

$$p_{i_0,\ldots,i_s} = \sum_{j=0}^{s-1} \lambda_j q^{(j)}_{i_j i_s}, \; i_0, \ldots, i_s \in \mathcal{A}, \tag{2}$$

где $Q^{(j)} = (q^{(j)}_{ik})$, $j \in \{0, \ldots, s-1\}$ – стохастическая $(N \times N)$-матрица, соответствующая лагу $s - j$.

Исследуем вероятностные свойства MTD-, MTDg-моделей. Установим вначале условия эргодичности [6] для MTDg-модели.

Лемма 1. *Если $Q^{(0)}$ обладает свойством эргодичности, то MTDg-модель (2) является эргодической.*

Лемма 2. *Для того, чтобы MTD-модель (1) была эргодической, необходимо и достаточно, чтобы матрица Q удовлетворяла условию эргодичности.*

Обозначим: $\pi^{(t)} = (\pi_0^{(t)}, \ldots, \pi_{N-1}^{(t)})'$ – одномерное распределение вероятностей в момент $t \in \mathbf{N}$, $\pi_i^{(t)} = \mathbf{P}\{x_t = i\}$, $i \in \mathcal{A}$; $\Pi^{(t)} = \left(\pi_{i_1,\ldots,i_s}^{(t)}\right)$ – s-мерное распределение вероятностей вектора $X_t = (x_{t-(s-1)}, \ldots, x_t)' \in \mathcal{A}^s$, $\pi_{i_1,\ldots,i_s}^{(t)} = \mathbf{P}\{x_{t-(s-1)} = i_1, \ldots, x_t = i_s\}$, $i_1, \ldots, i_s \in \mathcal{A}$; $\Pi^* = \left(\pi_{i_1,\ldots,i_s}^*\right)$, $i_1, \ldots, i_s \in \mathcal{A}$ – s-мерное стационарное распределение вероятностей цепи Маркова [6]; $\pi^* = (\pi_0^*, \ldots, \pi_{N-1}^*)'$ – одномерное стационарное распределение вероятностей.

Теорема 1. *Если $\{x_t\}$ - дискретный временной ряд, соответствующий MTDg-модели, то его одномерные распределения вероятностей $\{\pi^{(t)}\}$ связаны линейным соотношением:*

$$\pi^{(t)} = \sum_{j=0}^{s-1} \lambda_j \left(Q^{(j)}\right)' \pi^{(t-s+j)}, \ t > s.$$

Теорема 2. *Если для цепи Маркова порядка s имеют место линейные соотношения между векторами распределений вероятностей:*

$$\pi^{(t)} = \sum_{j=0}^{s-1} \left(A^{(j)}\right)' \pi^{(t-s+j)}, \ t > s,$$

где $A^{(j)} = (a_{ik}^j)$, $a_{ik}^{(j)} \geqslant 0$, $i, k \in \mathcal{A}$, $j = 0, \ldots, s-1$, то существуют s-вектор $\lambda = (\lambda_0, \ldots, \lambda_{s-1})'$, $\lambda_j \geqslant 0$, $\lambda_0 + \ldots + \lambda_{s-1} = 1$, и стохастические $(N \times N)$ - матрицы $Q^{(j)}$, $j = 0, \ldots, s-1$, такие что матрица вероятностей переходов P, представима в виде (2).

Лемма 3. *Для MTDg-модели распределение вероятностей $\Pi^{(t)}$ s-вектора X_t при $t \geqslant 2s$ имеет вид ($i_1, \ldots, i_s \in \mathcal{A}$):*

$$\pi_{i_1,\ldots,i_s}^{(t)} = \prod_{l=0}^{s-1} \left(\sum_{j=l+1}^{s-1} \lambda_j q_{i_{j-l}, i_{s-l}}^{(j)} + \sum_{j=0}^{l} \lambda_j \sum_{r=0}^{N-1} q_{r i_{s-l}}^{(j)} \pi_r^{(t-l-s+j)} \right).$$

Теорема 3. *Если выполнены условия Леммы 1, то для стационарного распределения Π^* справедливо выражение ($i_1, \ldots, i_s \in \mathcal{A}$):*

$$\pi_{i_1,\ldots,i_s}^* = \prod_{l=0}^{s-1} \left(\pi_{i_{s-l}}^* + \sum_{j=l+1}^{s-1} \lambda_j \left(q_{i_{j-l} i_{s-l}}^{(j)} - \sum_{r=0}^{N-1} q_{r i_{s-l}}^{(j)} \pi_r^*. \right) \right).$$

В дальнейшем, для оценивания параметров MTD-модели потребуется следующее следствие.

Следствие 1. *Если выполнены условия Леммы 1, то для стационарного двумерного маргинального распределения $\Pi^*(m) = (\pi_{ik}^*(m))$ векторов $(x_{t-m}, x_t)'$, $1 \leqslant m \leqslant s$ в случае MTDg-модели справедливо линейное соотношение:*

$$\pi_{ki}^*(m) = \pi_k^* \pi_i^* + \pi_k^* \lambda_{s-m} \left(q_{ki}^{(s-m)} - \sum_{r=0}^{N-1} q_{ri}^{(s-m)} \pi_r^* \right), \ i, k \in \mathcal{A};$$

в частности, в случае MTD-модели:

$$\pi_{ki}^*(m) = \pi_k^* \pi_i^* + \pi_k^* \lambda_{s-m} \left(q_{ki} - \pi_i^* \right), \ i, k \in \mathcal{A}. \tag{3}$$

3 Оценивание параметров и проверка гипотез

Рассмотрим задачу статистического оценивания параметров MTD-модели (1). Пусть наблюдается реализация $X = (x_1, \ldots, x_T)$ длительности T дискретного временного ряда, соответствующего MTD-модели. Построим оценки параметров, основанные на свойстве (3) стационарных распределений. Определим статистики:

$$\hat{q}_{ki} = \left\{ \sum_{j=1}^{s} \hat{\pi}_{ki}(j) / \hat{\pi}_k - (s-1)\hat{\pi}_i, \text{ если } \hat{\pi}_k > 0; \ 1/N, \text{ иначе} \right\}, \tag{4}$$

где $\hat{\pi}_i = \sum_{t=s+1}^{T-s+1} \mathbf{I}\{x_t = i\}/(T-2s+1)$, $\hat{\pi}_{ki}(j) = \sum_{t=s+j}^{T-s+j} \mathbf{I}\{x_{t-j} = k\}\mathbf{I}\{x_t = i\}/(T-2s+1)$, $i,k \in \mathcal{A}$, $j = 1,\ldots,s$; $\mathbf{I}\{a\}$ – индикаторная функция;

$$\hat{\lambda} = \arg\min_{\lambda} \sum_{i,k\in\mathcal{A}} \sum_{j=0}^{s-1} (z_{ki}(j) - \lambda_j d_{ki})^2, \tag{5}$$

где $z_{ki}(j) = \hat{\pi}_{ki}(s-j)/\hat{\pi}_k - \hat{\pi}_i$, $i,k \in \mathcal{A}$, $j = 0,\ldots,s-1$; $d_{ki} = \hat{q}_{ki} - \hat{\pi}_i$, $i,k \in \mathcal{A}$.

Теорема 4. *Если имеет место MTD-модель и выполнены условия Леммы 2, то статистики (4), (5) при $T \to \infty$ являются асимптотически несмещенными и состоятельными оценками для Q и λ соответственно, причем при $T > 2s$ матрица $\hat{Q}$ – стохастическая.*

К сожалению, использование этого метода оценивания в случае MTDg-модели невозможно. Более того, справедлив следующий результат.

Теорема 5. *Для $m < s$ либо не существует набора параметров $\{\lambda, Q^{(0)},\ldots,Q^{(s-1)}\}$, такого, что для любых фиксированных $1 \leqslant j_1 < \ldots < j_m \leqslant s$ стационарные распределения векторов $(x_t, x_{t-j_1},\ldots,x_{t-j_m})'$ совпадают с заданными распределениями $\pi^*(j_1,\ldots,j_m)$, либо такой набор параметров не единственный.*

Оценки (4), (5) будем использовать как начальное приближение для построения оценок максимального правдоподобия (ОМП) $\tilde{Q}$, $\tilde{\lambda}$ параметров MTD-модели. Логарифмическая функция правдоподобия (ЛФП) параметров Q, λ имеет вид:

$$l(Q,\lambda) = \sum_{t=s+1}^{T} \ln\left(\sum_{j=0}^{s-1} \lambda_j q_{x_{t-s+j},x_t}\right). \tag{6}$$

Задача вычисления ОМП $\tilde{Q}$, $\tilde{\lambda}$ состоит в отыскании точки максимума ЛФП (6) при ограничениях на параметры модели. Для практического вычисления ОМП в [7] предложен итерационный алгоритм. Однако, предложенные там же [7] начальные значения не являются состоятельными оценками и ухудшают работу алгоритма, т.к. с увеличением T число итераций алгоритма, необходимых для достижения результата, не уменьшается и не гарантируется сходимость итерационного процесса к ОМП. Поэтому оценки (4), (5) целесообразно использовать в качестве более точных начальных значений итерационного алгоритма. Эффективность данного подхода подтверждена численными экспериментами.

С помощью ОМП $\tilde{Q}$, $\tilde{\lambda}$ удается построить решающее правило для распознавания РРСП. Гипотеза $H_0 = \{x_t$ является РРСП$\}$ имеет в рамках модели (1) эквивалентное представление: $H_0 = \{q_{ki} = 1/N,\ i,k \in \mathcal{A}\}$. Построен тест проверки гипотез H_0, $H_1 = \bar{H}_0$, основанный на статистике обобщенного отношения правдоподобия $\lambda_T(X) = 2(l(\tilde{Q},\tilde{\lambda}) + T\ln N)$ и имеющий асимптотический (при $T \to \infty$) размер $\varepsilon \in (0,1)$:

$$d = d(X) = \{0,\ \lambda_T(X) < \Delta_\varepsilon;\ 1,\ \lambda_T(X) \geqslant \Delta_\varepsilon\},$$

где Δ_ε – квантиль уровня ε хи-квадрат распределения с $N(N-1)$ степенями свободы.

4 Численные результаты

Численные эксперименты проводились на выходной последовательности псевдослучайного генератора A5/1 [8]. В качестве модели использовалась MTD-модель при $N = 2$, $s = 10$, уровень значимости $\varepsilon = 0.05$, длительность наблюдений $T = 1048576$. Результаты экспериментов представлены в Таблице 1.

Литература

[1] Алферов А.П., Зубов А.Ю. Кузьмин А.С., Черемушкин А.В. Основы криптографии. М.: Гелиос, 2001.

[2] Зубков А.М. Датчики псевдослучайных чисел и их применения // Московский университет и развитие криптографии в России. Москва: МГУ, 2002, с. 200-206.

Таблица 1: Результаты тестирования

Начальные значения регистров A5/1	$\lambda_T(X)$	$d(X)$
$R_1 = 1$, $R_2 = 2$, $R_3 = 3$	3.43	0
$R_1 = 67BA$, $R_2 = 395AB$, $R_3 = BEBE8$	6.48	1
$R_1 = 67BA$, $R_2 = BEBE8$, $R_3 = 395AB$	0.95	0

[3] Харин Ю.С., Берник В.И., Матвеев Г.В., Агиевич С.В. Математические и компьютерные основы криптологии. Минск: Новое знание, 2003.

[4] Raftery A. E. A model for high-order Markov chains // J. R. Statist. Soc. 1985, Vol. 47, No. 3, p. 528-539.

[5] Raftery A. E. A new model for discrete-valued time series: autocorrelations and extensions // Rassegna di Metodi Statistici ed Applicazioni. 1985, Vol 3-4, p. 149-162.

[6] Боровков А.А. Теория вероятностей. - М.:Наука, 1986.

[7] Berchtold A. Estimation of the Mixture Transition Distribution Model. // J. of Time Ser. Anal., Vol.22, No.4, 2001, p.379-397.

[8] Ekdahl P, Johansson T. Another Attack on A5/1 // Proceeding of IEEE International Symposium Information Theory (ISIT) 2001, Washington D.C., 2001.

Ассоциированные метрики и их применение для модификации криптосистемы Нидеррайтера

Э. М. Габидулин, М. А. Чурусова

1 Введение

1.1 Криптосистема Нидеррайтера

В 1978 году была предложена криптосистема Нидеррайтера [1]. Эта система является криптосистемой с открытым ключом и основана на линейных кодах. Дадим краткое описание этой системы.

В качестве *закрытого ключа* выбираются:

- Проверочная $(d-1) \times n$ матрица $\mathbf{H}$, некоторого обобщенного кода Рида-Соломона над полем $GF(q)$;
- Случайно выбранная невырожденная скремблирующая матрица $\mathbf{S}$ порядка $(d-1)$ над полем $GF(q)$. Назначение этой матрицы - скрыть видимые закономерности в структуре проверочной матрицы;
- Быстрый алгоритм декодирования обобщенного кода Рида-Соломона.

Открытым ключом является скремблированная проверочная $(d-1) \times n$ матрица $\mathbf{H}_{cr} = \mathbf{SH}$.

Открытый текст $\underline{m} = (m_1, m_2, \ldots, m_n)$ имеет длину n, а его элементы выбираются из поля $GF(q)$ и имеют хэммингов вес не выше $\frac{d-1}{2}$.

Шифротекст, соответствующий сообщению m, представляет собой (d-1)-вектор и вычисляется по формуле:

$$\underline{c} = \underline{m}\mathbf{H}_{cr}^T = \underline{m}\mathbf{H}^T\mathbf{S}^T. \tag{1}$$

Законный пользователь после приема шифротекста $\underline{c}$, умножает его справа на матрицу $(\mathbf{S}^T)^{-1}$, а затем применяет известный только ему алгоритм быстрого декодирования к вектору $\underline{m}\mathbf{H}^T$ и получает переданное сообщение $\underline{m}$.

1.2 Взлом системы Нидеррайтера

В 1992 году система Нидеррайтера была взломана Сидельниковым и Шестаковым [2]. Основная идея, использованный при взломе, заключался в следующем. Криптоаналитик подбирает такие матрицы $\widetilde{\mathbf{S}}$ и $\widetilde{\mathbf{H}}$, что выполняется:

$$\mathbf{H}_{cr} = \mathbf{SH} = \widetilde{\mathbf{S}}\widetilde{\mathbf{H}}. \tag{2}$$

Напомним, что противнику известен открытый ключ $\mathbf{H}_{cr}$, но не матрицы $\mathbf{S}$ и $\mathbf{H}$ по отдельности.

Матрица $\widetilde{\mathbf{H}}$ позволяет вскрыть систему в том и только в том случае, если обе матрицы $\mathbf{H}$ и $\widetilde{\mathbf{H}}$ являются проверочными матрицами одного и того же ОРС-кода.

Число вычислений для взлома в алгоритме Сидельникова-Шестакова составляет порядка $O(n^3)$. Это означает, что первоначальная система Нидеррайтера при любом разумном выборе открытого ключа полностью вскрывается.

1.3 Модификации системы Нидеррайтера

Были предложены два основных способа модификации криптосистемы Нидеррайтера.

Первый способ заключается в зашумлении проверочной матрицы кода введением скрывающей матрицы. Например, в работе [3] в качестве скрывающей матрицы была предложена матрица единичного ранга. В работе [4] использовались скрывающие матрицы ранга, значительно большего единицы.

Вторым способом является использование нехэмминговой метрики. Например в [6], выбиралась ранговая метрика. В рассматриваемых далее модификациях вводились как зашумляющие матрицы, так и новые метрики.

2 Ассоциированные метрики

2.1 Определение ассоциированной и родительской метрик

Рассмотрим некоторую прямоугольную $n \times N$ матрицу $\mathbf{F} = (\underline{f}_1, \underline{f}_2, \ldots, \underline{f}_N)$ с элементами из некоторого поля $GF(Q)$, причем $n \leqslant N$. Выберем матрицу F так, чтобы она была полного ранга n.

Пусть $GF(Q)^n$- n-мерное линейное пространство. Любой вектор $\underline{x} \in GF(Q)^n$ можно представить в виде:

$$\underline{x} = \sum_{i=1}^{N} \underline{a}_i \underline{f}_i, \tag{3}$$

причем в случае $n \leqslant N$ количество таких представлений равно $L = Q^{(N-n)}$.

Вектору $\underline{x}$ можно поставить во взаимно однозначное соответствие $L = Q^{(N-n)}$ наборов коэффициентов $\underline{a}_i$ из (3). Эти наборы можно в упорядоченном виде записать как матрицу коэффициентов:

$$\mathbf{A}(\underline{x}) = \begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1N} \\ a_{21} & a_{22} & \ldots & a_{2N} \\ \ldots & \ldots & \ldots & \ldots \\ a_{L1} & a_{L2} & \ldots & a_{LN} \end{pmatrix}. \tag{4}$$

Обозначим транспонированные вектор-строки матрицы $\mathbf{A}$ как $(\underline{a}_1^T, \underline{a}_2^T, \ldots, \underline{a}_p^T)$.

Зададим некоторую метрику и найдем нормы векторов $(\underline{a}_1^T, \underline{a}_2^T, \ldots, \underline{a}_p^T)$, назовем эту метрику *родительской*. Тогда каждому вектору $\underline{x}$ будет соответствовать некоторый набор чисел:

$$N_{par_1}(\underline{a}_1^T), N_{par_2}(\underline{a}_2^T), \ldots, N_{par_L}(\underline{a}_L^T). \tag{5}$$

Определение 1. Нормой в метрике, ассоциированной с матрицей $\mathbf{F}$ с элементами из некоторого линейного пространства $GF(Q)$ вектора $\underline{x} \in GF(Q)^n$, назовем минимальное из чисел $N_{par_1}, N_{par_2}, \ldots, N_{par_p}$ в (5):

$$N_F(\underline{x}) = \min(N_{par_1}, N_{par_2}, \ldots, N_{par_p}).$$

Введенная таким образом норма, ассоциированная с матрицей $\mathbf{F}$, удовлетворяет аксиомам нормы.

1. $N(\underline{x}) = 0$ тогда и только тогда, когда $\underline{x} = 0$.
2. $N(\underline{x}) > 0$ для любого вектора $\underline{x} \neq 0$.
3. $N(\underline{x}_1 + \underline{x}_2) \leqslant N(\underline{x}_1) + N(\underline{x}_2)$.

Определение 2. Расстоянием между векторами $\underline{x_1}$ и $\underline{x_2}$ в метрике, ассоциированной с матрицей $\mathbf{F}$, назовем норму их разности.

$$d_F(\underline{x_1}, \underline{x_2}) = N_F(\underline{x_1} - \underline{x_2}).$$

Рассмотрим примеры ассоциированных метрик с различными родительскими метриками.

2.2 Примеры

1. Пусть родительской метрикой является хэммингова метрика. В этом случае нормой N_F в ассоциированной метрике будет минимальное возможное число ненулевых a_i в (3).

2. Рассмотрим теперь ассоциированную метрику, родительской для которой является ранговая метрика.

   Пусть в принятых нами обозначениях $Q = q^m$. Будем рассматривать некоторое расширенное поле $GF(q^m)$. По определению нормой векторов $(\underline{a}_i^T \in GF(q^m), i = 1, \ldots, p$ в ранговой метрике является максимальное число координат, линейно независимых над $GF(q)$ (см. [6]). В этом случае можно записать, что:

$$N_F(x) = \min[\mathrm{rank}(\underline{a}_1^T), \mathrm{rank}(\underline{a}_2^T), \ldots, \mathrm{rank}(\underline{a}_p^T)]. \tag{6}$$

3 Модификация системы Нидеррайтера с помощью ассоциированной метрики

3.1 Условия, накладываемые на элементы выбранной ассоциированной метрики

Для того, чтобы перейти к модификации криптосистемы необходимо наложить дополнительные ограничения на матрицу $\mathbf{F}$. Во-первых, необходимо, чтобы $\mathbf{F}$ была проверочной матрицей кода, имеющего алгоритм быстрого декодирования в какой-либо метрике. При этом все синдромы $\underline{m}\mathbf{F}^T$ являются n-векторами, имеющими норму в ассоциированной метрике не превосходящую t_1.

Во-вторых, должен существовать сам код, исправляющий t_1 ошибок, в ассоциированной метрике также с быстрым алгоритмом декодирования.

Наконец, должен существовать код $\mathbf{G}$ в ассоциированной метрике, имеющий алгоритм быстрого декодирования и исправляющий не менее t_1 N_F-ошибок.

4 Примеры модификации системы Нидеррайтера с помощью ассоциированной метрики

Рассмотрим примеры криптосистем на линейных кодах, которые можно отнести к системам, построенным на ассоциированных метриках. Первым примером будет система на основе метрики Вандермонда [4]. Вторым – система на основе ранговой метрики [7].

4.1 Система на основе метрики Вандермонда

Рассмотрим ассоциированную метрику, у которой родительской является хэммингова метрика.

Криптосистема строится следующим образом: сначала легальный пользователь выбирается матрицу $\mathbf{F}$, столбцы которой задаю ассоциированную метрику. Родительский код, проверочная матрица которого равна $\mathbf{F}$, должен иметь алгоритм быстрого декодирования в родительской метрике (в данном случае это хэммингова метрика). Затем нужно выбрать транспонированную порождающую матрицу $\mathbf{G}^T$ некоторого линейного кода, обладающего алгоритмом быстрого декодирования в ассоциированной метрике. В рассматриваемой системе [4] выбирается матрица $\mathbf{F}$ размером $n \times N$ с элементами из $GF(q)$ имеющая вид матрицы Вандермонда. А также выбирается матрица $\mathbf{G}^T$ размером $n \times k$ с элементами из $GF(q)$ задающая код и также имеющая структуру подобную матрице Вандермонда.

Далее выбирается некоторая квадратная невырожденная матрица $\mathbf{S}$ порядка n, а также матрица перестановки $\mathbf{P}$ порядка N.

Закрытый ключ представляет собой набор матриц $\{\mathbf{F}, \mathbf{G}^T, \mathbf{S}, \mathbf{P}\}$.

Открытым ключом будет матрица:

$$\mathbf{H}_{pub} = \mathbf{S}(\mathbf{F} + \mathbf{G}^T\mathbf{U})\mathbf{P},$$

где $\mathbf{U}$ - некоторая случайная матрица размером $k \times N$. Кодовыми векторами будут столбцы матрицы $\mathbf{G}^T\mathbf{U}$.

Текст сообщения представляет собой N-мерный $\underline{m} = (m_1, m_2, \ldots, m_N)^T$ вектор с весом равным минимальному значению среди корректирующей способностью кода, задаваемого матрицей $\mathbf{G}^T$ и корректирующей способностью родительского кода.

Шифрование: шифротекст вычисляется как синдром

$$\underline{c} = \mathbf{H}_{pub}\underline{m} = \mathbf{S}(\underline{g} + \underline{e})$$

Расшифрование: легальный пользователь умножает полученный шифротекст на $\mathbf{S}^{-1}$, применяет алгоритм быстрого декодирования в ассоциированной метрике и получает векторы $\underline{g}$ и $\underline{e}$. Затем к вектору $\underline{e}$ применяется алгоритм быстрого декодирования родительского кода. Для получения открытого текста остается полученный вектор умножить на матрицу $\mathbf{P}^{-1}$.

4.2 Система на основе ранговой метрики

Пусть для принятых ранее обозначений справедливо $Q = (q^N)^n$.

Рассмотрим ассоциированную метрику, у которой родительской является ранговая метрика. Определим матрицу $\mathbf{F}$:

$$\mathbf{F} = \begin{pmatrix} f_1 & f_1^q & \ldots & f_1^{q^{n-1}} \\ f_2 & f_2^q & \ldots & f_2^{q^{n-1}} \\ \vdots & \vdots & \ldots & \vdots \\ f_{N_1} & f_{N_1}^q & \ldots & f_{N_1}^{q^{n-1}} \end{pmatrix} \tag{7}$$

Введенную таким образом метрику можно назвать *фробениусовской метрикой*, для краткости F-метрикой, так как она ассоциирована с немодифицированной матрицей Фробениуса.

Новая криптосистема основывается на F-метрике и ассоциирована с ранговыми кодами.

Построим код оптимальный в F-метрик. Рассмотрим конкатенацию матриц:

$$\mathbf{Q} = \begin{pmatrix} h_1 & h_1^q & \ldots & h_1^{q^{n-1}} \\ h_2 & h_2^q & \ldots & h_2^{q^{n-1}} \\ \vdots & \vdots & \ldots & \vdots \\ h_{N_1} & h_{N_1}^q & \ldots & h_{N_1}^{q^{n-1}} \\ g_1 & g_1^q & \ldots & g_1^{q^{n-1}} \\ g_2 & g_2^q & \ldots & g_2^{q^{n-1}} \\ \vdots & \vdots & \ldots & \vdots \\ g_k & g_k^q & \ldots & g_k^{q^{n-1}} \end{pmatrix} = \begin{pmatrix} \mathbf{F} \\ \mathbf{G_k} \end{pmatrix}, \tag{8}$$

где $\mathbf{F}$ - проверочная матрица рангового кода, а матрица $\mathbf{G_k}$ матрица фробениусовского типа:

$$\mathbf{G_k} = \begin{pmatrix} g_1 & g_1^q & \ldots & g_1^{q^{n-1}} \\ g_2 & g_2^q & \ldots & g_2^{q^{n-1}} \\ \vdots & \vdots & \vdots & \vdots \\ g_k & g_k^q & \ldots & g_k^{q^{n-1}} \end{pmatrix}.$$

Выберем следующие параметры:$k < n < N_1, N_1 + k = N$.

Элементы g_j, $j = 1, 2, \ldots, k$ выбираются так, чтобы элементы $h_1, h, \ldots, h_{N_1}, g_1, g, \ldots, g_k$ в совокупности были независимы над базовым полем.

Секретный ключ представляет собой набор матриц $(\mathbf{F}, \mathbf{G}_k, \mathbf{S}, \mathbf{P})$.

Открытый ключ представляет собой матрицу $\mathbf{H}_{pub} = \mathbf{P}(\mathbf{F} + \mathbf{U}\mathbf{G}_k)\mathbf{S}$, где $\mathbf{U}$ - некоторая случайная матрица.

Кодовыми векторами являются строки матрицы $\mathbf{U}\mathbf{G}_k\mathbf{S}$. Матрица $\mathbf{U}$ не нужна при расшифровании, но для криптоаналитика она должна быть недоступной.

Текст сообщения представляет собой N_1-мерный вектор $\underline{m} = (m_1, m_2, \ldots, m_{N_1})$ такой, что $d_H(\underline{m}) = t_{min} = min(t_k, t_p)$, где t_k - корректирующая способность кода, задаваемого матрицей $\mathbf{G}_k$ в пространстве с новой метрикой, t_p - корректирующая способность родительского кода.

Шифрование: шифротекст вычисляется как синдром:

$$\underline{c} = \underline{m}\mathbf{H}_{pub} = \underline{m}\mathbf{P}(\mathbf{U}\mathbf{G}_k + \mathbf{F})\mathbf{S} = \underline{\widetilde{m}}(\mathbf{F} + \mathbf{U}\mathbf{G}_k)\mathbf{S},$$
$$\underline{c} = \left(m_1(\mathbf{F}_1 + \mathbf{G}_{k_1}) + m_2(\mathbf{F}_2 + \mathbf{G}_{k_2}) + \ldots + m_{N_1}(\mathbf{F}_{N_1} + \mathbf{G}_{k_{N_1}})\right)\mathbf{S} = (\underline{g} + \underline{e})\mathbf{S},$$

где $\underline{\widetilde{m}} = \underline{m}\mathbf{P}$, F_i и G_i - строки матриц $\mathbf{F}$ и $\mathbf{UG}$ соответственно.

Расшифрование: легальный пользователь умножает полученный шифротекст $(\underline{g} + \underline{e})$ на $\mathbf{S}^{-1}$. Затем применяет алгоритм быстрого декодирования в новой метрике.

В результате пользователь получит вектора $\underline{g}$ и $\underline{e}$.

После применения алгоритма быстрого декодирования родительского кода легальный пользователь получит вектор $\underline{\widetilde{m}}$. Далее, при умножении $\underline{\widetilde{m}}$ на $\mathbf{P}^{-1}$ получится сам открытый текст $\underline{m}$.

Литература

[1] Niederreiter H., Knapsack-Type Cryptosystem and Algebraic Coding Theory // Probl. Control and Inform. Theory, 1986. - Vol. 15 - P. 19-34.

[2] Сидельников В.М., Шестаков С.О., О системе шифрования, основанной на обобщенных кодах Рида-Соломона // Дискретная математика, 1992. - Т. 4, № 3.

[3] Gabidulin E., Ourivski A., Pavlouchkov V. On the modified Niederreiter cryptosystem // Proc. Information Theory and Networking Workshop, Metsovo, Greece, 1999. P. 50

[4] Габидулин Э.М., Обернихин В.А., Коды в F-метрике Вандермонда и их применение.

[5] Gabidulin E.M., Simonis J., Metrics Generated by Families of Subspaces // IEEE Trans. Inform.

[6] Габидулин Э.М., Теория кодов с максимальным ранговым расстоянием // Проблемы передачи информации, Вып. 1, 1985 - Т. XXI.

[7] Churusova M., Gabidulin E.M. The modified Niederreiter cryptosystem based on new metric. Proc. of the 8th International Symposium on communication theory and applications, 17-22 July 2005, pp. 66-70. St.Martin's College, Ambleside, UK.

Наследственные признаки в конечных полугруппах

Н. В. Фомичёв

В [1] описан общий подход к постановке и решению задачи дифференциации по заданному признаку элементов конечной группы $\langle S\rangle$, порожденной при помощи системы образующих S. В качестве меры сложности порождения подмножества элементов группы, обладающих признаком, предложена кратчайшая из длин элементов данного подмножества в системе образующих S. Приложения данных исследований могут быть весьма обширными. В частности, в криптологии данный подход целесообразно использовать для исследования слабых в определенном смысле шифрующих подстановок, которые могут содержаться в семействе шифра.

В криптографических схемах используются композиции как групповых, так и полугрупповых отображений, в связи с этим актуальной является задача распространения и развития подхода, предложенного в [1], для дифференциации элементов конечных полугрупп. Доклад посвящен данной задаче, в решении ее принципиальное значение имеет описание наследственных признаков в конечных циклических полугруппах.

Пусть Φ — конечная полугруппа и $G = \langle S\rangle$ — ее подполугруппа, порожденная системой образующих $S = \{s_1, s_2, \ldots, s_p\}$, $S \subseteq \Phi$, p — натуральное число. Дадим некоторые определения, аналогичные известным определениям из теории групп [1, 2].

Определение 1. Для непустого подмножества $Q \subseteq \Phi$ его длиной (или длиной покрытия) в системе образующих S (обозначается $L(Q, S)$) называют наибольшую из длин всех элементов множества Q в системе образующих S, то есть

$$\mathrm{pok}_S Q = \max_{g \in Q} L(g, S),$$

где $L(g, S)$ — длина элемента g в системе образующих S.

Определение 2. Для непустого подмножества $Q \subseteq \Phi$ его показателем в системе образующих S (обозначается $\mathrm{pok}_S Q$) назовем наименьшую из длин всех элементов множества Q в системе образующих S, то есть

$$\mathrm{pok}_S Q = \min_{g \in Q} L(g, S).$$

Рассмотрим отношение частичного порядка на элементах полугруппы $G : g_1 \leqslant g_2 \Leftrightarrow \langle g_1\rangle \subseteq \langle g_2\rangle$, где $g_1, g_2 \in G$ и $\langle g\rangle$ — циклическая полугруппа, порожденная g.

Определение 3. Непустое подмножество Q полугруппы Φ, называется наследственным, если из включения $g_1 \in Q$ и отношения $g_1 \leqslant g_2$ следует, что $g_2 \in Q$.

Пусть $HR(G)$ — множество всех наследственных подмножеств полугруппы G, упорядоченное относительно теоретико-множественного включения. Множество $HR(G)$ является решеткой ([3], гл.1, §1), элементы которой суть либо одна циклическая подполугруппа полугруппы G, либо объединение нескольких циклических подполугрупп. Неразложимыми в сумму элементами решетки $HR(G)$ являются циклические подполугруппы полугруппы G.

Рассмотрим подмножества Q и H полугруппы Φ, где H — множество элементов, обладающих определенным признаком.

Определение 4. Множество Q имеет H-признак или во множестве Q имеется H-признак, если $Q \cap H \neq \varnothing$. Назовем H-признак тривиальным (нетривиальным), если $Q \cap H$ — одноэлементное множество (содержит более одного элемента). Множество Q не имеет H-признака или во множестве Q имеется пустой H-признак, если $Q \cap H = \varnothing$.

В частности, если Φ — моноид и Q, H суть его подмоноиды, то Q имеет, по меньшей мере, тривиальный признак, так как множество $Q \cap H$ содержит единичный элемент.

Определение 5. Если множество Q имеет H-признак, то показателем H-признака множества Q в системе образующих S назовем показатель множества $Q \cap H$ в системе образующих S, обозначим его $\mathrm{pok}_S(Q \cap H)$.

Показатель H-признака циклической полугруппы $\langle g \rangle$ обозначим через $\mathrm{pok}_g H$.

Определение 6. H-признак в полугруппе G называется наследственным (полугрупповым), если $G \cap H \in HR(G)$ (если $G \cap H$ — подполугруппа полугруппы G).

С использованием следствия 13 ([3], гл.1, §1) в [4] показано, что каждое наследственное подмножество Q полугруппы G (в частности, сама полугруппа G) единственным образом представимо в виде объединения максимальных в этом подмножестве циклических полугрупп, где циклическая полугруппа $\langle g \rangle$ максимальна в подмножестве Q, если $\langle g \rangle \in Q$ и $\langle g \rangle$ не является собственной подполугруппой полугруппы $\langle g' \rangle$ такой, что $\langle g' \rangle \in Q$. Следовательно, изучение наследственного признака H в полугруппе G, где $H \subset \Phi$, можно свести к изучению этого признака во всех максимальных циклических подполугруппах, образующих каноническое представление полугруппы G.

Пусть $g \in G$ и требуется описать наследственное множество $\langle g \rangle \cap H$. Произвольный элемент множества $\langle g \rangle \cap H$ имеет вид g^t, где $t_i \in \{1, \ldots, |\langle g \rangle|\}$, при этом если $g^t \in H$, то $\langle g^t \rangle \in H$. Следовательно, если полугруппа $\langle g \rangle$ имеет наследственный H-признак, то для описания наследственного множества $\langle g \rangle \cap H$ достаточно определить множество чисел $\{t_1, \ldots, t_r\} \subseteq \{1, \ldots, |\langle g \rangle|\}$ таких, что есть максимальная во множестве $\langle g \rangle \cap H$ циклическая полугруппа, $i = 1, \ldots, r$. Множество чисел $\{t_1, \ldots, t_r\}$, где t_i — наименьшее число с указанным свойством, $i = 1, \ldots, r$, называется множеством (H, g)-пороговых чисел и обозначается $\Pi(H, g)$. Таким образом,

$$\langle g \rangle \cap H = \bigcup_{t \in \Pi(H,g)} \langle g^t \rangle$$

Пусть копредставление циклической полугруппы есть пара $\langle g; g^{d+1} = g^{d+n+1} \rangle$, где d, n — натуральные, тогда $\langle g \rangle = \{g, g^2, \ldots, g^d, g^{d+1}, \ldots, g^{d+n}\}$. Для описания множества $\Pi(H, g)$ рассмотрим решетку чисел $N_g = \{1, 2, \ldots, |\langle g \rangle|\}$ по отношению ρ, определяемому правилом: $t \rho \tau \Leftrightarrow g^\tau \leqslant g^t$. Решетка (N_g, ρ) обладает следующими свойствами:

1) на подмножестве $\{1, \ldots, d\}$ отношение ρ есть отношение делимости;

2) если $\tau \in \{d+1, \ldots, n\}$, то для любого $t \in N_g$ отношение $t \rho \tau$ выполнено тогда и только тогда, когда (t, n) делит (τ, n).

Лемма 1. *Пусть φ — гомоморфизм решетки (N_g, ρ) на решетку чисел (M_g, ρ), где $\varphi(\tau)$ для $\tau \in N_g$ есть наименьшее число t из N_g такое, что $\langle g^\tau \rangle = \langle g^t \rangle$. Тогда*

$$\varphi(\tau) = \begin{cases} \tau, & \tau = 1, \ldots d, \\ d + 1 + (\tau - d - 1) \bmod r, & \tau = d+1, \ldots d+n, \end{cases}$$

где $r = (\tau, n)$.

Теорема 1. *Если копредставление циклической полугруппы есть $\langle g; g^{d+1} = g^{d+n+1} \rangle$, то*

$$\Pi(H, g) = prm\{t \in M_g : g^t \in H\},$$

где для множества натуральных чисел M через $prmM$ обозначено его подмножество минимальных по отношению делимости чисел.

Следствие 1. *Величина $pok_g H$ равна наименьшему из чисел множества $\Pi(H, g)$.*

Следствие 2. *Наследственный H-признак в циклической полугруппе $\langle g \rangle$ является полугрупповым тогда и только тогда, когда множество $\Pi(H, g)$ состоит из единственного числа μ, в частности, тривиальным при $\mu = \lceil \frac{d+1}{n} \rceil n$.*

Литература

[1] Н. Д. Подуфалов, В. М. Фомичев. Признаки элементов в конечных группах — М.: Доклады академии наук, т.404, №3, 2005, с.308–311.

[2] М. М. Глухов. О числовых параметрах, связанных с заданием конечных групп системами образующих элементов. В сб. «Труды по дискретной математике», т.1, М.: ТВП, 1997г., с.43–66.

[3] Г. Гретцер. Общая теория решеток. Пер. с англ./Под редакцией Д. М. Смирнова. – М.: Мир, 1981.

[4] Н. В. Фомичев. О признаках элементов конечных полугрупп. Седьмая международная научно-практическая конференция «Информационная безопасность», Таганрог, 2005 г.

О сложности поиска аннигиляторов низкой степени для булевых функций

В. В. Баев

Вопрос существования и задача отыскания аннигиляторов низкой степени для булевых функций стал весьма актуальным в свете работ [1] и [2]. Связанный с существованием аннигиляторов низкой степени показатель алгебраической иммунности является важным криптографическим свойством булевой функции.

Введём необходимые обозначения и определения. $\mathbb{F}_2$ — поле из двух элементов. $V_n = \mathbb{F}_2^n$ — линейное пространство наборов длины n с компонентами из поля $\mathbb{F}_2$. $\mathcal{F}_n$ — множество всех функций $\mathbb{F}_2^n \to \mathbb{F}_2$. Степенью функции $g \in \mathcal{F}_n$ будем называть максимальную степень монома в полиноме Жегалкина этой функции и будем обозначать $\deg g$. Функция $g \in \mathcal{F}_n$ называется аннигилятором функции $f \in \mathcal{F}_n$, если $f \cdot g = 0$. Обозначим

$$A_d(f) := \{g \in \mathcal{F}_n | f \cdot g = 0,\ \deg g \leqslant d\}.$$

В работе [2] приведён детерминированный алгоритм поиска аннигиляторов из $A_d(f)$ для функции $f \in \mathcal{F}_n$ за полиномиальное от 2^n время, а также вероятностный алгоритм с математическим ожиданием времени работы, полиномиально зависящим от n, при условии прямого доступа к памяти.

Интересной представляется задача нахождения аннигиляторов низкой степени за время, по порядку меньшее 2^n, с помощью детерминированных алгоритмов (например, с использованием машин Тьюринга в качестве вычислительной модели). Однако, при этом нужно будет определиться, как мы хотим задавать функции из $\mathcal{F}_n$ при подаче на вход алгоритма. Ведь если просто задать функцию набором её значений на всех 2^n векторах из V_n, то размер входа алгоритма будет порядка 2^n, чего хотелось бы избежать.

Будем считать, что функции из $\mathcal{F}_n$ параметризованы некоторым словом конечной длины в алфавите $\{0,1\}$. То есть для некоторого $Y_n \subset \{0,1\}^*$ задано отображение $\varphi_n : Y_n \to \mathcal{F}_n$. Теперь мы можем считать, что булева функция задана парой (n, y), где $n \in \mathbb{N},\ y \in Y_n$. Нас будут интересовать алгоритмы нахождения аннигиляторов низкой степени, сложность которых полиномиально зависит от длины входа — (n, y). Здесь стоит отметить, что мы можем использовать не любые параметризации φ_n, а только «разумные», например, вычислимые за полиномиальное от 2^n время. Мы будем использовать параметризации с более жёстким ограничением на φ_n: существует полиномиальный алгоритм со входом (n, y, x), где $n \in \mathbb{N},\ y \in Y_n,\ x \in V_n$, вычисляющий значение функции $\varphi_n(y) \in \mathcal{F}_n$ на векторе x.

Теорема 1 ([4]). *Пусть параметром y функции $f_y \in \mathcal{F}_n$ будет список всех мономов, входящих в полином Жегалкина этой функции. Тогда существует алгоритм, получающий на вход (n, d, y) и вычисляющий базис линейного пространства $A_d(f_y)$, причём сложность этого алгоритма есть $O(M_y \cdot (S_n^d)^3)$, где M_y — количество мономов в списке y, а $S_n^d = \sum_{k=0}^{d} C_n^k$.*

Утверждение 1. *Для произвольных $f_1, f_2 \in \mathcal{F}_n$ верны следующие соотношения линейных подпространств пространства $\mathcal{F}_n$:*

$$A_d(f_1) + A_d(f_2) \subset A_d(f_1 \cdot f_2),$$
$$A_d(f_1 + 1) + A_d(f_2 + 1) \subset A_d(f_1 \vee f_2 + 1),$$
$$A_d(f_1) \cap A_d(f_2) = A_d(f_1 \vee f_2),$$
$$A_d(f_1 + 1) \cap A_d(f_2 + 1) = A_d(f_1 \cdot f_2 + 1).$$

Пусть $x, \alpha \in V_n,\ x = (x_1, \ldots, x_n),\ \alpha = (\alpha_1, \ldots, \alpha_n)$. Введём следующее обозначение:

$$x^\alpha := \prod_{i=1}^{n} x_i^{\alpha_i},$$

где

$$x_i^{\alpha_i} := \begin{cases} x_i, & \alpha_i = 1; \\ 1, & \alpha_i = 0. \end{cases}$$

Обозначим также $B_{n,d} := \{f \in \mathcal{F}_n | \deg f \leqslant d\}$.

Теорема 2. *Пусть функция $f \in \mathcal{F}_n$ задана в виде ДНФ. Тогда существует алгоритм, получающий на вход эту ДНФ и вычисляющий базис линейного пространства $A_d(f)$, причём сложность этого алгоритма ограничена сверху полиномом от n и от длины ДНФ.*

Доказательство. Для любого $\alpha \in V_n$, используя алгоритм из теоремы 1, можно вычислить базис пространства $A_d(x^\alpha)$ со сложностью $O\left((S_n^d)^3\right)$. Для $\sigma \in V_n$ рассмотрим аффинное преобразование $\tau : x \mapsto x + \sigma$ линейного пространства V_n. Оно индуцирует изоморфизм пространств аннигиляторов $\varphi_\sigma : A_d(x^\alpha) \to A_d((x+\sigma)^\alpha)$, который задаётся формулой $\varphi_\sigma(g)(x) = g(x+\sigma)$. Зафиксируем базис в пространстве $B_{n,d}$, состоящий из всех мономов степени $\leqslant d$. В этом базисе линейное отображение φ_σ имеет матрицу размера $S_n^d \times S_n^d$. Столбец этой матрицы, отвечающий базисному моному x^β, можно получить, раскрыв скобки в произведении $(x+\sigma)^\beta$. Поскольку количество множителей в этом произведении $\leqslant d$, то сложность данной процедуры имеет порядок 2^d, то есть некоторая константа, не зависящая от n. Умножив матрицу отображения φ_σ на каждый базисный вектор из $A_d(x^\alpha)$, мы получим базис в $A_d((x+\sigma)^\alpha)$. Таким образом строится полиномиальный алгоритм вычисления базиса пространства $A_d((x+\sigma)^\alpha)$ для любых фиксированных $\alpha, \sigma \in V_n$.

Пусть функция представлена в виде ДНФ:

$$f(x) = \bigvee_{k=1}^{T} (x + \sigma^k)^{\alpha^k},$$

где $\sigma^k, \alpha^k \in V_n$ $(k = 1, \ldots, T)$. Тогда по утверждению 1

$$A_d(f) = \bigcap_{k=1}^{T} A_d\left((x+\sigma^k)^{\alpha^k}\right).$$

Алгоритмы вычисления базисов пространств $A_d\left((x+\sigma^k)^{\alpha^k}\right)$ получаются по приведённой выше схеме. По базисам этих пространств можно вычислить базис пересечения с полиномиальной от n и от T сложностью. □

Теорема 3. *При любом $d \geqslant 0$ задача вычисления базиса линейного пространства $A_d(f)$ для функции $f \in \mathcal{F}_n$, заданной в виде КНФ, является NP-трудной.*

Доказательство. Полиномиальная сводимость задачи вычисления базиса пространства $A_d(f)$ по заданной КНФ к NP-полной задаче о выполнимости этой КНФ напрямую следует из следующей цепочки равносильностей:

$$f = 0 \iff A_d(f) = B_{n,d} \iff \dim A_d(f) = S_n^d. \quad \square$$

Теперь рассмотрим ситуацию, когда функция задаётся формулой с использованием булевых операций $\neg, \&, \vee$. Использование отрицания будем в некоторых случаях заменять операцией «+1». Мы хотим искать аннигиляторы низкой степени рекурсивно по этой формуле. Для каждой подформулы, задающей функцию f', мы будем получать пару линейных пространств

$$G_d(f') \subset A_d(f'+1),\ H_d(f') \subset A_d(f'), \tag{1}$$

заданных своими базисными функциями. Каждую базисную функцию мы представляем вектором координат в базисе всех мономов степени $\leqslant d$. Длина этого вектора координат равна S_n^d, то есть полиномиальна от n.

В листьях дерева рекурсии стоят функции вида $f_i(x_1, \ldots, x_n) = x_i$. В этом случае мы можем полиномиально от n вычислить базисы в следующих линейных пространствах:

$$A_d(x_i + 1) = \{g \cdot x_i |\ g \in \mathcal{F}_n, x_i \text{ фиктивна для } g,\ \deg g \leqslant d-1\}$$
$$A_d(x_i) = \{g \cdot (x_i + 1) |\ g \in \mathcal{F}_n, x_i \text{ фиктивна для } g,\ \deg g \leqslant d-1\}$$

и положить $G_d(f_i) := A_d(f_i + 1)$, $H_d(f_i) := A_d(f_i)$.

Пусть подформула имеет вид $f' = f_1 + 1 = \neg f_1$, и для функции f_1 выполняется рекурсивное предположение (1). Тогда, если положить

$$H_d(f') := G_d(f_1),$$
$$G_d(f') := H_d(f_1),$$

то рекурсивное предположение (1) будет выполняться для функции f'.

Пусть теперь подформула имеет вид $f' = f_1 \cdot f_2$, и для функций f_1 и f_2 выполняется рекурсивное предположение (1). Положим $G_d(f') := G_d(f_1) \cap G_d(f_2)$, $H_d(f') := H_d(f_1) + H_d(f_2)$. Тогда в силу утверждения 1 и рекурсивного предположения (1) для функций f_1 и f_2

$$G_d(f') \subset A_d(f_1 + 1) \cap A_d(f_2 + 1) = A_d(f_1 \cdot f_2 + 1) = A_d(f' + 1),$$
$$H_d(f') \subset A_d(f_1) + A_d(f_2) \subset A_d(f_1 \cdot f_2) = A_d(f').$$

Пусть, наконец, подформула имеет вид $f' = f_1 \vee f_2$, и для функций f_1 и f_2 выполняется рекурсивное предположение (1). Положим $G_d(f') := G_d(f_1) + G_d(f_2)$, $H_d(f') := H_d(f_1) \cap H_d(f_2)$. Тогда в силу утверждения 1 и рекурсивного предположения (1) для функций f_1 и f_2

$$G_d(f') \subset A_d(f_1 + 1) + A_d(f_2 + 1) \subset A_d(f_1 \vee f_2 + 1) = A_d(f' + 1),$$
$$H_d(f') \subset A_d(f_1) \cap A_d(f_2) = A_d(f_1 \vee f_2) = A_d(f').$$

Итого, для формулы f' мы можем с полиномиальной от n сложностью вычислить пару подпространств $G_d(f'), H_d(f')$, удовлетворяющих условию (1), имея такие пары для всех подформул формулы f'. Используя этот рекурсивный метод, можно за время, полиномиальное от n и от длины формулы для f, получить подпространства $G_d(f) \subset A_d(f + 1)$, $H_d(f) \subset A_d(f)$.

Недостатком этого алгоритма является то, что он вычисляет лишь некоторые подпространства, которые могут оказаться нулевыми, в то время как $A_d(f+1)$ и $A_d(f)$ могут быть нетривиальными. В некоторых случаях во вложении $A_d(f_1) + A_d(f_2) \subset A_d(f_1 \cdot f_2)$ достигается равенство. Ниже приведены два результата, полученных в этом направлении.

Теорема 4. *Пусть $f_1, f_2 \in \mathcal{F}_n$ — ненулевые аффинные функции, причём $f_1 \neq f_2$ и $f_1 \neq f_2 + 1$. Тогда пространство $A_1(f_1 \cdot f_2)$ представимо в виде прямой суммы*

$$A_1(f_1 \cdot f_2) = A_1(f_1) \oplus A_1(f_2).$$

Доказательство. Заметим сначала, что если $\ell \in \mathcal{F}_n$ — ненулевая аффинная функция, то $A_1(\ell) = \{0, \ell + 1\}$. Поэтому сумма пространств $A_1(f_1)$ и $A_1(f_2)$ — прямая, и достаточно доказать, что $\dim A_1(f_1 \cdot f_2) = 2$.

Для функций f_1 и f_2 существует такое обратимое аффинное преобразование $\tau : V_n \to V_n$, что

$$\ell_1(x_1, \ldots, x_n) := f_1 \circ \tau(x_1, \ldots, x_n) = x_1,$$
$$\ell_2(x_1, \ldots, x_n) := f_2 \circ \tau(x_1, \ldots, x_n) = x_1 + x_2.$$

Кроме того, отображение $\varphi_\tau : g \mapsto g \circ \tau$ задаёт изоморфизм $A_1(f) \cong A_1(f \circ \tau)$ для любой функции $f \in \mathcal{F}_n$, то есть

$$A_1(f_1) \cong A_1(f_1 \circ \tau) = A_1(\ell_1),$$
$$A_1(f_2) \cong A_1(f_2 \circ \tau) = A_1(\ell_2),$$
$$A_1(f_1 \cdot f_2) \cong A_1((f_1 \cdot f_2) \circ \tau) = A_1((f_1 \circ \tau) \cdot (f_2 \circ \tau)) = A_1(\ell_1 \cdot \ell_2).$$

Составим систему линейных уравнений на коэффициенты аффинной функции $g \in A_1(\ell_1 \cdot \ell_2)$. Очевидно, что при переменных $x_3, \ldots, x_n$ стоят нулевые коэффициенты, и функция g имеет вид $g(x) = a_0 + a_1 x_1 + a_2 x_2$.

$$\begin{aligned} & g \cdot \ell_1 \cdot \ell_2 = 0 \\ & \iff a_0 x_1 + a_1 x_1 + a_2 x_1 x_2 + a_0 x_1 x_2 + a_1 x_1 x_2 + a_2 x_1 x_2 = 0 \\ & \iff \begin{cases} a_0 + a_1 = 0 \\ a_2 + a_0 + a_1 + a_2 = 0 \end{cases} \iff a_0 + a_1 = 0. \end{aligned}$$

Размерность пространства функций g, удовлетворяющих последнему уравнению, равна двум, а следовательно $\dim A_1(f_1 \cdot f_2) = \dim A_1(\ell_1 \cdot \ell_2) = 2$. □

Теорема 5. *Пусть $f_1, f_2 \in \mathcal{F}_n \setminus \{0\}$. Первые m переменных фиктивны для функции f_2, а остальные переменные фиктивны для функции f_1. Тогда пространство $A_1(f_1 \cdot f_2)$ представимо в виде прямой суммы*

$$A_1(f_1 \cdot f_2) = A_1(f_1) \oplus A_1(f_2).$$

Доказательство. Очевидно, что $A_1(f_1) \cap A_1(f_2) = \{0\}$. Покажем, что любая функция $\ell \in A_1(f_1 \cdot f_2)$ представима в виде суммы $\ell = \ell_1 + \ell_2$, где $\ell_1 \in A_1(f_1)$ и $\ell_2 \in A_1(f_2)$.

Пусть $z = (z_1, \ldots, z_n) \in V_n$. Обозначим $x := (z_1, \ldots, z_m)$, $y := (z_{m+1}, \ldots, z_n)$. В этих обозначениях имеем $(x, y) = z$. Каждая функция $\ell \in A_1(f_1 \cdot f_2)$ имеет вид

$$\ell(z) = \sum_{i=1}^{n} a_i z_i + b.$$

То есть ℓ представляется в виде суммы

$$\ell(z) = \ell'(x) + \ell''(y),$$

где

$$\ell'(x) = \sum_{i=1}^{m} a_i z_i, \quad \ell''(y) = \sum_{i=m+1}^{n} a_i z_i + b.$$

Тогда

$$\begin{aligned} \ell \in A_1(f_1 \cdot f_2) &\iff \forall x \forall y \quad \ell(x,y) \cdot f_1(x) \cdot f_2(y) = 0 \\ &\iff \forall x \forall y \quad \ell'(x) \cdot f_1(x) \cdot f_2(y) + \ell''(y) \cdot f_1(x) \cdot f_2(y) = 0 \end{aligned} \tag{2}$$

Возможны 2 случая:

(а) $\boxed{\forall x \quad \ell'(x) \cdot f_1(x) = 0}$: По условию $f_1 \neq 0$, то есть $\exists x_0 : f_1(x_0) = 1$. Подставив в (2) $x = x_0$, получим:

$$\forall y \quad 0 \cdot f_2(y) + \ell''(y) \cdot 1 \cdot f_2(y) = 0 \iff \forall y \quad \ell''(y) \cdot f_2(y) = 0.$$

Таким образом, $\ell' \in A_1(f_1)$ и $\ell'' \in A_1(f_2)$.

(б) $\boxed{\exists x_0 : \ell'(x_0) \cdot f_1(x_0) = 1}$: Тогда $f_1(x_0) = 1$. Подставим в (2) $x = x_0$:

$$\begin{aligned} &\forall y \quad 1 \cdot f_2(y) + \ell''(y) \cdot 1 \cdot f_2(y) = 0 \\ &\iff \forall y \quad (\ell''(y) + 1) \cdot f_2(y) = 0 \\ &\iff \forall y \quad \ell''(y) \cdot f_2(y) = f_2(y). \end{aligned}$$

Подставим последнее равенство в (2), чтобы избавиться от $\ell''(y)$:

$$\forall x \forall y \quad \ell'(x) \cdot f_1(x) \cdot f_2(y) + f_1(x) \cdot f_2(y) = 0.$$

По условию $\exists y_0 : f_2(y_0) = 1$. Тогда

$$\forall x \quad (\ell'(x) + 1) \cdot f_1(x) = 0.$$

Таким образом, для этого случая $\ell' + 1 \in A_1(f_1)$ и $\ell'' + 1 \in A_1(f_2)$, а $(\ell' + 1) + (\ell'' + 1) = \ell$. □

Литература

[1] Courtois N., Meier W. Algebraic Attacks on Stream Ciphers with Linear Feedback. Eurocrypt 2003, LNCS 2656, pp. 345-359, Springer, 2003.

[2] Meier W., Pasalic E., Carlet C. Algebraic Attacks and Decomposition of Boolean Functions. Eurocrypt 2004, LNCS 3027, pp. 474-491, Springer, 2004.

[3] Armknecht F. On the Existence of low-degree Equations for Algebraic Attacks, Cryptology ePrint Archive: Report 2004/185, http://eprint.iacr.org/2004/185

[4] Баев В. В. О некоторых алгоритмах построения аннигиляторов низкой степени для булевых функций. Представлено к опубликованию в журнале «Дискретная математика».

О числе булевых функций, имеющих линейный аннигилятор

М. С. Никифоров, А. В. Покровский

Пусть $(\mathcal{F}_m, \oplus)$ — пространство булевых функций от m переменных и 0 — нулевой элемент данного пространства. Обозначим $\mathrm{wt}(f)$ — вес функции f, $\overrightarrow{c_f}$ — вектор коэффициентов Фурье и $c_f(i)$ его i-ю координату. Скалярное произведение векторов $\overrightarrow{\gamma}, \overrightarrow{\beta} \in V_m$ будем записывать в виде $\langle \overrightarrow{\gamma}, \overrightarrow{\beta} \rangle$.

Определение 1. Аннигилятором функции $f \in \mathcal{F}_m$ называется функция $g \in \mathcal{F}_m$ для которой выполняется тождество $fg = 0$.

Обозначим множество всех аннигиляторов функции f через $\mathrm{Ann}(f)$. Для любого a из $\mathbb{R}$ будем обозначать

$$\lceil a \rceil = \begin{cases} \text{ближайшее целое сверху}, & \text{если } a \notin \mathbb{Z} \\ a, & \text{если } a \in \mathbb{Z}. \end{cases}$$

В работе [1] рассматриваются функции, имеющие линейные аннигиляторы. Использование линейных аннигиляторов позволяет переходить от решения нелинейной системы к решению ее линейного следствия. В связи с этим актуален вопрос о числе функций имеющих ненулевой линейный аннигилятор. Ответ на него дает нижеследующая теорема.

Теорема 1. *1. Условие $g(x) \in \mathrm{Ann}(f)$ эквивалентно равенству $\langle \overrightarrow{c_f}, \overrightarrow{c_g} \rangle = 0$. В частности, для произвольной булевой функции f множество $\mathrm{Ann}(f)$ содержит аффинную функцию $a(x)$ тогда и только тогда, когда в векторе $\overrightarrow{c_f}$ существует хотя бы одна координата по модулю равная $c_f(0)$.*

2. Число функций, имеющих линейный аннигилятор, рассчитывается по формуле:

$$(2^m - 1)2^{2^{m-1}} + \sum_{k=2}^{2^m-1} (-1)^{k+1} \sum_{r=\lceil \log_2(k+1) \rceil}^{k} N(k,m,r) 2^{2^m - r},$$

где $N(k,m,r)$ — число $k \times m$ матриц ранга r над полем $\mathbb{F}_2$ не имеющих нулевой и одинаковых строк (матрицы берутся с точностью до перестановки строк).

3. Коэффициент $N(k,m,r)$ вычисляется по рекуррентной формуле:

$$N(k,n,r) = \frac{(2^m - 2^{r-1})N(k-1,m,r-1)}{k!} + \frac{(2^r - k)N(k-1,m,r)}{k!},$$
$$N(k,m,r) = 0 \quad \text{при } r < \log_2(k+1),$$
$$N(2,m,2) = \frac{(2^m - 1)(2^m - 2)}{2}.$$

Доказательство. 1. Пусть $g(x) \in \mathrm{Ann}(f)$. Тогда из определения коэффициентов Фурье следует, что $\langle \overrightarrow{c_f}, \overrightarrow{c_g} \rangle = 2^m \cdot \mathrm{wt}(f \cdot g)$, откуда получаем первую часть утверждения. Вторая часть следует из того, что в векторе коэффициентов Фурье аффинной функции $a(x_1, \ldots, x_m) = \langle \alpha, x \rangle + \varepsilon$, $\varepsilon \in \{0, 1\}$ отличны от нуля и равны по модулю лишь $c_f(0)$ и $c_f(\alpha)$.

2. Пользуясь тем, что отношение f «аннулирует» g симметрично, для доказательства теоремы достаточно найти мощность множества:

$$\left| \bigcup_{l=1}^{2^m-1} \mathrm{Ann}(\langle \overrightarrow{l}, \overrightarrow{x} \rangle) \right|, \tag{1}$$

где $\langle \overrightarrow{l}, \overrightarrow{x} \rangle$ — линейная функция, для которой $\overrightarrow{l}$ — вектор из V_m, являющийся двоичным разложением числа l.

Для нахождения числа (1) воспользуемся формулой включения — исключения, тогда можно записать равенство:

$$\left| \bigcup_{l=1}^{2^m-1} \mathrm{Ann}(\langle \overrightarrow{l}, \overrightarrow{x} \rangle) \right| = \sum_{k=1}^{2^m-1} (-1)^{k+1} \times \sum_{1 \leqslant i_1 < \ldots < i_k \leqslant 2^m-1} \left| \mathrm{Ann}(\langle \overrightarrow{l_{i_1}}, \overrightarrow{x} \rangle) \cap \ldots \cap \mathrm{Ann}(\langle \overrightarrow{l_{i_k}}, \overrightarrow{x} \rangle) \right| \quad (2)$$

Поскольку линейная функция равновероятна, то $|\mathrm{Ann}(\langle \overrightarrow{l}, \overrightarrow{x} \rangle)| = 2^{\sum_{i=0}^{2^m-1} binommi} = 2^{2^{m-1}}$, поэтому сумма слагаемых при $k = 1$ равна $(2^m - 1)2^{2^{m-1}}$. Пусть $k > 2$, тогда найдем $|\{\overrightarrow{x} : \langle \overrightarrow{l_{i_1}}, \overrightarrow{x} \rangle = 0, \ldots, \langle \overrightarrow{l_{i_k}}, \overrightarrow{x} \rangle = 0\}|$ Это множество совпадает с множеством решений системы

$$\begin{cases} \langle \overrightarrow{l_{i_1}}, \overrightarrow{x} \rangle = 0 \\ \vdots \\ \langle \overrightarrow{l_{i_k}}, \overrightarrow{x} \rangle = 0. \end{cases}$$

и его мощность равна 2^{m-r}, где r — ранг системы. Из этого следует, что $\left| \mathrm{Ann}(\langle \overrightarrow{l_{i_1}}, \overrightarrow{x} \rangle) \cap \ldots \cap \mathrm{Ann}(\langle \overrightarrow{l_{i_k}}, \overrightarrow{x} \rangle) \right| = 2^{2^{m-r}}$. Число слагаемых такого вида в формуле (2) ровно $N(k, m, r)$. Нижняя граница суммирования в (2) равна $\lceil \log_2(k+1) \rceil$ потому, что для $k \times n$ матриц ранга r, удовлетворяющих условию теоремы, число линейно зависимых строк равно $k - r$ и не может быть больше $2^r - r - 1$, откуда получаем, что $N(k, m, r)$ равно нулю при $r < \log_2(k+1)$.

3. Это утверждение вытекает из того, что при добавлении к $(k-1) \times m$ матрице ранга $r - 1$, удовлетворяющей второму пункту теоремы, k-ая строка может быть выбрана $2^m - 2^{r-1}$ способом. Если ранг $(k-1) \times m$ равен r, тогда k-ая строка может быть выбрана $2^r - (k-1) - 1 = 2^r - k$ способами. Поскольку матрицы берутся с точностью до перестановки строк, необходимо разделить получившееся выражение на $k!$. Условие $r < \log_2(k+1)$ было обосновано в пункте 2. Условие $N(2, m, 2) = \frac{(2^m-1)(2^m-2)}{2}$ очевидно. □

Данная теорема дает критерий наличия у функции линейного аннигилятора и способ подсчета числа функций, имеющих линейный аннигилятор.

Литература

[1] О. А. Логачев, А. А. Сальников, В. В. Ященко. Многогранники в конечной абелевой группе и их криптографические приложения. Третья Общероссийская научная Конференция «Математика и безопасность информационных технологий» (МаБИТ-04), Москва, 2004.

[2] W. Meier, E. Pasalic, C. Carlet. Algebraic Attacks and Decomposition of Boolean Functions. Proc. EUROCRYPT'2004, pp. 475-491.

[3] N. Courtois, W. Meier Algebraic attacks on stream ciphers with linear feedback. Proc. EUROCRYPT'2003, LNCS, v. 2656, pp. 346–359, Springer-Verlag, 2003.

[4] N. Courtois. Fast algebraic attacks on stream ciphers with linear feedback. Proc. CRYPTO'2003, LNCS, v. 2729, pp. 176–194, Springer-Verlag, 2003.

Аффинные преобразования, распространяющие искажения, и проблема А. А. Маркова

Б. А. Погорелов, М. А. Пудовкина

В 1956 г. А. А. Марковым была доказана теорема о биективных преобразованиях слов длины n в конечном алфавите, сохраняющих их длины и не распространяющих искажения. Одновременно, в связи с исследованием шифров, не распространяющих искажения, им был поставлен вопрос о природе обратимых преобразований, распространяющих искажения не более, чем в k раз, где $k \geqslant 2$.

При этом отмечалось, что задача представляет большие трудности уже при $k = 2$ и что соответствующие преобразования, по-видимому, весьма разнообразны. Ставилась задача (см. [1],[8]) так или иначе «обозреть» эти преобразования путем надлежащей классификации. В ряде работ (см. [5], [7], [3], [6]) в связи с обобщением теоремы А. А. Маркова (см., например, [2]) рассматривались инъективные отображения, не размножающие искажения типа замены, пропуска и вставки букв. В этом случае удалось полностью описать подобные отображения [4].

В работе [10] при исследовании подметрик метрики Хэмминга на векторном пространстве $V_n(2)$ был приведен первый пример группы преобразований, увеличивающих число искажений типа замены букв не более, чем в $k = 2$ раза, относительно метрики Хэмминга. Эта группа оказалась подгруппой аффинной группы $\mathrm{AGL}_n(2)$.

В настоящей работе проблема А. А. Маркова рассмотрена для преобразований векторных пространств размерности n над конечным полем, а также преобразований модулей над кольцом вычетов. Над этими структурами описаны все аффинные преобразования, распространяющие искажения не более, чем в k раз, где $k = 2, 3, \ldots, n$. Аналогичные преобразования можно строить в случае модулей над другими кольцами. Всюду ниже будем придерживаться следующих обозначений:

- $\mathbb{N}$ – множество натуральных чисел;
- n, q – произвольные числа из $\mathbb{N}$, отличные от единицы;
- R – кольцо вычетов Z_q или поле $GF(q)$;
- $\overline{m,k} = m, m+1, \ldots, k, m < k$;
- $V_n = \{(\alpha_1, \ldots, \alpha_n) \in R^n\}$ – R-модуль;
- GL_n – полная линейная группа размерности n над R;
- e_n – единичная $(n \times n)$-матрица;
- $\alpha^{\downarrow}$ – вектор-столбец;
- $\chi(\alpha, \beta) = |\{i : \alpha_i \neq \beta_i, i = \overline{1,n}\}|$ – расстояние Хэмминга между векторами $\alpha = (\alpha_1, \ldots, \alpha_n)$ и $\beta = (\beta_1, \ldots, \beta_n)$; $\|\alpha\| = \|\alpha\|_1$;
- $\vec{0}, \vec{1}$ – нулевой и единичный векторы, соответственно;
- $\|\alpha\|_p = \chi_p(\alpha, \vec{0})$ –вес вектора $\alpha \in V_n$ в метрике χ_p;
- $\varepsilon_j = (0 \ldots 0 \overset{j}{1} 0 \ldots 0)$, $\delta_j = (\overbrace{1 \ldots 1}^{j} 0 \ldots 0)$–векторы из $V_n, j = \overline{1,n}$;
- M_n – множество всех $(n \times n)$-матриц над R (линейных преобразований R-модуля V_n в базисе $\varepsilon_1, \ldots, \varepsilon_n$);
- $M_{n,p}^{(k)}$ – множество всех преобразований из M_n, распространяющих искажения не более, чем в k раз, при действии на V_n относительно метрики χ_p;

- $\tilde{M}_{n,p}^{(k)}$ – множество всех преобразований из $M_{n,p}^{(k)}$ без нулевых столбцов;
- $\mathrm{GL}_{n,p}^{(k)} = M_{n,p}^{(k)} \cap \mathrm{GL}_n$;
- $\mathrm{diag}(b_1, b_2, \ldots, b_r)$ – клеточно-диагональная матрица с матрицами $b_1, b_2, \ldots, b_r$ на диагонали;
- $V_n^{(r)} = \{\alpha \in V_n : \|\alpha\| = r\}$;
- $\langle a_1, \ldots, a_m \rangle$ – группа, порожденная элементами $a_1, \ldots, a_m$;
- $\mathrm{Isom}\,\chi$ – группа изометрий метрического пространства (V_n, χ) с метрикой χ;
- $J_n(r)$ – жорданова клетка размера n с корнем r;
- $\widetilde{S}_n$ – группа подстановочных матриц из GL_n.

Опишем преобразования, распространяющие искажения не более, чем в k, раз относительно одной из подметрик метрики Хэмминга, приведенных в [10]. Пусть $\chi_p(\alpha, \beta) = \left\lceil \frac{\chi(\alpha,\beta)}{p} \right\rceil$ для любых $\alpha, \beta \in V_n$, $p \in \{\overline{1,n}\}$. Покажем, что χ_p – метрика.

Утверждение 1. *Пусть $p \in \{\overline{1,n}\}$. Тогда функция χ_p является метрикой на V_n.*

Доказательство. Очевидно, что $\chi_p(\alpha, \beta) = 0$ только при $\alpha = \beta$ и $\chi_p(\alpha, \beta) = \chi_p(\beta, \alpha)$ для любых $\alpha, \beta \in V_n$. Неравенство треугольника следует из соотношений

$$\left\lceil \frac{\chi(\alpha,\beta)}{p} \right\rceil \leqslant \left\lceil \frac{\chi(\alpha,\gamma)}{p} + \frac{\chi(\gamma,\beta)}{p} \right\rceil \leqslant \left\lceil \frac{\chi(\alpha,\gamma)}{p} \right\rceil + \left\lceil \frac{\chi(\gamma,\beta)}{p} \right\rceil. \quad \square$$

Будем говорить, что преобразование $g : V_n \to V_n$ k-распространяет искажения относительно метрики χ_p, если для любых векторов $\alpha, \beta \in V_n$ справедливо неравенство

$$\chi_p(\alpha^g, \beta^g) \leqslant k\chi_p(\alpha, \beta).$$

Пусть $H_n = \{h_\alpha : \beta \to \beta + \alpha, \forall \alpha \in V_n\}$ – группа сдвигов R -модуля V_n. Если $G \subseteq M_n$ и множество H_nG является группой, то будем использовать обозначение $AG(\leqslant \mathrm{AGL}_n)$. Поскольку $\mathrm{Isom}\,\chi = A\tilde{S}_n$ (см., например, [10]), то группа сдвигов не распространяет искажения, а множество преобразований $H_nM_{n,p}^{(k)}$ k-распространяет искажения относительно метрики χ_p. Таким образом, достаточно описать только преобразования, стабилизирующие $\vec{0}$, т.е. такие $g \in M_n$, что

$$\|\gamma^g\|_p \leqslant k \|\gamma\|_p \tag{1}$$

для любого вектора $\gamma \in V_n$.

Теорема 1. *Пусть $k, p \in \{\overline{2,n}\}$. Преобразование $g \in M_n$ k-распространяет искажения относительно метрики χ_p тогда и только тогда, когда вес суммы любых $r \in \{\overline{1,p}\}$ строк матрицы преобразования g не больше $k \cdot p$.*

Доказательство. Необходимость. Пусть $g \in M_{n,p}^{(k)}$, т.е. для любого $\alpha \in V_n$ справедливо неравенство

$$\|\alpha^g\|_p = \left\lceil \frac{\|\alpha^g\|}{p} \right\rceil \leqslant k \left\lceil \frac{\|\alpha\|}{p} \right\rceil,$$

которое эквивалентно $\|\alpha^g\| \in \left\{ \overline{0, k \cdot p \left\lceil \frac{\|\alpha\|}{p} \right\rceil} \right\}$. Тогда для любых $r \in \{\overline{1,p}\}$, $\alpha = \sum\limits_{j \in J} \alpha_j \varepsilon_j \in V_n^{(r)}$, где $\alpha_j \neq 0$ при $j \in J \subseteq \{\overline{1,n}\}$, $|J| = r$, справедливы соотношения

$$\|\alpha^g\| = \left\| \left(\sum_{j \in J} \alpha_j \varepsilon_j \right)^g \right\| = \left\| \sum_{j \in J} \alpha_j \varepsilon_j^g \right\| \leqslant \sum_{j \in J} \left\| \varepsilon_j^g \right\| \leqslant k \cdot p$$

Для завершения доказательства осталось заметить, что $\left\|\varepsilon_j^g\right\|$ – вес j-й строки матрицы преобразования g в базисе $\varepsilon_1, \ldots, \varepsilon_n$.

Достаточность. Рассмотрим произвольное множество $J \subseteq \{\overline{1,n}\}$, $r = |J|$, $r \in \left\{\overline{(t-1)\cdot p+1, t\cdot p}\right\}$, $t \geqslant 1$, и вектор $\alpha \in V_n^{(r)}$, $\alpha = \sum\limits_{j\in J} \alpha_j \varepsilon_j$, $\alpha_j \neq 0$ при $j \in J$. Пусть $J_1, \ldots, J_t$ произвольное t-разбиение множества J такое, что мощности всех подмножеств равны p, кроме J_t. Тогда справедливы соотношения

$$\|\alpha^g\| = \left\|\left(\sum_{j\in J}\alpha_j\varepsilon_j\right)^g\right\| = \left\|\sum_{j\in J}\alpha_j\varepsilon_j^g\right\| \leqslant$$
$$\leqslant \sum_{j\in J_1}\left\|\varepsilon_j^g\right\| + \ldots + \sum_{j\in J_{t-1}}\left\|\varepsilon_j^g\right\| + \sum_{j\in J_t}\left\|\varepsilon_j^g\right\| \leqslant k\cdot p\cdot t$$

Следовательно, для любого числа r и произвольного вектора $\alpha \in V_n^{(r)}$ справедливо $\|\alpha^g\| \in \{\overline{0, k\cdot t\cdot p}\}$. □

При $p = 1$ из теоремы 1 непосредственно получаем критерий того, что преобразование $g \in M_n$ k-распространяет искажения относительно метрики Хэмминга.

Следствие 1. *Пусть $k \in \{\overline{2,n}\}$. Преобразование $g \in M_n$ k-распространяет искажения относительно метрики Хэмминга тогда и только тогда, когда число ненулевых элементов в каждой строке матрицы преобразования g не больше k.*

Тем самым в теореме 1 описан широкий класс преобразований, необязательно обратимых. Приведем некоторые примеры обратимых линейных преобразований из $\mathrm{GL}_{n,1}^{(k)}$.

Следствие 2. *Если $\beta_{(i)}^{\downarrow}$ – вектор-столбец из V_n с нулевой i-й координатой, $g_{i,\beta}$ – матрица, у которой i-столбец равен $\beta_{(i)}^{\downarrow}$, остальные нулевые, $i \in \{\overline{1,n}\}$, то трансвекция $e_n + g_{i,\beta}$ 2-распространяет искажения относительно метрики Хэмминга.*

Доказательство следует из следствия 1.

Отметим, что первый пример преобразования $b = e_n + g_{1,\beta}$, $\beta_{(1)} = (0, 1, \ldots, 1)$, 2-распространяющего искажения относительно метрики Хэмминга, приведенный в [10], лежит в описанном классе. Более того, $S_n^{(1)} \subset \mathrm{GL}_{n,1}^{(2)}$, где $S_n^{(1)} = \left\langle \tilde{S}_n, b \right\rangle$, причем последняя группа является группой изометрий метрики χ_2.

Следствие 3. *Для произвольных чисел $k, t \in \{\overline{1,n}\}$ и произвольного упорядоченного t-разбиения $(n_1, \ldots, n_t)$ числа n имеет место включение:*

$$\{\mathrm{diag}(g_1, \ldots, g_t) | g_i \in \mathrm{GL}_{n_i}^{(k)}, i = \overline{1,t}\} \subset \mathrm{GL}_{n,1}^{(k)},$$

Приведем пример преобразований из $\mathrm{GL}_{n,1}^{(k)}$, образующих группу для любого натурального числа $k \geqslant 1$.

Следствие 4. *Пусть $t \in \{\overline{1,n}\}$, $(n_1, \ldots, n_t)$ – произвольное упорядоченное r -разбиение числа n, $k = max\{n_1, \ldots, n_t\}$. Тогда любой элемент группы*

$$G(n_1, \ldots, n_t) = \{\mathrm{diag}(g_1, \ldots, g_t) | g_i \in \mathrm{GL}_{n_i}, i = \overline{1,t}$$

k-распространяет искажения относительно метрики Хэмминга.

Ясно, что

$$\tilde{S}_n G(n_1, \ldots, n_r)\tilde{S}_n \subseteq \mathrm{GL}_{n,1}^{(k)}.$$

Следствие 5. *В условиях следствия 3 для произвольных элементов $r_1, \ldots, r_t$ из R и произвольного преобразования*

$$g \in H_n \widetilde{S}_n \,\mathrm{diag}(J_{n_1}(r_1), \ldots, J_{n_t}(r_t))\widetilde{S}_n$$

выполняется неравенство

$$\chi(\alpha^g, \beta^g) \leqslant 2\chi(\alpha, \beta).$$

Найдем число преобразований из M_n, k-распространяющих искажения относительно метрики Хэмминга.

Утверждение 2. *Пусть k – произвольное натуральное число, $k \geqslant 1$. Тогда справедливы равенства*

$$|M_{n,1}^{(k)}| = \left((q-1)\sum_{i=1}^{k}\binom{n}{i}\right)^n,$$

$$|\tilde{M}_{n,1}^{(k)}| = \sum_{r=0}^{n}(-1)^r\left(\binom{n}{r}\right)\left((q-1)\sum_{i=1}^{k}\binom{n-r}{i}\right)^n.$$

Доказательство проводится непосредственно с использованием метода включения-исключения (см., например, [9]).

Очевидно, что $M_{n,p-1}^{(k)} \subseteq M_{n,p}^{(k)}$ для любого натурального $p \geqslant 2$. Покажем, что $M_{n,p}^{(k)} \backslash M_{n,p-1}^{(k)} \neq \varnothing$. В следующем утверждение описаны треугольные преобразования из $M_{n,p}^{(k)} \backslash M_{n,p-1}^{(k)}$.

Утверждение 3. *Пусть $p \geqslant 2$, $k \in \{\overline{2,n}\}$. Пусть также $t = (k-1)\cdot p + 1$, и преобразование g обладает свойствами*

$$\begin{aligned}
&\varepsilon_j^g - \varepsilon_j \in \langle \varepsilon_1, \ldots, \varepsilon_{j-1}\rangle, j = \overline{1, t-1},\\
&\varepsilon_t^g = \delta_t,\\
&\varepsilon_j^g - \varepsilon_j \in \langle \varepsilon_j - t + 1, \ldots, \varepsilon_{j-1}\rangle, j = \overline{t+1, p-1+t},\\
&\varepsilon_j^g = \varepsilon_j, j = \overline{p+t, n}.
\end{aligned} \tag{2}$$

Тогда имеет место включение

$$\tilde{S}_n g \tilde{S}_n \subseteq M_{n,p}^{(k)} \backslash M_{n,p-1}^{(k)}.$$

Доказательство. Согласно теореме 1 для построения преобразования $g \in M_{n,p}^{(k)} \backslash M_{n,p-1}^{(k)}$ достаточно указать n линейно независимых векторов $\alpha^{(1)}, \ldots, \alpha^{(n)}$ таких, что вес суммы любых r из них для каждого $r \in \{\overline{1,p}\}$ был не больше $k \cdot p$. Пусть $\varepsilon_j^g = \alpha^{(j)}$, $j = \overline{1,n}$.

Из способа задания (2) преобразования g легко видеть, что векторы $\alpha^{(1)}, \ldots, \alpha^{(n)}$ линейно независимы, значит $g \in \mathrm{GL}_n$. Наибольший вес суммы $r \in \{\overline{1,p}\}$ строк равен $p - 1 + t$, причем $k \cdot (p-1) < p - 1 + t \leqslant k \cdot p$. □

Литература

[1] Марков А. А. О преобразованиях, не распространяющих искажений. Избранные труды, т. 2, 70-93.

[2] Алферов А.П.,Зубов А.Ю.,Кузьмин А.С., Черемушкин А.В. Основы криптографии. М., Гелиос АРВ, 2001г., 480с.

[3] Глухов М. М. Инъективные отображения слов, не размножающие искажения типа пропуска букв. Дискретная математика, т. 11, в. 2, 1999, 20-39.

[4] Глухов М.М. Инъективные отображения слов, не размножающие искажения. Труды по дискретной математики. т. 4, 2001, 17-32.

[5] Бабаш А. В., Глухов М. М., Шанкин Г. П. О преобразованиях множества слов в конечном алфавите, не размножающих искажений. Дискретная математика, т. 9, в. 3, 1997, 3-19.

[6] Левенштейн В.И. О совершенных кодах в метрике выпадений и вставок. Дискретн. матем., 1991, т. 3, с. 1-20.

[7] Глухов М.М. Инъективные отображения слов, не размножающих искажений. Матем. вопросы кибернетики. 1998, в. 7, с. 349-350.

[8] Глухов М. М., Погорелов Б. А. О некоторых применениях групп в криптографии. Труды конференции «Математика и безопасность информационных технологий», 2004.

[9] Сачков В. Н. Введение в комбинаторные методы дискретной математики. М.: Наука, 1982. 384 с.

[10] Погорелов Б. А. Метрические пространства типа Хэмминга и теорема Маркова А.А. В печати.

Часть IV
Секция «Математическое и программное обеспечение безопасности компьютерных систем»

Методология динамической защиты

П. Д. Зегжда, Д. П. Зегжда

Аннотация

Целью настоящего сообщения является попытка построения теоретических основ новой парадигмы так называемой динамической защиты компьютерных систем, позволяющей на основе анализа каналов информационного обмена обнаруживать попытки вторжения, контролировать текущее состояние безопасности, прогнозировать ее уровень и управлять защитой, для того чтобы непрерывно поддерживать систему в безопасном состоянии. Реализация этой технологии приводит к построению опережающей стратегии защиты, позволяющей не только обеспечить безопасность, но и предотвратить ее нарушение.

1 Технологии защиты

Анализ существующих тенденций позволяет сделать вывод о ретроспективной смене технологий защиты, которые условно могут быть определены как статистическая, активная, адаптивная и динамическая.

Статическая защита исходит из выбора наиболее опасных угроз, совокупность которых определяет набор функций, который необходимо реализовать, и класс защиты, которому должна соответствовать система. Набор функций защиты должен быть адекватен угрозам. Основной недостаток такой технологии состоит в ограничении класса угроз, расширение которого может привести к несостоятельности защиты.

Остальные рассматриваемые технологии защиты включают более или менее развитую систему контроля состояния системы, что позволяет расширить класс угроз, которым противостоит защита, и сделать ее многорубежной, что позволяет принимать решение о подключении дополнительных средств или административных мер, направленных на поддержание безопасности.

В основу предлагаемой систематизации технологий защиты положено два основных показателя: наличие средств анализа состояния системы и среды ее функционирования, а также используемые критерии безопасности (см. таблицу 1).

Для того чтобы раскрыть смысл этих определений в понимании авторов, введем следующую систему моделей.

2 Модель системы

Представим систему в виде набора объектов, образующих иерархическую структуру, представляющую семантическую сеть. Учитывая сложность компьютерной системы объекты могут рассматриваться на физическом, сигнальном, программном, алгоритмическом уровне. С точки зрения защиты информации элементы сети являются либо источниками информации (объектами O_i), либо получателями (субъектами S_i), причем один и тот же объект может выступать в зависимости от обстоятельств и как субъект и как объект. Субъект может выполнять над объектом обычный набор операций — чтение, запись и т.д. Установленная над множеством $\{S_i\}$ и $\{O_i\}$ алгебра отношений задает установленную модель политики безопасности.

Предлагается представить систему сетевой моделью в общем случае сетью фреймов, описываемых конструкцией вида

$$H = \langle I, C_1, C_2, \ldots, C_n, \Gamma \rangle,$$

где:

I — множество информационных единиц;

$C_1, C_2, \ldots, C_n$ — множество типов связей между информационными элементами;

Таблица 1: Характеристики существующих технологий построения защиты

Характер защиты	Объекты мониторинга			Методы оценки безопасности	Основные характеристики
	Состояние системы	Состояние системы защиты	Обмен с окружающей средой		
Статическая	Отсутствует	Отсутствует	Частичный	Оценка по нормативным документам	Адекватность угрозам
Активная	Частичный	Отсутствует	Анализ входящей информации	Анализ информационной среды	Надежность анализа входящей информации
Адаптивная	Частичный	Частичный	Анализ входящей информации	Контроль состояния средств защиты	Толерантность к угрозам, устойчивость управления
Динамическая	Полный	Полный	Анализ входящей информации и канала связи	Мониторинг безопасности системы, оценка рисков	Инвариантность защиты, достаточность, устойчивость к уязвимостям

Г — отображение, задающее связи из принятого набора между информационными единицами.

Компьютерную систему предлагается моделировать G-графом [4] с иерархической конструкцией с именованными вершинами функциональными дугами.

При этом объект представляется системой, состоящей из простых объектов O, не имеющих структуры и представляемых в виде набора признаков $\Pi = \langle p_1, \ldots, p_n \rangle$. Простые объекты на каждом уровне иерархии могут быть отнесены к одному из K классов. Например, для рассматриваемой области такими классами могут являться файлы, записи, запросы, поля базы данных и т.д., вплоть до сегментов памяти на твердом диске. Непосредственно анализируемый объект представляется семантическим графом SG, который является структурой, содержащей вершины двух типов — объектные — I_n и предикатные (процедурные) — Y, что соответствует декларативному и процедурному способам представления знаний в виде

$$SG = \langle I, Y, G \rangle,$$

где Y — структурные связи между вершинами.

Объектная вершина i задается тройкой множеств $\langle L, K, P \rangle$, где:

L — тип данного объекта O;

K — имя класса, к которому он принадлежит;

P — совокупность признаков, определяющих состояние объекта.

Каждой объектной вершине i соответствует один вход из множества $\{P\}$, задающий свойства и размеры элемента и множества связей с процедурными вершинами $y \in Y$, определяющими структурные связи между простыми объектами.

Множество типов Y определяется с помощью набора характеристических функций

$$g_{i_1}^{(g)}, g_{i_2}^{(g)}, \ldots, g_{i_m}^{(g)},$$

сопоставляющих типы связей g между объектами вершинами $i_1, i_2, \ldots, i_m$.

Функция $g_{i_m}^{(g)}$ принимает значение 1, если отношение, в общем случае, r-е типа $\langle g \rangle$, выполняется для набора $i_1, i_2, \ldots, i_m$ вершин, относящихся к классам $K_1, \ldots, K_N$, в противном случае $g_{i_m}^{(g)} = 0$. Если отношение $g_1^{(g)}$ выполняется для вершины i, то в графе SG существует предикатная вершина g и соединение и объектными вершинами $i = 1$. Множество связей образуют дуги $p \in \Gamma$.

Дуги помечены именами отношений g, т.е. являются ролевыми дугами. Суть ролевой дуги задается функцией преобразования, которая соответствует операции, выполняемой в компьютерной системе. В зависимости от выбранного уровня иерархии такими операциями могут быть операции над файлами, выполнение запроса, формирование транзакций или функции защиты, например аутентификация или криптографическое преобразование.

Выполняемая операция может быть формализована в виде бинарного отношения между взаимодействующими объектами в виде $O_i R O_{i+1}$. Возможно ввести понятие операции в виде последовательности отношений $O_i R_{i+1} \ldots R_{k,j} O_j$. Примером объектов O могут являться биты, файлы, поля записи, программные единицы. Выбор типа объекта определяется уровнем иерархии.

Отношения определяют как возможные действия (чтение, запись, уничтожение, назначение прав), так и функции защиты (аутентификация, аудит, криптопреобразование).

В системе определена политика безопасности, определяющая допустимые и недопустимые цепочки отношений.

Кроме того, в системе осуществляется контроль управления доступом в виде ограничения на определенные функции дуг и контроля за состоянием.

3 Общая модель нарушения компьютерной безопасности

Применительно к задаче компьютерной безопасности состояние системы определяется на каждом уровне иерархии набором множеств

$$\langle O, R, Rul, L_{ij} \rangle,$$

где:

O — множество допустимых объектов $\{O_i\}$;

R — множество отношений, построенное по бинарному принципу;

$\{R_{ij}\}$ — тип отношений — определен для типа объекта;

Rul — правила контроля цепочки отношений в соответствии с политикой безопасности. Цепочки отношений строятся в соответствии с объектным подходом с использованием принципов наследования, инкапсуляции, полиморфизма;

L_{ij} — функции передачи от объекта к объекту, построенные на типе отношений R_{ij} и включающие запись, чтение, изменение и т.д.

Изменение состояния происходит одним из следующих способов: путем изменения множества $\{O_i\}$; путем реализации отношений R_{ij}; путем построения цепочки $R \times R$ с инициализацией функций L_{ij}, что приводит к действиям над объектом O или к изменению $\{O_i\}$.

Функции L_{ij}, реализуемые на отношениях R_{ij}, могут быть систематизированы следующим образом:

- функции, изменяющие объект L^{new}, что приводит к изменению $\{O_i\}$;
- функции, не изменяющие (чтение, копирование), что сохраняет $\{O_i\}$, но может привести к нарушениям Rul;
- функции, включающие функции безопасности, такие как идентификация O_i, аутентификация O_iRO_j, фильтрация O_iRO_j по параметрам;
- криптографические преобразования объекта O_i в объект O_i^k: $O_i^k \underset{M_k}{\longrightarrow} O_i^k$. $M_k\,(K)$ ставят в соответствие $O^k : O_i$, причем обратное преобразование требует генерации ключа K.

Следует учесть, что некоторые объекты могут иметь деструктивный характер — осуществлять сканирование других объектов, их разрушение, реализовать отношения, нарушающие $\{Rul\}$.

4 Систематизация возможных механизмов нарушения безопасности

Нарушение безопасности может происходить по одному из следующих механизмов:

1. Создание нового объекта, не входящего в допустимое множество объектов O или путем изменения параметров (отношений) свойственных существующим объектам. При данном механизме возникают следующие частные случаи:

   Утверждение 1.1. Создание «деструктивного» объекта, что представляет собой реализованную *атаку.*

   Утверждение 1.2. Возможное превышение заданной скорости создания «правильных» объектов, что приводит к отказу в обслуживании. Изменение параметров объекта или параметров функции его преобразования, называется *уязвимостью.*

2. Изменение (возникновение) новых отношений (типов отношений) между объектами и нарушении Rul.

   Утверждение 2.1. Возникновение новых отношений может не противоречить (не контролироваться) политикой безопасности, что является недостатком средств контроля или признаком уязвимости, т.е. Rul не изменяется.

   Утверждение 2.2. Возникновение новых отношений может появляться при временном отключении защиты (внутренний нарушитель).

   Утверждение 2.3. Из-за ошибок в реализации могут возникнуть случайные неконтролируемые отношения, что не является вторжением.

   Утверждение 2.4. Организация новых отношений вследствие появления новых объектов представляет собой *атаку.*

3. Изменение функций, построенных на типе отношений, причем возможны три случая:

   - целенаправленное изменение функций L_{ij}, что является следствием появления новых объектов (как правило, это направлено на нарушение функций защиты и является механизмом атаки). Частный случай — копирование или изменение ключевой информации, как параметра L_{ij};
   - не контролируемое, но стабильное изменение L_{ij} вследствие наличия уязвимости;
   - целенаправленное использование уязвимости, что является механизмом атаки.

   Предложенная модель позволяет построить полное множество механизмов нарушения безопасности и выделить среди них то, что называется *вторжением*.

   Предложенная систематизация возможных механизмов нарушений может быть обобщена в виде таблицы 2, которая является основанием для систематизации нарушений и позволяет в первом приближении указать признаки, по которым можно отделить вторжение как специфический вид нарушения безопасности, отличающийся от атак и несанкционированного доступа в виде нарушений политики безопасности.

Таблица 2: Механизм в терминах предложенной модели

Атака	Создание нового объекта (или изменение ролевых дуг) обладающего свойствами нарушающими правила доступа или нарушающих свойства защиты Rul
Вторжение	Изменение параметров и функций, реализуемых объектом L_{ij} или нового объекта без нарушения Rul
Уязвимости	Изменение функций L_{ij} или их параметров
Нарушение правил установления политики безопасности	Не соответствие L_{ij} множеству Rul
Аномалии поведения пользователя	Случайные или преднамеренные изменения параметров L_{ij} со стороны пользователей или администратора

Как следует из таблицы 2, под собственно вторжением, целесообразно понимать возникновение нового объекта вычислительной системы, не приводящее к нарушению правил политики безопасности, или изменение параметров функций, реализуемых существующими объектами, что не всегда приводит к изменению существующей политики безопасности и не контролируется системой управления доступом.

5 Феноменологический подход к построению безопасных информационных систем

В общем случае задача построения защищенных систем обработки информации может быть формализована следующим образом:

U — множество лиц-участников информационного процесса (потенциальных пользователей компьютерной системы), осуществляющих доступ к информации и ее обработку и обменивающихся информацией;

I — множество информационных объектов-контейнеров (документов, книг, папок, файлов и т.д.), хранящих информацию. Информация не может существовать сама по себе — она хранится в каком-либо контейнере.

С точки зрения безопасности информационные процессы моделируются с помощью отношений информационных потоков, определенных на этих базовых множествах. Под информационным потоком понимается событие, приведшее к появлению в точке назначения потока информации, находящейся перед этим событием в точке исхождения потока. С точки зрения безопасности алгоритмы обработки информации не имеют значения, важен только информационный обмен между пользователями и системой.

Существует два вида потоков:

$F^W \subseteq U \times I$ — отношение, описывающее потоки от пользователей к контейнерам;

$F^R \subseteq I \times U$ — отношение, описывающее потоки от контейнеров к пользователям.

Для того чтобы судить о безопасности системы должны быть определены базовые положения, характеризующую предметную область с точки зрения безопасности. Эти положения должны быть сформулированы в виде следующих *аксиом безопасности*:

1. Для каждой информации существует по крайней мере один пользователь, являющийся ее *доверенным источником*. Доверенные источники описываются функцией $TrustSrc\colon I \to U$.
2. Для каждого пользователя известен набор информации, для которой он является *уполномоченным потребителем*. Эти полномочия описываются функцией $Authority\colon U \to I$.

В каждый момент времени распределение информации в системе характеризуется следующими отношениями между пользователями и информацией:

1. $Know \subseteq U \times I$ — отношение известности, которое определяет какой пользователь знает какую информацию.
2. $Create \subseteq U \times I$ — отношение порождения, которое определяет какой пользователь поставляет какую информацию.

В общем случае задача обеспечения безопасности может быть сформулирована следующим образом:

Состояние системы является безопасным, если выполняются следующие *критерии безопасности состояния*:

1. Отношение известности не противоречит функции авторизации $Know \subseteq Authority$.
2. Отношение порождения не противоречит функции доверенного источника $Create \subseteq TrustSrc$.

Система в целом является безопасной, если выполняются следующие *критерии безопасности для системы*:

1. Текущее состояние системы безопасно.
2. Транзитивное замыкание отношений $Know$ и $Create$ не противоречит аксиомам безопасности.

5.1 Модель безопасности

В компьютерной системе пользователи не могут обрабатывать информацию непосредственно и вынуждены использовать инструменты-посредники — программные средства обработки информации, которые представляют их интересы в системе. Для того чтобы отразить это, в модель вводятся следующие понятия:

S — множество субъектов;

O — множество объектов;

P — множество программ-приложений, с помощью которых пользователи работают с информацией, находящейся в объектах, $P \subseteq O$;

Id — отношение идентификации, которое сопоставляет пользователю по крайней мере одного субъекта, $Id \subseteq U \times P(S)$;

Imp — отношение имперсонализации, которое для каждой программы определяет субъект, интересы которого она представляет, $Imp \subseteq P \times S$;

Sem — отношение семантики, которое устанавливает связь между объектами и информацией, которая в них содержится, $Sem \subseteq O \times I$.

Набор операций, осуществляемых программами над объектами, обозначается Op и представляет собой множество отношений вида $x \subseteq P \times O$, где x — тип операции (чтение, запись и др.). Тип операции зависит от природы объекта и возможностей программы.

Связь между операциями и информационными потоками описывается функцией $InfFlow\colon Op \to F$.

Доступ описывается отношением $A \subseteq P \times Op \times O$, которое определяет возможности программ по осуществлению операций в отношении объектов.

Модель безопасности $SM = \{R, A^A\}$ представляется в виде совокупности множества прав доступа R и отношения авторизованного (санкционированного) доступа $A^A \subseteq S \times O \times P(R)$, определяющего права субъектов на доступ к объектам.

Средства контроля доступа опираются на модель безопасности и запрещают операции, которые противоречат правилам модели. Функционирование средств контроля доступа описывается следующими отношениями:

$Map \subseteq Op \times R$ — устанавливает соответствие между операциями и правами доступа;

$A^S \subseteq P \times Op \times O$ — определяет множество операций программ над объектами, контролируемых средствами защиты.

Авторы предлагают новый *феноменологический* подход к построению защищенных систем и к оценке их защищенности, основанный на результатах исследования феномена уязвимости и предложенных критериях безопасности. В соответствии с этим подходом безопасность системы определяется, с одной стороны, отсутствием в ней так называемых уязвимостей, использующихся в качестве механизма нарушения безопасности, а с другой — способностью средств контроля и управления доступом реализовать требуемые ограничения, т.е. их толерантностью к НСД и угрозам в целом.

Предлагаемый критерий безопасности формулируется следующим образом:

1. Средства контроля доступа реализуют модель безопасности $(p, op, o) \in A^S \to \exists (s, o, R) \notin A^A$, что $(s, p) \in Imp, (op, R) \in Map$.

2. Реализация модели безопасности соответствует аксиома безопасности: $\forall (s, o, R) \in A^A$ выполняются следующие условия:

   $\exists u$ и I, такие, что $(u, s) \in Auth$ и $(I, o) \in Sem$, что $(u, i) \in Authority$, если $InfFlow(op) \in F^R$ для всех op, для которых $(op, R) \in Map$, и $(u, i) \in TustSrc$, если $InfFlow(op) \in F^W$ для всех op, для которых $(op, R) \in Map$.

3. Все функциональные возможности программ находятся под контролем средств защиты: $A^S \supseteq P \times Op \times O$.

Отличие предложенного подхода к безопасности состоит в том, что он может быть положен в основу технологии разработки защищенных систем, поскольку его критерий может быть использован в качестве целевой функции в процессе проектирования и разработки защищенной системы.

В пространстве модели системы предложенные выше условия текущей безопасности состоит из двух условий:

1. Реализация требуемого информационного процесса в виде модели взаимодействия M^G, реализуемой на графе G. При этом текущий информационный поток V является подмножеством $M^o \in M^G$.

2. Реализация для всего M^G отношения $R(s \times o)$ в соответствии с правилами политики безопасности $M^o \in M^G; R(s \times o)|_{i=\overline{1,n}} \in R^G$.

5.2 Модель уязвимости

Возникшие благодаря уязвимости дополнительные возможности доступа не обязательно будут противоречить политике безопасности, поэтому назовем такой доступ нелегитимным, чтобы отличать его от несанкционированного. Нелегитимный доступ — это всегда результат нарушения баланса между функциональными возможностями прикладных программ и средств защиты. Нелегитимный доступ не контролируется средствами защиты, поскольку либо осуществляется в обход них, либо игнорируется ими. Как и несанкционированный доступ — нелегитимный доступ может привести к нарушению политики безопасности (если таковая имеется) или к утечке информации, нарушению ее целостности, к потере работоспособности всей системы.

В соответствии с предложенной общей моделью определение уязвимости формализуется следующим образом: система содержит уязвимость, если:

либо $\exists(p, op, o) \in A^S$ для которых $\exists s, r$, такие, что $(p, s) \in Imp, (op, R) \in Map, (s, o, R) \notin A^A$,
либо $\exists(p, op, o) \in A^S$, такое, что $(p, op, o) \notin A^S$.

От обычных ошибок программирования и проектирования уязвимости отличает следующее. Во-первых, они появляются в условиях, которые появляются вследствие преднамеренно созданного стечения обстоятельств, а вероятность их случайного появления ничтожно мала. Во-вторых, их использование позволяет осуществлять действия, которые не пресекаются средствами защиты. Уязвимости можно разделить на два класса — уязвимости в средствах защиты и уязвимости в прикладных программах. Для средств защиты уязвимость — это свойство терять способность осуществлять свои функции при наступлении определенных условий. Для прикладных программ уязвимость — это свойство при определенных условиях приобретать новые функциональные возможности, благодаря которым программа приобретает способность осуществлять одно из следующих действий: чтение, запись данных, исполнение кода и потребление ресурсов. Соответственно, уязвимостям первого типа соответствуют ошибки программирования средств защиты и недостатки администрирования безопасности, а уязвимостям второго типа — недостаточность средств защиты, заключающаяся в невозможности контролировать действия прикладных программ, и отсутствие защиты как таковой.

6 Технология создания защищенных систем обработки информации

Предложенные пять основных принципов построения защищенных систем порождают пять компонент технологии их построения, каждая из которых определяет последовательность действий и условия достижения необходимых свойств защищенной системы. Рассмотрим основные положения разработанных технологий и преимущества их применения для построения защищенных систем.

6.1 Инвариантность моделей безопасности

Развиваемый в СЦЗИ ГОУ «СПбГПУ» новый подход к определению понятия «защищенная система» формулирует ее основную задачу как обеспечение адекватной реализации компьютерной системой информационных потоков и правил управления ими, существовавших до применения автоматизированных средств обработки информации. Это означает, что защищенная система должна, во-первых, реализовывать только те потоки информации, которые существовали до ее применения, и не создавать новых, и, во-вторых, обеспечить возможность управления потоками информации в соответствии с заданным набором правил политики безопасности.

В рамках этого подхода в СЦЗИ ГОУ «СПбГПУ» разработана комплексная модель безопасности защищенной информационной системы, отражающая прохождение информационных потоков и управление ими, основанная на обобщенном представлении политик безопасности и универсальной модели взаимодействий компонентов системы. При этом основным объектом моделирования является не осуществление тех или иных операций (доступ к файлам, обмен сообщениями и т.д.), а стоящие за ними информационные процессы. На рисунке 1 представлены основные стадии построения модели безопасности информационной системы на основе предложенной технологии.

Рис. 1: Моделирование информационных потоков

Использование такого подхода позволяет осуществлять сквозной контроль взаимодействия субъектов и объектов в соответствии с прохождением информационных потоков и выявлять точки размещения средств контроля доступа. Кроме того, за счет абстрагировании от особенностей архитектуры вычислительной системы появляется возможность применения унифицированных методов защиты, таких как средства управления доступом и контроля за его осуществлением, независящие от политики безопасности, средства идентификации и аутентификации, независящие от особенностей функционирования прикладных средств и т.д.

Разработанная модель взаимодействий субъектов и объектов, отражающая прохождение информационных потоков, может применяться к самому широкому классу систем, начиная от операционных систем и заканчивая распределенными вычислительными системами и сложным проблемно-ориентированным прикладным программным обеспечением.

Принципиальная блок-схема управления системой защиты, для предложенной технологии, построенной на динамическом принципе показана на рисунке 2.

Отличительной чертой предложенной схемы является наличие функций, перечисленных в таблице 1 и возможность реализации стратегии опережающей динамической защиты, предполагающей поддержание безопасности на заданном уровне в динамическом режиме.

Анализ существующих технологических приемов в виде комплексирования межсетевых экранов с системами обнаружения вторжений и антивирусной защитой [1], [2], построения гибридных ОС [3], позволяющих предотвратить ряд причин возникновения уязвимостей, а также системы адаптивного управления криптографической защитой, интегрирующей ЭЦП, защиту трафика и систему аутентификации позволяют надеяться, что предложенные теоретические положения находят практическое подтверждение.

7 Выводы

В статье предложен подход, обобщающий историю развития технологий обеспечения защиты, и приведены модели и принципы организации защиты, описывающие различные функционирование средств защиты на всех этапах развития технологий безопасности.

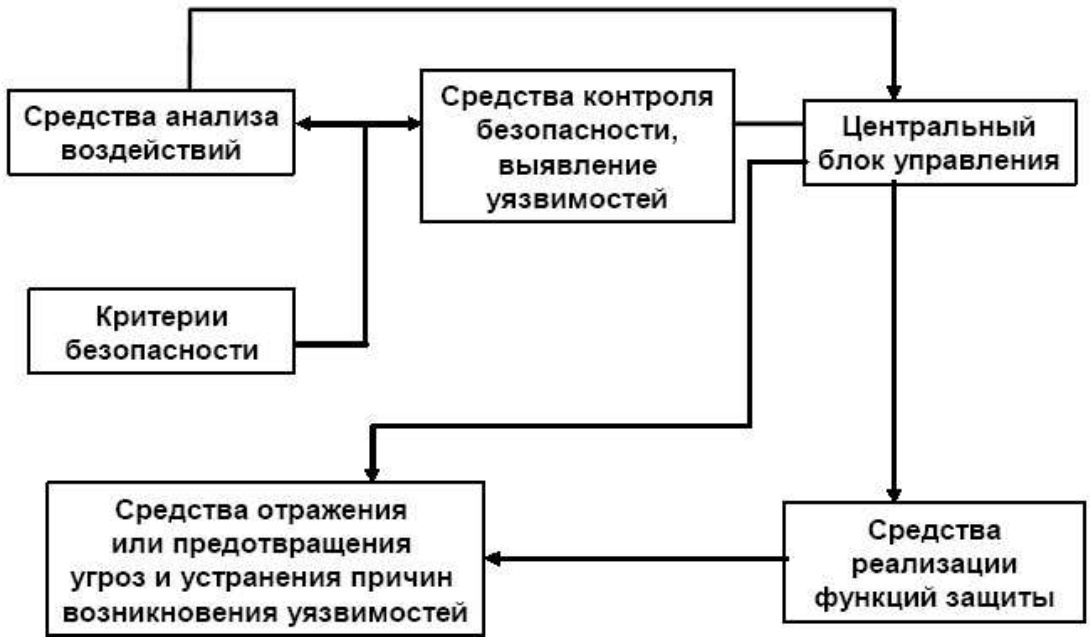


Рис. 2: Принципиальная блок-схема системы динамической защиты

Показана возможность построения принципиально новой технологии динамической защиты, состоящей в непрерывной оценке условий безопасности, на основе анализа состояния как самой защищаемой системы, так и средств защиты и вносимой в систему информации.

Литература

[1] Васильев Ю.С., Зегжда П.Д. Информационная безопасность. Развитие научно-исследовательских работ и подготовка кадров в Санкт-Петербургском государственном политехническом университете. СПб.: Изд-во Политехнического ун-та, 2005.

[2] Зегжда Д.П., Ивашко А.М. Основы безопасности информационных систем. М.: Горячая линия — Телеком, 2000.

[3] Зегжда Д.П., Вовк А.М. Защищенная гибридная операционная система «Linux over Феникс». Материалы конференции в МГУ 28-29 октября 2004 г. М.: МЦНМО, 2005.

[4] Искусственный интеллект. Модели и методы. Под ред. Д.А. Поспелова. Кн. 2. М.: Радио и связь, 1990.

О проверке свойств информационных потоков в распределенных информационных системах

Ф. М. Пучков

Одним из важнейших направлений исследований в современной теории безопасности информационных технологий является поиск математических моделей, позволяющих адекватно описывать и контролировать выполнение политик безопасности в больших, практически значимых распределенных информационно-вычислительных системах [1, 2]. В рамках указанного направления можно выделить такую актуальную в настоящее время задачу, как моделирование процесса вычислений в условиях, когда отдельные компоненты информационной системы (ИС) не доверяют друг другу полностью. Эта задача возникает, например, в больших распределенных информационных системах, использующих совместно как данные в хранилищах, так и вычислительные ресурсы, и имеющих в общем случае различную ведомственную принадлежность, разные политики их использования.

Рассмотрим возможную структуру подобной информационной системы более формально (см. рис. 1).

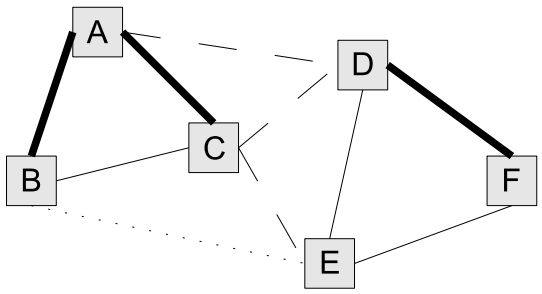


Рис. 1: A, B, C, D, E, F — компоненты ИС. Линии обозначают различные уровни доверия компонентов ИС друг к другу (жирной сплошной линии соответствует максимальный уровень доверия)

Будем предполагать, что в системе существуют данные, разделяемые для всех компонентов (называемые далее объектами), причем каждый компонент ИС может взаимодействовать с ними, однако имеет ограниченный доступ к ним. Такое предположение означает, что существует некоторая *функция доверия*, регламентирующая взаимодействие различных вычислительных узлов и, таким образом, задающая часть политики безопасности (ПБ) в рассматриваемой ИС. Для реализации указанного регламента необходимы средства, позволяющие контролировать информационные потоки, возникающие в информационной системе, с тем, чтобы не позволить недоверенным компонентам осуществить доступ к данным, обрабатываемым или хранимым в системе, запрещенный ПБ.

Современные средства разграничения доступа, активного аудита и идентификации/аутентификации не позволяют достичь указанной цели. В подтверждение отмеченного обстоятельства обратим внимание на то, что политики безопасности, основанные на MultiLevel Security [3], традиционно являющейся стандартом в области построения гарантированно защищенных информационных систем, запрещают все информационные потоки на запись или на чтение между компонентами различных уровней решетки. По этой причине в терминах данной политики нельзя описать взаимодействие различных компонентов информационной системы в полном объеме.

Предлагаемый подход к решению поставленной задачи основан на выделении подмножества «безопасных» информационных потоков. Необходимо отметить, что проблема ослабления условий политики MultiLevel Security уже рассматривалась ранее, например, в работе [4] вводилось такое понятие, как «контролируемое рассекречивание». Однако, в своей статье автор не касался вопроса о практической

применимости построенной модели с точки зрения вычислительной сложности построенных алгоритмов.

В настоящей работе формально определяется свойство безопасности информационного потока, а также выводится необходимое и достаточное условие, позволяющее выполнять проверку свойства безопасности в режиме реального времени.

Рассмотрим указанный подход более подробно, опираясь на модель «клиент-сервер». Под термином «клиент» при этом будем понимать недоверенный компонент информационной системы, а под термином «сервер» — компонент информационной системы, в котором хранятся основные ее данные. Далее потребуются следующие обозначения.

- $O_1, \ldots, O_m$ — объекты, сохраняющие данные информационной системы.
- $X_1, \ldots, X_m$ — пространства допустимых состояний объектов.
- $f_1, \ldots, f_s$ — функции, определяющие внешний интерфейс информационной системы. В общем случае для функции f_k определен набор индексов $A_k \subseteq \{1, \ldots, m\}$ и $\beta_k \in A_k$, причем

$$f_k \colon \bigotimes_{t \in A_k} X_t \longrightarrow X_{\beta_k},$$

  где выражение $\bigotimes_{t \in A_k} X_t$ означает декартово произведение всех множеств, номер которых принадлежит набору A_k. Для произвольного подмножества P из области определения функции f_k определим множество $f_k(P)$ как полный образ P при отображении f_k.
- Для каждого объекта O_i определим конечную систему подмножеств $\mathfrak{B}_i$ пространства допустимых состояний X_i. Каждое $b \in \mathfrak{B}_i$ ассоциируется с некоторым логическим свойством соответствующего объекта. Систему $\mathfrak{B}_i$ будем считать замкнутой относительно операций объединения, пересечения и дополнения (что соответствует логическим операциям дизъюнкции, конъюнкции и отрицания соответственно). Подобная система множеств в функциональном анализе носит название алгебры множеств.

Событиями рассматриваемой системы являются операции вызова функции (обозначение: $[f_k]$) и операция проверки свойства объекта (обозначение: $[read\, O_i.b]$, причем $b \in \mathfrak{B}_i$). Множество всех событий системы обозначим Ev.

Сценарием вычислений в информационной системе будем называть автоматную функцию

$$Aut(ev, \sigma) = Aut : Ev \times \{true, false\} \longrightarrow Ev.$$

Здесь $\sigma \in \{true, false\}$ — результат операции ev, если ev — это событие проверки условия. В противном случае σ — произвольно, а Aut не зависит от этого аргумента. В любом случае операция $Aut(ev, \sigma)$ будет выполнена следующей за ev.

Будем считать, что задан некоторый сценарий вычислений. Клиент может определенным образом влиять на ход вычислений, а именно, он имеет право произвольно менять состояние некоторых объектов. Однако, поскольку клиент — лицо недоверенное, то будем считать, что для каждого объекта O_i в его алгебре $\mathfrak{B}_i$ выделены свойства, выполнение или невыполнение которых не должно зависеть от действий клиента. При этом множество выделенных свойств, которое будем обозначать $\mathcal{A}_i$, также считаем замкнутым относительно операций объединения, пересечения и дополнения. Таким образом для объекта O_i определена дополнительно *подалгебра* множеств (свойств) $\mathcal{A}_i$ основной алгебры $\mathfrak{B}_i$.

Замечание. Подход, основанный на выделении интересующих свойств объектов, а также алгебр свойств естественным образом соответствует общей модели знаний, изложенной более подробно в книге [5].

С учетом изложенного основная задача настоящей работы может быть сформулирована следующим образом: выяснить, выполнены ли для данного сценария вычислений указанные выше условия *невлияния* клиента. Отрицательный ответ на этот вопрос означал бы наличие в информационной системе потенциальной уязвимости.

Определение. *Информационный поток* — это множество наборов состояний объектов системы:

$$\mathcal{F} \subseteq \bigotimes_{t=1}^{m} X_t.$$

Данное определение на первый взгляд значительно отличается от обычного понятия потока, представленного в [6, 7], однако, необходимо заметить, что в указанных работах задача классификации информационных потоков не ставилась.

Рассмотрим семейство $\mathcal{A} = (\mathcal{A}_1, \ldots, \mathcal{A}_m)$, где $\mathcal{A}_i$ — алгебра свойств i-го объекта системы. Будем считать, что поток $\mathcal{F}$ *сохраняет* $\mathcal{A}$, если для любых $(\alpha_1, \ldots, \alpha_m), (\beta_1, \ldots, \beta_m) \in \mathcal{F}$, для любых i и $n_i \in \mathcal{A}_i$ из условия $\alpha_i \in n_i$ следует, что $\beta_i \in n_i$. Это означает, что клиент не может «заставить» систему изменить состояние объекта таким образом, что это можно было бы «распознать» с помощью свойств алгебры $\mathcal{A}$. Свойство инвариантности семейства алгебр относительно потока будем для краткости называть основным свойством.

Главная цель в контексте настоящей работы — проверить, сохраняет ли в каждый момент функционирования системы текущий поток $\mathcal{F}$ семейство $\mathcal{A}$.

Перечислим следующие основные операции над потоками.

- Слияние потоков $\mathcal{F}_1, \mathcal{F}_2 \colon \mathcal{F}' = \mathcal{F}_1 \cup \mathcal{F}_2$.
- Принятие условия $O_t.a \colon \mathcal{F}' = \{(a_1, \ldots, a_m) \in \mathcal{F} : a_t \in a\}$.
- Замещение. Для простоты рассмотрим пример. Пусть в системе три объекта O_1, O_2, O_3, функция f имеет два входных параметра O_1, O_2 и выходной параметр O_3. Пусть поток до вызова функцииf равнялся $\mathcal{F} = \{(\alpha_i, \beta_i, \gamma_i)\}$, тогда $\mathcal{F}' = \{(\alpha_i, \beta_i, f(\alpha_i, \beta_i))\}$ — поток после выполнения функции.

При выполнении различных операций над потоками их мощность может расти очень быстро (в худшем случае — экспоненциально от количества выполняемых операций). Принимая во внимание этот факт, не будем стремиться *вычислить* поток в каждой точке сценария. Покажем, что для выполнения цели достаточно в каждой точке знать всего лишь один элемент потока, как множества.

Утверждение 1. *Пусть $\mathcal{F}_1, \mathcal{F}_2$ — два потока, $\alpha = (\alpha_1, \ldots, \alpha_m) \in \mathcal{F}_1$, $\beta = (\beta_1, \ldots, \beta_m) \in \mathcal{F}_2$ — их представители. Поток $\mathcal{F}$ получается слиянием $\mathcal{F}_1, \mathcal{F}_2$. Тогда если $\mathcal{F}_1, \mathcal{F}_2$ сохраняют $\mathcal{A}$ и двухэлементный поток $\{\alpha, \beta\}$ сохраняет $\mathcal{A}$, то $\mathcal{F}$ сохраняет $\mathcal{A}$.*

Утверждение 2. *Пусть $\mathcal{F}'$ получен из $\mathcal{F}$ операцией принятия условия и $\mathcal{F}$ сохраняет $\mathcal{A}$. Тогда $\mathcal{F}'$ сохраняет $\mathcal{A}$.*

Доказательства обоих утверждений очевидны.

Пусть $\mathcal{D}$ — алгебра множеств с единицей X, а $\alpha \subseteq X$ — некоторое подмножество элементов. Считаем, что множество X конечно. Тогда рассмотрим $\mathcal{D}_\alpha = \{d \in \mathcal{D} : \alpha \subseteq d\}$ и положим

$$[\alpha] = [\alpha]_{\mathcal{D}} = \bigcap_{\beta \in \mathcal{D}_\alpha} \beta$$

— минимальный элемент алгебры, содержащий α. Отметим следующее утверждение.

Утверждение 3. *Пусть $p, q \subseteq X$ — произвольны, $\mathcal{D}$ — некоторая алгебра с единицей X. Тогда*

$$[p \cap [q]] = [p] \cap [q]. \tag{1}$$

Доказательство. Пусть алгебра $\mathcal{D}$ порождается конечным набором попарно непересекающихся непустых множеств $D_1, \ldots, D_r$ (такие всегда существуют в силу конечности X). Отсюда следует, что $D_i \in \mathcal{D}$ и $X = \bigsqcup_{i=1}^{r} D_r$. Пусть J, $K \subseteq \{1, \ldots, r\}$ — подмножества индексов такие, что $[p] = \bigsqcup_{j \in J} D_j$, $[q] = \bigsqcup_{k \in K} D_k$. Легко видеть, что тогда $p \cap D_j \neq \varnothing$, $q \cap D_k \neq \varnothing$ для $j \in J, k \in K$. В самом деле, если бы это было не так, то, рассмотрев $J \setminus \{j\}$ (или $K \setminus \{k\}$), получили бы меньшее множество алгебры $\mathcal{D}$, содержащее p (или q). Следовательно, если $s \in J \cap K$, то верны следующие утверждения:

- $D_s \subseteq [q] \implies (p \cap [q]) \cap D_s \neq \varnothing$, поэтому, в силу выбора разбиения $\{D_i\}$, заключаем, что $D_s \subseteq [p \cap [q]]$;
- $(D_s \subseteq [p]) \wedge (D_s \subseteq [q]) \implies D_s \subseteq [p] \cap [q]$.

С другой стороны, справедлива цепочка включений: $[p \cap [q]] \subseteq [[p] \cap [q]] = [p] \cap [q] \subseteq \bigsqcup_{s \in J \cap K} D_s$. Следовательно, $[p \cap [q]] = [p] \cap [q] = \bigsqcup_{s \in J \cap K} D_s$. □

Лемма. *Поток $\mathcal{F}$ сохраняет семейство $\mathcal{A}$ тогда и только тогда, когда для любых $(\alpha_1,\dots,\alpha_m)$, $(\beta_1,\dots,\beta_m)\in\mathcal{F}$ и любого i выполнено*

$$[\alpha_i]_{\mathcal{A}_i}=[\beta_i]_{\mathcal{A}_i}. \tag{2}$$

Доказательство. Всюду в доказательстве вместо $[\{x\}]_{\mathcal{A}_i}$ будем писать $[x]$. Пусть выполнено (2), и докажем инвариантность $\mathcal{A}$ относительно $\mathcal{F}$. Требуется проверить, что если $n_i\in\mathcal{A}_i$, то $\alpha_i\in n_i \iff \beta_i\in n_i$. Очевидно имеем:

$$\begin{aligned}\alpha_i\notin n_i \iff [\alpha_i\cap n_i]=\varnothing \overset{(1)}{\iff} [\alpha_i]\cap n_i=\varnothing \overset{(2)}{\iff}\\ \overset{(2)}{\iff} [\beta_i]\cap n_i=\varnothing \overset{(1)}{\iff} [\beta_i\cap n_i]=\varnothing \iff \beta_i\notin n_i.\end{aligned} \tag{3}$$

Обратно, пусть $\mathcal{F}$ сохраняет $\mathcal{A}$. Докажем (2). Рассуждаем от противного. Пусть $[\alpha_i]\cap n_i=\varnothing$, но $[\beta_i]\cap n_i\neq\varnothing$. Тогда по (3) получим, что $\alpha_i\notin n_i$, а $\beta_i\in n_i$, что противоречит условию инвариантности $\mathcal{A}$. □

Скажем, что функция f_k *сохраняет* $\mathcal{A}$, если для любого потока $\mathcal{F}$, сохраняющего $\mathcal{A}$, результат операции замещения $\mathcal{F}$ при помощи f_k (поток $\mathcal{F}'$) сохраняет $\mathcal{A}$.

Теорема. *Для инвариантности семейства $\mathcal{A}$ относительно функции $f_k\colon \bigotimes_{\alpha\in A_k}X_\alpha\longrightarrow X_{\beta_k}$ необходимо и достаточно, чтобы для любых $a_\alpha\in X_\alpha$, $\alpha\in A_k$ выполнялось условие*

$$\big[f_k\big(\{a_\alpha\}_{\alpha\in A_k}\big)\big]_{\mathcal{A}_{\beta_k}}=\big[f_k\big(\{[a_\alpha]_{\mathcal{A}_\alpha}\}_{\alpha\in A_k}\big)\big]_{\mathcal{A}_{\beta_k}}. \tag{4}$$

Доказательство. Без ограничения общности считаем, что $\beta_k=1$.

Достаточность. Пусть $\mathcal{F}=\{(\gamma_{1,t},\dots,\gamma_{m,t})\mid t\in T\}$ — произвольный поток, сохраняющий $\mathcal{A}$. Тогда

$$\mathcal{F}'=\big\{\big(f_k\big(\{\gamma_{\alpha,t}\}_{\alpha\in A_k}\big),\gamma_{2,t},\dots,\gamma_{m,t}\big)\mid t\in T\big\}$$

— поток, полученный по правилу замещения. Достаточно доказать, что для $\mathcal{F}'$ выполняется условие (2), тогда по доказанной лемме получим, что $\mathcal{F}'$ сохраняет семейство $\mathcal{A}$. Поскольку для $\mathcal{F}$ условие (2) справедливо, а $\mathcal{F}'$ отличается от него только в первой компоненте, то достаточно рассмотреть случай $i=1$:

$$\big[f_k\big(\{\gamma_{\alpha,t_1}\}_{\alpha\in A_k}\big)\big]_{\mathcal{A}_1}=\big[f_k\big(\{\gamma_{\alpha,t_2}\}_{\alpha\in A_k}\big)\big]_{\mathcal{A}_1}.$$

Из условия инвариантности $\mathcal{A}$ относительно $\mathcal{F}$ следует, что

$$[\gamma_{\alpha,t_1}]_{\mathcal{A}_\alpha}=[\gamma_{\alpha,t_2}]_{\mathcal{A}_\alpha}=n_\alpha.$$

- Положим в условии (4) $a_\alpha:=\gamma_{\alpha,t_1}$. Тогда будем иметь:

$$\begin{aligned}\big[f_k\big(\{\gamma_{\alpha,t_1}\}_{\alpha\in A_k}\big)\big]_{\mathcal{A}_1}&=\big[f_k\big(\{[\gamma_{\alpha,t_1}]_{\mathcal{A}_\alpha}\}_{\alpha\in A_k}\big)\big]_{\mathcal{A}_1}=\\&=\big[f_k\big(\{n_\alpha\}_{\alpha\in A_k}\big)\big]_{\mathcal{A}_1}=N_1.\end{aligned}$$

- Положим в условии (4) $a_\alpha:=\gamma_{\alpha,t_2}$. Тогда будем иметь:

$$\begin{aligned}\big[f_k\big(\{\gamma_{\alpha,t_2}\}_{\alpha\in A_k}\big)\big]_{\mathcal{A}_1}&=\big[f_k\big(\{[\gamma_{\alpha,t_2}]_{\mathcal{A}_\alpha}\}_{\alpha\in A_k}\big)\big]_{\mathcal{A}_1}=\\&=\big[f_k\big(\{n_\alpha\}_{\alpha\in A_k}\big)\big]_{\mathcal{A}_1}=N_1.\end{aligned}$$

Необходимость. Зафиксируем произвольный набор $a_\alpha\in\mathcal{A}_\alpha$, $\alpha\in A_k$ и дополним его до полного набора $a_i\in\mathcal{A}_i$, $i=1,\dots,m$ произвольным образом. Рассмотрим следующий двухэлементный поток

$$\mathcal{F}:=\big\{(a_1,\dots,a_m),\ ([a_1]_{\mathcal{A}_1},\dots,[a_m]_{\mathcal{A}_m})\big\}.$$

По доказанной лемме он сохраняет семейство $\mathcal{A}$. Следовательно, поток

$$\mathcal{F}'=\big\{\big(f_k(\{a_\alpha\}_{\alpha\in A_k}),a_2,\dots,a_m\big),\ \big(f_k(\{[a_\alpha]_{\mathcal{A}_\alpha}\}_{\alpha\in A_k}),[a_2]_{\mathcal{A}_2},\dots,[a_m]_{\mathcal{A}_m}\big)\big\},$$

полученный из $\mathcal{F}$ по правилу замещения, тоже сохраняет $\mathcal{A}$. В частности, это означает, что

$$\big[f_k\big(\{a_\alpha\}_{\alpha\in A_k}\big)\big]_{\mathcal{A}_1}=\big[f_k\big(\{[a_\alpha]_{\mathcal{A}_\alpha}\}_{\alpha\in A_k}\big)\big]_{\mathcal{A}_1}.$$

Сравнивая последнее выражение с (4) и учитывая, что $\beta_k=1$, заключаем, что теорема полностью доказана. □

Доказанная теорема предоставляет средство проверки сценариев вычислений на выполнение условий невлияния недоверенного клиента. В распределенных информационных системах такое средство может быть использовано при описании политик разграничения доступа, основанных на отношениях доверия. При этом функция доверия задается описанием «фильтра» данных, передаваемых от одного компонента информационной системы к другому. В данной работе таким фильтром являлось семейство подалгебр свойств $\mathcal{A}$ объектов системы. Поскольку на структуру или размер алгебр не накладывалось никаких ограничений, то данная модель является общей в смысле указанного взаимодействия клиента с сервером.

Представленная в работе модель является более общей, чем упомянутая выше модель разграничения доступа MLS, поскольку не накладывает столь жестких ограничений на информационные потоки и с другой стороны остается достаточно строгой для доказательства инвариантности свойств безопасности вычислений в информационной системе.

Литература

[1] Васенин В. А. *Проблемы математического, алгоритмического и программного обеспечения компьютерной безопасности в Интернет* // Материалы конференции МаБИТ-03, 23–24 октября 2003 г. — М.: МЦНМО, 2004, с. 111–143.

[2] Васенин В. А., Галатенко А. В. *Математические модели распределенных компьютерных систем* // Материалы конференции МаБИТ-04, 28–29 октября 2004 г. — М.: МЦНМО, 2005, с. 91–99.

[3] Denning D. *A Lattice Model of Secure Information Flow* // Communication of ACM, 19:5, May 1976. — P. 236–243.

[4] Mantel H., Sands D. *Controlled Declassification based on Interactive Noninterference* // APLAS 2004, LNCS 3302, 2004. — P. 129–145.

[5] Чечкин А. В. *Математическая информатика.* — М.: «Наука», Физматлит, 1991.

[6] Moonen L. *Data Flow Analysis For Data Engineering.* — University of Amsterdam, 1996.

[7] Holloway G., Dimock A. *The Machine SUIF Bit-Vector Data-Flow-Analysis Library.* — Harvard University, July 2002.

Выявление аномального сетевого трафика на основе нейросетевой кластеризации векторов статистических показателей сетевых соединений

В. В. Корнеев, В. В. Райх

1 Введение

Для обнаружения компьютерных атак принято использовать две группы методов: основанные на знаниях и основанные на поведении. Методы, основанные на знаниях, выявляют заранее известные совокупности событий, возникающие при известных атаках и не появляющиеся ни при каком режиме нормального функционирования. Характерным представителем этих методов служит сигнатурный анализ, при котором возникающей при каждой атаке совокупности событий сопоставляется сигнатура. Данная сигнатура однозначно отличает каждую атаку от любой другой или нормального поведения контролируемой системы. Данный подход является простым в реализации, обладает высоким быстродействием и очень малыми ошибками 1-го рода (пропуск известных атак) и 2-го рода (ложная тревога). Однако он позволяет выявлять только известные атаки. Также существенный недостаток этого подхода — необходимость привлечения высококвалифицированных экспертов для выявления вариантов существующих или новых типов атак, формирования сигнатур и постоянного пополнения используемой базы сигнатур.

Методы, основанные на поведении, частично лишены указанных недостатков. Поведение системы характеризуется выбранным заранее набором признаков, временные ряды значений которых собираются датчиками системы мониторинга. Исходно, необходимо создать профиль нормального поведения контролируемой системы. Для этого, чаще всего, используют значения признаков, собранные при нормальной работе в течение определенного периода. В ходе функционирования значения признаков, собираемых за определенный период, сравниваются с созданным заранее профилем. Задача составления профиля поведения является достаточно трудоемкой и не имеет универсального решения. Дополнительной трудностью является также то, что сам профиль с течением времени может изменяться.

Поведенческие методы используют для выявления аномальной активности статистический и регрессионный анализ, а также интеллектуальные методы анализа данных, основным достоинством которых является способность к самообучению, позволяющая автоматизировать формирование и адаптацию профилей поведения контролируемых систем. Одним из таких методов служит кластеризация многомерных векторов признаков с использованием нейросетей. Применение кластеризации позволяет выявлять аномальное поведение без предварительно созданных экспертами выборок векторов признаков, соответствующих аномальному и нормальному поведению системы.

В данной работе представлены результаты исследований по сравнительному анализу производительности и эксплуатационных характеристик нейросетей, реализующих алгоритмы кластеризации SOM, ART1, ART2 [1, 2], разработке оригинальной нейронной сети адаптивного резонанса для вещественных векторов ART2M [3], анализу возможности применения нейросетей для выявления сетевых соединений, содержащих аномальную активность.

Работа частично поддержана грантом РФФИ 04-07-90010.

2 Существующие работы по выявлению аномальной активности посредством кластеризации

В работе [4] рассмотрены вопросы применения кластерного анализа для выявления аномальной активности. В серии экспериментов было показано, что при формировании профиля посредством кластеризации обучающей выборки методом k-средних (в качестве параметра используется ширина кластера) может быть достигнута средняя ошибка 2-го рода (ложная тревога) — $1,3 \div 2,3\%$ при средней ошибке 1-го рода (пропуск атак) $45 \div 60\%$. Приведенные результаты получены при выявления атак в сетевых сессиях, содержащихся в базе данных UCI KDD (`http://kdd.ics.uci.edu`), созданной исследовательской лабораторией Калифорнийского университета по заказу DARPA специально для сравнения эффективности поведенческих алгоритмов обнаружения атак. Однако применяемый метод кластеризации не позволяет адаптировать профиль к изменяемому поведению системы без проведения повторной кластеризации.

В работе [5] для формирования профиля и выявления аномального поведения применяются нейросети с парадигмой теории адаптивного резонанса ART1 и ART2, но формирование обучающей и тестовых выборок из базы данных UCI KDD выполнено с заменой символьных переменных на числовые значения, что ведет к повышенным ошибкам 1-го и 2-го рода.

В работе [6] для выявления аномального поведения предлагается использовать многоуровневые нейросети на базе парадигмы самоорганизующихся сетей Кохонена. Приведены достаточно хорошие значения достижимых ошибок 1-го и 2-го рода, однако сети Кохонена не допускают дообучения при изменении профиля поведения системы.

3 Сравнительный анализ нейросетевых алгоритмов кластеризации

3.1 Используемые нейросети

Из описанных в литературе [1, 2] нейронных сетей, которые применяются для кластеризации данных в условиях отсутствия сведений о свойствах многомерного пространства признаков, наиболее распространенными являются самоорганизующиеся карты Кохонена (Self-Organizing Map, SOM) и сети теории адаптивного резонанса или классификаторы Карпентера — Гроссберга (Adaptive Resonance Theory Network, ART), известные как нейропарадигмы ART1 (для бинарных векторов) и ART2 (для вещественных векторов).

В ходе экспериментов (см. ниже) не удалось подобрать параметры для сети ART2 такие, чтобы она выполнила кластеризацию с заранее известным числом кластеров множества случайно сгенерированных бинарных векторов. Поэтому была разработана модифицированная парадигма ART2M [3], реализующая непосредственную нормализацию входных векторов в первом слое и последующую обработку полученного вектора с учетом параметра близости. Модифицированная сеть состоит из двух слоев нейронов (рис. 1): входного (нормализующего), число нейронов в котором фиксировано и равно размерности векторов, и выходного (соревновательного), с переменным количеством нейронов, где каждому нейрону соответствует один класс объектов (кластеров).

Суть работы модифицированной сети адаптивного резонанса для вещественных векторов практически аналогична работе сети ART1 для бинарных векторов и заключается в следующем:

1. При поступлении очередного входного вектора I активируются разрешающие элементы G, входной вектор нормализуется в первом слое и по восходящим связям S передается во второй слой.

2. Второй слой осуществляет поиск наиболее подходящей для полученного вектора категории (нейрона-победителя), направляет ее параметры (веса нейронов) по нисходящим связям V первому слою, а также закрывает элемент G до того момента, пока не будет завершена обработка текущего вектора.

3. Первый слой направляет полученные данные в систему обучения, где предложенная вторым слоем категория проходит проверку на близость (в зависимости от значения соответствующе-

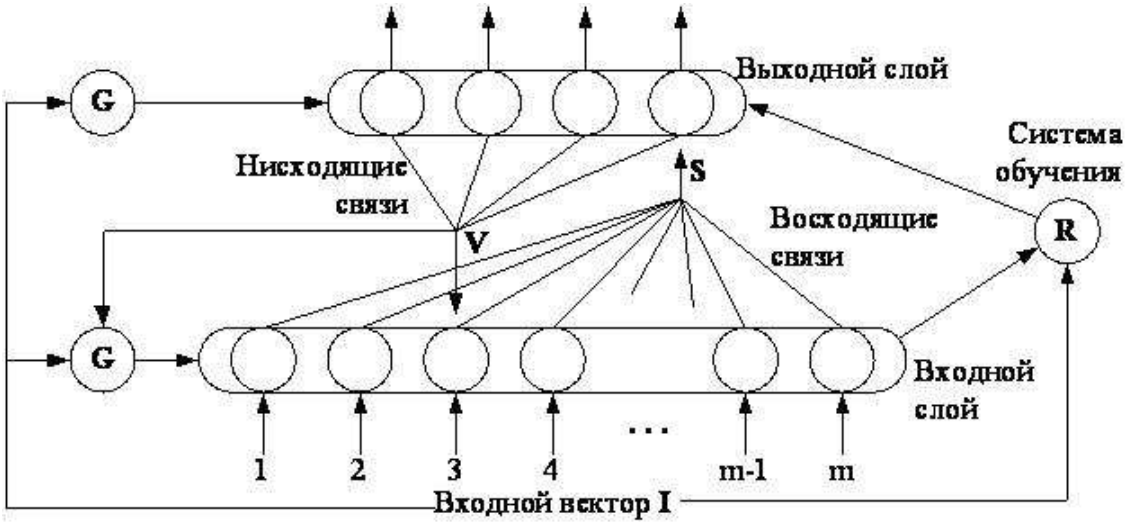


Рис. 1: Модифицированная нейронная сеть ART2M

го параметра). В случае успешной проверки параметры предложенной категории корректируются, и обработка завершается (случай резонанса). В противном случае посылается сигнал сброса предложенного результата R, и второй слой предлагает другую категорию (следующий нейрон-победитель) для проверки. Если все имеющиеся во втором слое категории исчерпаны, то на основе входного вектора во втором слое формируется новая категория.

Единственным параметром сети является порог близости ρ — действительное число из интервала $(0;1)$, чем ближе оно к 1, тем строже требование близости при проверке категорий.

Формально алгоритм обучения и функционирования модифицированной нейронной сети ART2M можно описать следующим образом.

Входными данными сети являются последовательно подаваемые вещественные вектора произвольной размерности m:

$$X^{(0)} = \left(x_1^{(0)}, x_2^{(0)}, \ldots, x_m^{(0)}\right).$$

В начале работы сеть включает m нейронов во входном слое и $n = 0$ нейронов выходного слоя.

При поступлении очередного вектора

$$X^{(0)} = \left(x_1^{(0)}, x_2^{(0)}, \ldots, x_m^{(0)}\right)$$

выполняются следующие операции:

1. Нормализация: $x_j = \dfrac{x_j^{(0)}}{\sqrt{\sum\limits_{l=1}^{m}\left(x_l^{(0)}\right)^2}}, \quad j = 1, \ldots, m.$
2. Все нейроны выходного слоя делаются активными.
3. Для всех активных нейронов вычисляется значение функции состояния как расстояние между вектором весов и входным вектором в некоторой метрика, например:
   - в евклидовом пространстве: $f_j(X) = \sqrt{\sum\limits_{i=1}^{m}(w_{ji} - x_i)^2}, \quad j = \overline{1,n};$
   - как угол между векторами: $f_j(X) = \arccos\left(\sum\limits_{j=1}^{m} w_{ji} \cdot x_i\right).$

4. Из всех активных нейронов выявляется нейрон-победитель:

$$N = \arg \min_{j=\overline{1,n}} \{f_j\}.$$

Если активных нейронов больше нет, то генерируется новый выходной нейрон с номером $n+1$, с весами:

$$w_{in+1} = x_i, \quad C_{n+1} = 1, \quad i = \overline{1,m}; \quad n = n+1,$$

где C_{n+1} — счетчик векторов, отнесенных к данному $n+1$ нейрону.

Переход на п. 7.

5. Нейрон-победитель проходит проверку на близость:

- в случае евклидовой метрики: $f_j \leqslant (1-\rho)$;
- в случае угла между векторами: $f_j \leqslant (1-\rho) \cdot \frac{\pi}{2}$.

Если условие выполняется, то переход на п. 6, иначе — нейрон-победитель становится неактивным и переход на п. 4.

6. Коррекция весов связей нейрона-победителя.

- для евклидовой метрики: $w_{ji} = w_{ji} + \frac{1}{C_j} \cdot (x_i - w_{ji})$;
- для угла между векторами: $w_{ji} = \frac{w_{ji} + \frac{1}{C_j} \cdot x_i}{\sqrt{\sum\limits_{l=1}^{m} \left(w_{jl} + \frac{1}{C_j} \cdot x_l\right)^2}}$.

$C_j = C_j + 1$, $i = 1...m$,

здесь C_j — количество векторов ранее отнесенных к данному нейрону.

7. Работа с очередным вектором закончена.

3.2 Постановка экспериментов

С целью выбора нейронной сети, наиболее эффективно решающей задачу формирования профиля и последующего выявления аномального поведения, были проведены исследования трех указанных выше нейропарадигм. При этом основное внимание было уделено следующим характеристикам:

- сложности подбора управляющих параметров;
- скорости обучения;
- производительности в процессе обработки входных векторов;
- компактности;
- эквивалентности.

Для исследования использовалась разработанная платформо-независимая программная библиотека НейроЭксперт [7].

Суть экспериментов заключалась в обучении сетей на тестовых выборках, представляющих собой совокупность случайным образом сгенерированных бинарных векторов размерности 10. Общее число векторов в выборке составляло 100 тыс., а количество классов варьировалось: 10, 100 и 1000.

Проводились замеры времени обучения нейросетей на описанных выше тестовых выборках и времени обработки этих же выборок обученными нейросетями. Для каждой выборки и каждого вида нейросетей проводилось 10 экспериментов, результаты которых затем усреднялись. Работы проводились на ПК с процессором Pentium IV 2,8 ГГц, 512 Мбайт ОЗУ под управлением ОС Linux Red Hat 9 (ядро 2.6.8).

3.3 Результаты экспериментов

По количеству и сложности подбора управляющих параметров предпочтение было отдано сетям ART1 и ART2M, для которых требуется указать только параметр близости ρ (vigilance) векторов, объединяемых в один класс. При этом, учитывая, что с использованием сетей ART1 обрабатываются бинарные вектора, оценить значение параметра на практике оказалось достаточно просто — путем расчета доли координат, при несовпадении значений которых, вектора будут считаться относящимися к разным классам.

Сети Кохонена более сложны в настройке, поскольку требуют подбора 4-х параметров: размера решетки нейронов, количества эпох обучения, начального и конечного значений коэффициента обучения.

Результаты *исследования скоростных характеристик* нейросетей представлены в табл. 1 и 2 [8, 9]. В ходе испытаний обнаружились следующие особенности:

1. Сеть ART1 успешно обучалась на всех представленных множествах.
2. Сеть Кохонена выделяла заданное число классов только при исходном количестве нейронов в решетке, значительно превосходящем (более 50%) количество классов (см. примечание табл. 1).
3. Сеть ART2M успешно обучалась на всех представленных множествах.

По результатам экспериментов были сделаны следующие выводы:

1. По скорости обучения сети ART для различных типов векторов практически эквивалентны и значительно превосходят по этому показателю сети Кохонена.
2. По скорости обработки данных после обучения все сети эквивалентны.

Таблица 1: Скоростные характеристики обучения нейронных сетей

Вид сети	Параметры	Время обучения		
		10 классов	100 классов	1000 классов
ART1	$\rho = 1$	2 сек.	18–19 сек.	48 мин.
ART2M	$\rho = 1$	2 сек.	19–20 сек.	48 мин.
SOM	Количество эпох 40	30 сек.	730–740 сек.	193 мин.

Примечание: при обучении сети Кохонена размеры решетки нейронов были следующими: для 10 классов — 4×4, для 100 классов — 15×15, для 1000 классов — 60×60.

Таблица 2: Скоростные характеристики эксплуатации нейронных сетей

Вид сети	Время обработки выборки, сек.		
	10 классов	100 классов	1000 классов
ART1	2 сек.	18–19 сек.	52 мин.
ART2M	2 сек.	19–20 сек.	52 мин.
SOM	2 сек.	19–20 сек.	52 мин.

С точки зрения компактности получаемого результата, который оценивался количеством нейронов в обученной сети, все три рассмотренных нейропарадигмы продемонстрировали свою зависимость от параметров настройки. Для сетей ART получаемый объем зависит от близости к единице параметра близости ρ, а в сетях Кохонена — от первоначального размера решетки нейронов. Таким образом, компактность получаемого профиля поведения, которым и является сама нейросеть, может регулироваться пользователем исходя из соображений требуемого качества кластеризации. Отметим только, что в данном случае мы имеем дело с компромиссной задачей. Точность кластеризации ведет к увеличению объема нейросети, что в свою очередь снижает ее производительность.

Учитывая существенную разницу в скорости обучения сетей Кохонена и ART2M, были проведены дополнительные эксперименты, связанные с выяснением эквивалентности получаемого результата.

Для этого обе нейросети обучались на одной и той же выборке векторов, а затем веса нейронов одной сети использовались в качестве входных векторов для другой. По итогам обработки определялась доля нераспознанных векторов. Такой подход основан на предположении, что если результаты кластеризации обеими сетями эквивалентны, то центры гипершаров, формируемых в процессе обучения, будут достаточно близки. Иными словами, центры гипершаров одной сети должны попадать в соответствующие области другой.

Обучение на совокупности векторов, координаты которых сгенерированы случайным равновероятным образом в диапазоне $(0;1)$, не дало результатов, поскольку, вследствие равномерного распределения, вектора от эксперимента к эксперименту распределялись по нейронам случайным образом. Поэтому была использована обучающая выборка, сформированная на базе измерения статистических показателей сетевого трафика. Параметры сетей были подобраны таким образом, чтобы число нейронов после обучения составило более 1000 (для сети Кохонена получилось 1940, для сети ART2M — 1360). При взаимной обработке векторов весов нейронов каждой из сетей количество ошибок составило: для сети ART2M — 16 (0,8%), для сети Кохонена — 9 (0,7%). Таким образом, в рамках сделанного предположения можно считать, что результаты кластеризации векторов обеими сетями являются эквивалентными.

На основе представленных результатов сравнения свойств нейросетей можно сделать следующие выводы:

1. Предпочтительными для решения задач выявления аномального поведения на основе кластеризации являются сети адаптивного резонанса, как наиболее быстродействующие при обучении и простые в подборе параметров.

2. Выбор между нейропарадигмами ART1 и ART2M может делаться в сторону ART1 при наличии четких правил бинаризации входных вещественных векторов (например, на основе пороговых значений) или в сторону ART2M — в противном случае.

3. Основной недостаток сетей адаптивного резонанса — возможность «враждебного» переобучения в ходе эксплуатации — может быть преодолен реализацией в соответствующем ПО возможности отключения режима дообучения сети, либо применением двух сетей (дообучаемой и недообучаемой), одна из которых будет выполнять роль ограничителя.

4 Анализ возможности выявления сетевых соединений, содержащих аномальную активность

4.1 Исходные данные для экспериментов и их предобработка

В качестве исходных данных для экспериментов была использована база данных UCI KDD, описывающая около 5 млн. сетевых сессий. Каждая запись о сетевом соединении представляет собой строку из 42 значений, разделенных запятыми. Все значения можно разделить на четыре категории: текстовые строки, флаговые значения (возможные значения 0 и 1), количественные показатели (возможные значения — не меньше нуля), статистические показатели (возможные значения в интервале от 0 до 1). Для каждой из сессий указан тип обнаруженной в ней атаки либо ее отсутствие.

Для проводимых исследований использовались данные только двух файлов: kddcup_data.gz — архив полных данных обо всех сетевых соединениях и kddcup_data_10_percent.gz — архив выборки десятой части всех данных, которую можно использовать для составления профилей поведения, в частности, для обучения нейросетей.

Поскольку анализ данных предполагалось вести с использованием нейросетей, все значения необходимо было привести к единому интервалу. Для этого была выполнена следующая последовательность действий:

1. Исключение показателей, являющихся текстовыми строками.

2. Разделение выборки на группы по признаку используемого протокола транспортного и прикладного уровней.

3. Исключение неинформативных флаговых и количественных показателей для каждого из протоколов.

4. Нормализация количественных показателей для каждого из протоколов с целью приведения значений различных признаков во входном векторе нейросети к одному интервалу.

5. Нормализация векторов признаков путем приведения их к 1 длине.

Текстовые значения имеют четыре показателя из 42-х: protocol — название протокола сетевого или транспортного уровней (tcp, udp, icmp), service — название протокола прикладного уровня (http, ftp, telnet и др.), flag — флаги соединения, type — тип соединения, здесь указывалось normal, если соединение не несло атаку, или название класса атаки в противном случае. Так как в качестве целей исследования распознавание классов компьютерных атак не входило, то все соединения, помеченные как содержащие атаку, в дальнейшем рассматривались как один класс аномальных соединений. Поскольку перекодировка текстовых значений в количественные не несет в себе дополнительной статистической информации, все перечисленные показатели в дальнейшем использовались только в качестве источника априорных сведений для разделения исходных данных по группам протоколов, а векторов соответствующих признаков — на нормальные и аномальные при обучении нейросетей.

Важность отдельного формирования сети для каждого символьного значения показывается в следующем эксперименте.

При простой замене символьного значения, обозначающего протокол, на, например, порядковый номер, резко возрастает ошибка. Независимо от количества кластеров в одни и те же классы попадали соединения, как несущие атаку, так и нормальные, что демонстрируется результатами кластеризации, приведенными в табл. 3.

Таблица 3: Результаты кластеризации при работе сети с символьными параметрами

Номер кластера	1	2	3	4	5	6	7	8	9
Количество соединений, соответствующих атакам	102	125	4	58	100	103	43	181	1
Количество соединений, соответствующих нормальной активности	407	407	39	180	354	455	148	630	33

В табл. 4 представлены результаты анализа информативности показателей соединений для различных протоколов. Критерием неинформативности в данном случае являлось равенство всех значений показателя константе (в основном 0) как для нормальных, так и для аномальных соединений.

Некоторые показатели в табл. 4 отмечены как исключенные по иным причинам. Поясним их подробнее.

Показатель 1 (продолжительность сеанса в секундах) в базе тестовых данных представлен с недостаточной точностью (большинство сеансов длилось менее 1 секунды, и поэтому им была приписана нулевая длительность). В реальной системе обнаружения атак он может быть измерен более точно и нести в себе некоторую информацию, полезную для выявления атак.

Показатели под номерами 2–4 и 42 исключены как текстовые (см. выше), хотя их информация косвенным образом все же использовалась при проведении экспериментов.

Показатель 6 был исключен при приведении количественных значений к единому интервалу измерений. Вместо двух показателей 5 и 6 для анализа использовалось более интересное в статистическом плане отношение числа переданных в сессии байтов к числу полученных (отношение значения показателя 5 к значению показателя 6, в табл. 6 указано как показатель 5).

Показатели 30 и 35 были исключены как избыточные по отношению к показателям 29 и 34, поскольку в парах они составляют полную группу событий (доли обращений к тому же хосту или сервису и

к остальным хостам или сервисам, что в сумме составляет 100%), что при статистическом анализе малоинформативно.

Таким образом, на основе оставшихся 32 показателей сетевых соединений можно сделать однозначный вывод: с точки зрения размерности и состава признакового пространства раздельное рассмотрение сетевых соединений по различным протоколам является полностью оправданным, поскольку позволяет снизить исходную размерность на 10–60%.

Таблица 4: Результаты анализа информативности показателей соединений (Знаком «+» помечены информативные показатели, знаком «*» показатели, исключенные по иным причинам, все остальные показатели признаны неинформативными.)

№	Название показателя	Протоколы							
		Все	icmp	udp	tcp	http	smtp	ftp	telnet
1	duration	*	*	*	*	*	*	*	*
2	protocol_ type	*	*	*	*	*	*	*	*
3	service	*	*	*	*	*	*	*	*
4	flag	*	*	*	*	*	*	*	*
5	src_bytes	+		+	+	+	+	+	+
6	dst_bytes	*	*	*	*	*	*	*	*
7	land								
8	wrong_ fragment	+	+	+					
9	urgent	+							+
10	hot	+			+	+	+	+	+
11	num_ failed_ logins	+			+			+	+
12	logged_in	+			+	+	+	+	+
13	num_ compromised	+			+		+		+
14	root_shell	+			+		+	+	+
15	su_ attempted	+			+		+		+
16	num_root	+			+		+	+	+
17	num_file_ creations	+			+		+		+
18	num_shells	+			+			+	+
19	num_ access_files	+			+	+	+	+	+
20	num_ outbound_ cmds	+			+				
21	is_host_ login								
22	is_guest_ login	+			+			+	
23	count	+	+	+	+	+	+	+	+
24	srv_count	+	+	+	+	+	+	+	+
25	serror_rate	+	+	+	+	+	+	+	+
26	srv_ serror_rate	+			+	+	+	+	+
27	rerror_rate	+	+		+	+	+	+	+

28	srv_ rerror_rate	+			+	+	+	+	+
29	same_srv_ rate	+	+	+	+	+	+	+	+
30	diff_srv_ rate	*	*	*	*	*	*	*	*
31	srv_diff_ host_rate	+	+	+	+	+	+	+	+
32	dst_host_ count	+	+	+	+	+	+	+	+
33	dst_host_ srv_count	+	+	+	+	+	+	+	+
34	dst_host_ same_srv_ rate	+	+	+	+	+	+	+	+
35	dst_host_ diff_srv_ rate	*	*	*	*	*	*	*	*
36	dst_host_ same_src_ port_rate	+	+	+	+	+	+	+	+
37	dst_host_ srv_diff_ host_rate	+	+	+	+	+	+	+	+
38	dst_host_ serror_rate	+	+	+	+	+	+	+	+
39	dst_ host_srv_ serror_rate	+			+	+	+	+	+
40	dst_host_ rerror_rate	+	+	+	+	+	+	+	+
41	dst_host_ srv_rerror_ rate	+			+	+	+	+	+
42	type	*	*	*	*	*	*	*	*
Всего информативных показателей		32	14	14	30	21	26	26	29

При формировании признакового пространства для нейросетевого анализа важным является не только состав и количество признаков, но и их нормировка. Поскольку большинство статистических признаков являются отношениями, то в качестве базового интервала был выбран отрезок $[0;1]$. Признаки, соответствующие количественным характеристикам (например, показатель 23, представляющий количество новых обращений к тому же хосту при окне в 2 секунды), нормировались путем деления своего значения на максимальное среди всех встречающихся в выборке. Так, например, были скорректированы значения следующих показателей: 13, 16, 23, 24. Полученные таким образом вектора приводились к единичной длине в евклидовом пространстве и снабжались специальной меткой, обозначающей тип описываемого ими соединения: нормальное или аномальное.

4.2 Результаты экспериментов по выявлению атак

Основным вопросом первого этапа исследования являлось выяснение того, насколько сети теории адаптивного резонанса способны адекватно выполнять разделение векторов, описывающих нормальные и аномальные сетевые соединения. Всего было проведено 80 экспериментов (по 10 для каждого протокола), на основе которых для каждой выборки были определены значения параметра близости ρ, приемлемые как с точки зрения качественных, так и скоростных характеристик. При этом приемлемыми считались значения для ошибки первого рода порядка 10^{-3} (десятые доли процента), для

ошибки второго рода порядка 10^{-2} , а для скорости — порядка 10^3 векторов/сек. Измерения проводились на компьютере с процессором Pentium IV 2,6 ГГц, 512 Мбайт ОЗУ под управлением ОС Linux с ядром 2.6.8.

Для каждого нейрона по результатам обучения вычислялась ошибка кластеризации (в %) как отношение числа «чужих» векторов, к общему числу векторов, отнесенных к данному нейрону (например, число аномальных векторов, отнесенных к нормальному нейрону, и наоборот, число нормальных векторов, отнесенных к аномальному нейрону). Для всей нейросети в целом подсчитывалось взвешенное среднее ошибки нормальных и аномальных нейронов (ошибка каждого нейрона учитывалась с весом, зависящим от числа векторов, отнесенных к нейрону).

В табл. 5 приведены лучшие с точки зрения скоростных и качественных характеристик результаты обучения нейросетей для различных протоколов (значения ошибок указаны в процентах).

Таблица 5: Параметры, иллюстрирующие способность разделения нейросетями нормальных и аномальных векторов

Соединения	Все	icmp	udp	tcp	http	smtp	ftp	telnet
Размерность векторов после исключения неинформативных показателей	32	14	14	30	21	26	26	29
Параметр близости ρ	0.75	0.90	0.65	0.70	0.70	0.80	0.80	0.85
Скорость обучения, век./с	81	3 458	5 404	257	3 571	3 997	10230	3 828
Нормальных нейронов	631	76	70	348	152	120	6	83
Аномальных нейронов	182	76	12	117	18	13	17	50
Ср. взвеш. ошибка норм. н.	0.29	1.50	0.19	0.18	0.02	0.01	3.98	0.44
Ср. взвеш. ошибка аном. н.	0.05	0.01	0.00	0.14	0.12	0.00	22.32	0.00

В ходе проведения экспериментов выяснилось, что выбор параметра близости ρ при обучении нейросетей представляет собой компромиссную задачу, где противоречивыми характеристиками являются скорость обработки данных с одной стороны и величина ошибки с другой. Приемлемый результат может быть получен не обязательно при значении ρ, близком к 1, он достигается при значении ρ, при котором наступает определенное насыщение в значениях ошибки кластеризации. В качестве критерия выбора значения параметра близости для обучения нейросетей можно предложить такое значение, при котором количество нейронов в сети не превышает нескольких десятков, что позволяет достичь приемлемой скорости (3–5 тыс. векторов/сек.) обработки данных, с одной стороны, и допустимой величины ошибки (десятые или сотые доли процента) — с другой. В табл. 6 приведены соответствующие результаты, на основе которых были выбраны значения параметра близости ρ, приемлемые как с точки зрения качественных, так и скоростных характеристик.

Таблица 6: Зависимость результатов обучения нейросетей от параметра близости для соединений по протоколу tcp

Параметр близости ρ	0.55	0.60	0.65	**0.70**	0.75	0.80	0.85	0.90
Скорость обучения, век./с	1 959	852	410	**257**	150	76	31	9
Нормальных нейронов	139	178	249	**348**	505	738	1 284	2 554
Аномальных нейронов	67	75	95	**117**	134	177	226	322
Ср. взвеш. ошибка норм. н.	2.03	1.82	0.33	**0.18**	0.17	0.16	0.14	0.11
Ср. взвеш. ошибка аном. н.	0.06	0.05	0.89	**0.14**	0.16	0.13	0.12	0.12

4.3 Оценка характеристик нейросетевых средств выявления аномалий

Анализ данных таблиц 5 и 6 показывает, что сети ART2M способны с приемлемым качеством разделять вектора признаков, описывающих нормальные и аномальные сетевые соединения. Поэтому на втором этапе исследований была проведена оценка применимости данных методов с практической точки зрения. Критерием применимости следует считать достаточное быстродействие обученной нейронной сети, работающей с приемлемыми значениями ошибок первого и второго рода.

Суть экспериментов заключалась в следующем.

1. Для каждого из протоколов (icmp, udp, tcp/http, tcp/smtp, tcp/ftp и tcp/telnet) на основе файла kddcup_data_10_percent.gz обучалось по две нейронных сети: одна на векторах, описывающих нормальные соединения, вторая — на векторах, описывающих аномальные соединения.

2. С использованием полученных нейросетей обрабатывались полные данные, содержащиеся в файле kddcup_data.gz. При этом результат обработки трактовался соответствующим образом. Для нейросети, обученной на нормальных векторах, обработанный вектор считался также нормальным, если он попадал в какой-либо из уже имевшихся нейронов, и аномальным — в противном случае. Для нейросети, обученной на аномальных векторах, обработанный вектор считался нормальным, если он не попадал ни в какой из уже имевшихся нейронов, и аномальным — в противном случае. Дообучение в процессе обработки не использовалось.

3. Для каждой нейросети подсчитывалась скорость обработки как отношение числа обработанных векторов к времени работы, а также ошибки первого и второго рода относительно обнаружения аномального поведения. Ошибка первого рода (true negative) рассчитывалась как отношение числа аномальных векторов, не обнаруженных сетью, к общему числу аномальных векторов в выборке. Ошибка второго рода (false positive) рассчитывалась как отношение числа нормальных векторов, ошибочно отнесенных сетью к аномальным, к общему числу нормальных векторов.

Полученные результаты представлены в табл. 7 и 8.

5 Заключение

Распознавание сетью ART2M аномальных сетевых соединений зависит от репрезентативности и достаточности объема обучающей выборки (см. низкий результат по протоколам telnet и ftp и доста-

Таблица 7: Состав данных для обучения и тестирования нейросетей

Протокол	Параметр близости	Обучающая выборка		Полные данные	
		Нормальных векторов	Аномальных векторов	Нормальных векторов	Аномальных векторов
icmp	0.90	1 288	282 314	12 763	2 820 782
udp	0.65	19 177	1 177	191 348	2 940
http	0.70	61 885	2 506	619 045	5 092
smtp	0.80	4 141	1 348	95 371	1 183
ftp	0.80	9 598	125	41 914	3 997
telnet	0.85	219	294	2 227	2 050

Таблица 8: Результаты обработки полных данных нейросетями

Протокол	Обучение на нормальных векторах			Обучение на аномальных векторах		
	Скорость обработки, вект./сек.	Ошибка 1-го рода, %	Ошибка 2-го рода, %	Скорость обработки, вект./сек.	Ошибка 1-го рода, %	Ошибка 2-го рода, %
icmp	6 811	0.038	5.727	4 810	0.013	12.497
udp	10 793	39.116	0.184	97 144	1.497	6.320
http	2 600	1.375	0.179	56 739	1.532	0.030
smtp	5 081	0.845	1.101	96 554	6.002	0.000
ftp	4 591	23.242	2.560	22 955	2.977	28.740
telnet	4 277	0.049	41.311	4 277	7.561	0.000

точно высокий результат по протоколу http в табл. 8), и действительно сессионного характера анализируемых сетевых взаимодействий (см. низкий результат в табл. 8 по протоколам icmp и udp, предполагающим взаимодействие в режиме запрос-ответ, и более приемлемый результат по протоколам http и smtp, организующим более продолжительные соединения).

Для достижения приемлемой скорости обработки данных для каждого из протоколов должна обучаться своя нейронная сеть (см. разницу в столбцах «все» и для отдельных протоколов в табл. 4). При этом снижение быстродействия при отсутствии предварительного разделения данных по протоколам обусловлено не только более высокой размерностью обрабатываемых векторов, но и большим числом порождаемых во время обучения нейронов, на которые ложится дополнительная нагрузка по учету не только специфики нормального и аномального поведения, но и сетевых протоколов в целом.

Обученная на репрезентативной выборке примеров нейросеть для прикладных протоколов, реализующих обмен данными с установлением относительно продолжительных во времени сессий, позволяющих набрать достаточную статистику, в частности http и smtp, обеспечивает при эксплуатации значения ошибок первого рода порядка 10^2 и второго рода порядка 10^{-4} при быстродействии около 10^4 векторов/сек., что соответствует средней загрузке сети FastEthernet.

Литература

[1] Корнеев В. В., Гареев А. Ф., Васютин С. В., Райх В. В. Базы данных. Интеллектуальная обработка информации. 2-е изд. М.: Нолидж, 2001, 496 с.

[2] Круглов В. В., Борисов В. В. Искусственные нейронные сети. Теория и практика. М.: Горячая линия — Телеком, 2001, 382 с.

[3] Корнеев В. В., Райх В. В. Нейросетевой алгоритм кластеризации на базе модифицированной сети адаптивного резонанса для вещественных векторов // Материалы Второй Всероссийской научно-технической конференции «Методы и средства обработки информации» МСО-2005, 5–7 октября 2005 г., г. Москва.

[4] Portnoy L., Eskin E., Stolfo S. Intrusion detection with unlabeled data using clustering. In Proceedings of ACM CSS Workshop on Data Mining Applied to Security (DMSA-2001).

[5] Morteza Amini, Rasool Jalili. Network-Based Intrusion Detection Using Unsupervised Adaptive Resonance Theory (ART).

[6] Peter Lichodzijewski, A. Nur Zincir-Heywood, Malcolm I. Heywood. Dynamic Intrusion Detection Using Self-Organizing Maps. Faculty of Comp. Science Dalhousie University Halifax, NS.

[7] Отчет о работах в рамках гранта РФФИ 04-07-90010 «Исследование методов обнаружения аномальной активности в распределенных компьютерных системах и разработка системы обнаружения компьютерных атак, сочетающей сигнатурный и интеллектуальный анализ данных» за 2004 год.

[8] Райх В. В. Применение нейронных сетей Кохонена для решения задач кластеризации в процессе мониторинга информационной безопасности // Математика и безопасность информационных технологий. Материалы конференции в МГУ 23–24 октября 2003 г. М.: МЦНМО, 2004, с. 321–327.

[9] Райх В. В. Исследование свойств нейронных сетей Кохонена и адаптивного резонанса применительно к задачам мониторинга информационной безопасности // Информационная безопасность: Материалы VI Международной научно-практической конференции, 1–7 июля 2004 г., Таганрог: Изд-во ТРТУ, 2004, с. 193–195.

К вопросу о средствах ОС Linux для управления доступом при использовании ролевых политик безопасности

К. А. Шапченко

1 Введение

Управление доступом в современных операционных системах является одной из важнейших задач, решение которых во многом определяет уровень защищенности информационно-вычислительных комплексов в условиях их функционирования в небезопасной среде [1]. По данной тематике существует большое число программных решений, из которых многие не подкреплены теоретическими исследованиями. Настоящая работа посвящена анализу функциональных свойств и архитектурно-технологических решений подсистем управления доступом в ядре операционной системы (ОС) Linux. В работе рассматриваются следующие программные средства управления логическим разграничением доступа (ЛРД) в ОС Linux:

- Security Enhanced Linux (SELinux) [2];
- Rule-Set Based Access Control (RSBAC) [3];
- grsecurity [4].

Большой интерес представляют компоненты указанных подсистем управления доступом, поддерживающие ролевые политики — SELinux/Type Enforcement (TE), RSBAC/Role Compatibility (RC) [5] и grsecurity/Role-Based Access Control (RBAC). Выбор ролевых политик как базового объекта исследования обусловлен возможностью в их рамках более адекватного разделения привилегий и изоляции процессов в соответствии с потребностями современных практически значимых компьютерных систем.

Инструментальные средства, поддерживающие механизмы ролевых политик информационной безопасности с использованием соответствующих операционных систем могут быть логически сложными и содержать значительный объем как программного кода, так и конфигурационных данных. Кроме того, система управления ЛРД может иметь средства автоматической настройки, например, таких как система обучения в grsecurity или средства построения политики по журналам аудита в SELinux. Однако, заданная с их помощью политика не всегда подходит для применения в конкретной системе, а анализ ее может вызывать затруднения из-за плохо-структурированного ее описания.

В крупных распределенных системах вполне закономерно использование различных программно-технических средств обеспечения информационной безопасности, в том числе, — средств управления ЛРД. В случае инструментальных средств управления ЛРД гетерогенность операционной среды распределенной системы приводит к задачам сравнения различных моделей управления доступом, выяснения взаимосвязей между ними. С практической точки зрения, например, для проведения аудита безопасности информационных технологий, интересна задача обзора общей политики управления ЛРД в распределенной системе. Удобным и логичным решением здесь представляется выражение различных политик управления доступом в терминах одной, унифицированной модели.

С учетом изложенных фактов, можно выделить три следующих класса практически значимых задач.

1. Построение формальных моделей управления доступом, реализуемых теми или иными программно-техническими средствами управления ЛРД.

Решение такого класса задач в рамках настоящей работы носит вспомогательный характер, обусловленный необходимостью формальных моделей для решения других классов задач. Однако, с практической точки зрения формальное описание позволяет более эффективно реализовать инструментальное средство управления ЛРД и произвести настройку конфигурации для конкретной задачи. Таким образом, повышение оценочного уровня доверия возможно только при условии существования формальной модели ЛРД.

2. Выражение одних моделей управления ЛРД в терминах других аналогичных моделей.

Решение данного класса задач позволит, с одной стороны, сравнить средства управления ЛРД с точки зрения «выразительности», то есть возможности описания той или иной конструкции конфигурации средства управления доступом. С другой стороны, использование некоторого языка, позволяющего выразить конструкции используемые при конфигурировании различных средств управления доступом в гетерогенной среде, предоставит возможность для обзора общей политики ЛРД в распределенной системе. Кроме того, приняв некоторые ограничения на модели, возможна реализация взаимно однозначного преобразования конфигурационных данных между различными средствами управления доступом, что позволит автоматизированно переходить при необходимости от одного средства управления ЛРД к другому.

3. Проверка свойств политики управления ЛРД на основе конфигурационных данных используемых инструментальных средств управления доступом.

На этом направлении исследований необходима разработка языка спецификаций свойств системы управления ЛРД и механизмов проверки выполнения спецификаций в низкоуровневой модели, описывающей поведение системы управления доступом с учетом заданных конфигурационных данных. В настоящей работе в качестве базы для языка спецификаций предлагается линейная временна́я логика. Описание модели поведения системы как машины состояний позволяет использовать для проверки свойств использовать стандартные средства проверки моделей такие, как, например, NuSMV [6]. Похожие исследования проведены, например, в [8]. Предлагаемая в настоящей работе модель направлена также на исследование информационных потоков, однако при этом она шире модели, описанной в [8].

Необходимо отметить, что используя подходы к решению задач предыдущего класса, можно проверять свойства политик управления доступом в гетерогенных системах, использующих различные средства управления ЛРД. Для этого достаточно преобразовать конфигурации используемых средств управления доступом к унифицированному описанию.

2 Описание исследуемых средств управления доступом

В данном разделе в краткой форме представлены базовые сведения об исследуемых инструментальных средствах управления ЛРД. Для каждого из трех рассматриваемых средств удается построить теоретико-множественную модель, отвечающую ролевой политике управления доступом, поддерживаемой этим средством.

2.1 SELinux/Type Enforcement

Security Enhanced Linux (SELinux) [2] — подсистема управления доступом, официально включенная в ядро Linux 2.6 и предоставляющая средства разграничения доступа для реализации ролевой (Type Enforcement — TE) и многоуровневой политик (Multi Level Security — MLS).

Подсистема управления доступом Type Enforcement (TE), реализованная в SELinux основана на сочетании двух используемых в ней моделей: Domain and Type Enforcement (DTE) и Role-Based Access Control (RBAC). Принцип действия DTE основан на разделении субъектов-процессов по доменам и объектов-данных по типам, а также на определении разрешенных доступов от домена к типу. RBAC применяется для привязки пользователей к доменам через роли, когда каждому пользователю доступно некоторое множество ролей, а каждой роли соответствует множество разрешенных доменов. Опишем основные понятия, используемые в рамках этих моделей, более подробно.

Доступ в рамках указанных моделей определяется как пара вида (класс объекта доступа, наименование доступа). Классы объектов доступа и наименования доступов жестко привязаны к способам

реализации системы в ядре ОС, тем не менее конфигурация применяемой модели разграничения доступа (далее, для краткости, — политики) содержит описание классов и наименований доступов для каждого класса. В конфигурации модели управления доступом (далее для краткости используется термин «конфигурация»), реализованной в системе SELinux, класс определяется строкой

```
class <имя класса>
```

Например:

```
class file
class dir
```

Классы, реализованные в системе, содержат

- общие классы: security, process, system, capability;
- классы объектов файловой системы: filesystem, file, dir, fd, lnk_file, chr_file, blk_file, sock_file, fifo_file;
- классы сетевых объектов: socket, tcp_socket, udp_socket, rawip_socket, node, netif, netlink_socket, packet_socket, key_socket, unix_stream_socket, unix_dgram_socket;
- классы объектов межпроцессного взаимодействия: sem, msg, msgq, shm, ipc.

Определение набора доступов для каждого класса в конфигурации обеспечивается двумя командами. Для определения общих прав доступов, встречающихся во многих классах используется команда

```
common <идентификатор> <множество прав доступов>
```

например,

```
common file { ioctl read write create getattr setattr lock relabelfrom
relabelto append unlink link rename execute swapon quotaon mounton }
```

Для определения прав доступа для конкретного класса с возможным наследованием множества общих прав доступов используется команда

```
class <имя класса> [inherits <идентификатор>] <множество прав доступов>
```

например,

```
class dir inherits file { add_name remove_name reparent search rmdir }
class file inherits file { execute_no_trans entrypoint }
```

Право доступа `entrypoint` следует отметить отдельно, так как при таком доступе к исполняемому файлу возможен переход в другой домен по правилам, описываемым ниже.

Далее, следует описать различные типы и домены, использующиеся в системе. Для этого вводится множество типов и набор атрибутов. Каждый атрибут выделяет подмножество типов, например, атрибут `domain` различает домены и остальные типы. Атрибут описывается в конфигурации следующим образом:

```
attribute <имя атрибута>;
```

Тип описывается командой

```
type <имя типа> <атрибуты типа>;
```

Например:

```
attribute domain;
type sshd_t, domain;
```

Правила установки типа объекта при его создании описываются с помощью ключевого слова `type_transition`:

```
type_transition <исходные типы> <целевые типы> : <классы> <новый тип>;
```

Здесь и далее в примерах предполагается, что в определении множества типов отдельный элемент означает множество из одного этого элемента, а набор атрибутов означает множество типов, обладающих любым атрибутом из набора.

Примеры двух вариантов использования такой команды:

```
type_transition sshd_t tmp_t : file sshd_tmp_t;
type_transition sshd_t shell_exec_t : process user_t;
```

В первом примере указывается, что объекты класса `file` создаваемые процессом из домена `sshd_t` в каталоге с типом `tmp_t` будут иметь тип `sshd_tmp_t`. Во втором примере описывается аналогичная ситуация с созданием процесса. В этом случае при запуске процессом из домена `sshd_t` исполняемого файла типа `shell_exec_t` созданный процесс (объект класса `process`) будет иметь тип `user_t`.

Для определения разрешенных доступов от одного типа (домена) к другому используется следующая конструкция:

```
<тип аудита> <исходные типы> <целевые типы> : <классы> <права>;
```

где `<тип аудита>` предоставляет следующие возможности:

- `allow` — регистрировать только попытки запрещенного доступа;
- `auditallow` — попытка доступа регистрируется всегда;
- `dontaudit` — попытка доступа никогда не регистрируется;

Пример использования:

```
allow sshd_t shell_exec_t : file { read execute entrypoint };
```

Для запрещения некоторых доступов в политике используется команда `neverallow`:

```
neverallow <исходные типы> <целевые типы> : <классы> <права>;
```

Например:

```
neverallow domain ~domain : process transition;
```

В этом примере запрещается переход от типа-домена к типу, не являющемуся доменом, при создании процесса. Конструкция `neverallow` применяется только на этапе компиляции политики и убирает указанные доступы из уже сформированной политики.

Для описания RBAC-части политики используются команды `user` и `role` следующего вида:

```
role <имя роли> types <разрешенные типы (домены)>;
user <имя пользователя> roles <разрешенные роли>;
```

Например:

```
role user_r types user_t;
user guest_u roles user_r;
```

Возможность перехода из одной роли в другую описывается следующим образом:

```
allow <имя роли> <разрешенные роли>;
```

Важным понятием в политике TE является контекст безопасности — тройка вида (пользователь, роль, тип). Каждая попытка доступа в системе рассматривается как попытка доступа от исходного контекста безопасности к целевому. Поскольку для объектов, не являющимися процессами, понятие роли не определено, вводится специальная роль `object_r`, которая всегда является ролью в контексте безопасности таких объектов. Заметим также, что для функционирования системы SELinux/TE необходима предварительная разметка файловых систем для присвоения объектам соответствующих контекстов безопасности.

С целью ограничения доступа, основываясь на двух контекстах безопасности, вводится механизм `constrain`, запрещающий доступ, если не выполнено некоторое заранее заданное условие:

```
constrain <классы> <права> <условное выражение>
```

Здесь условное выражение `expr` от переменных `u1`, `r1`, `t1`, `u2`, `r2`, `t2` определяется следующим образом:

```
expr ::= (expr) | not expr | expr and expr | expr or expr
         | u1 op u2 | t1 op t2 | r1 op r2
         | u1 op <имя пользователя> | t1 op <имя типа>
         | r1 op <имя роли>
         | u2 op <имя пользователя> | t2 op <имя типа>
         | r2 op <имя роли>

op   ::= == | !=
```

Текстовое представление политики (модели разграничения доступом), принимаемое на обработку компилятором политики состоит из определенных правил. Далее приведем теоретико-множественную модель как интерпретацию перечисленных выше конструкций.

Перечислим следующие базовые множества в предположении, что они конечны и непусты:

- C — множество классов объектов доступа, соответствует классам, определенным в политике;
- P — множество доступов, соответствует объединению множеств прав доступов для всех классов объявленных в политике;
- $\Gamma \subset C \times P$ — множество корректных пар (класс, право), соответствующих реальным парам, описанным в политике;
- U — множество пользователей в политике;
- R — множество ролей в политике, $r_o \in R$ — роль, соответствующая `object_r`;
- T — множество типов в политике;
- $D \subset T$ — множество доменов, соответствует типам в политике с атрибутом `domain`.

Далее следуют предикаты, показывающие взаимосвязь понятий политики.

- $\mu(u, r), u \in U, r \in R$ — в политике пользователю u разрешена роль r, причем выполнено свойство $\forall u \in U \mu(u, r_o)$ для корректной обработки объектной роли.
- $\rho(r, t), r \in R, t \in T$ — в политике роли r разрешен тип t, причем выполнено свойство $\forall t \in T \rho(r_o, t)$.
- $\alpha_\rho(r_1, r_2), r_1 \in R, r_2 \in R$ — разрешен переход из роли r_1 в роль r_2.
- $\alpha(t_1, t_2, c, p), t_1 \in T, t_2 \in T, c \in C, p \in P$ — в политике типу t_1 разрешен доступ (c, p) к типу t_2.
- $\chi_{(c,p)}(u_1, r_1, t_1; u_2, r_2, t_2)$ — отношение, соответствующее следующим правилам `constrain` в конфигурации:
  - если $(c, p) \notin \Gamma$, то этот предикат ложен;
  - если $(c, p) \in \Gamma$ и в политике нет правил `constrain` для доступа (c, p), то предикат истинен;

– если $(c,p) \in \Gamma$ и в конфигурации есть соответствующие правила, то истинность предиката есть принадлежность параметров предиката множеству, определяемому условным выражением из соответствующего правила `constrain`.

В таких обозначениях можно следующим образом записать предикат предоставления доступа (c,p) от контекста безопасности (u_1, r_1, t_1) к контексту (u_2, r_2, t_2):

$$\Delta_{(c,p)}(u_1,r_1,t_1;u_2,r_2,t_2) = \alpha(t_1,t_2,c,p) \wedge \rho(r_1,t_1) \wedge \rho(r_2,t_2) \wedge \mu(u_1,t_1) \wedge$$
$$\wedge\, \mu(u_2,t_2) \wedge \chi_{(c,p)}(u_1,r_1,t_1;u_2,r_2,t_2) \wedge$$
$$\wedge\, ((c,p) = (process, transition) \implies \alpha_\rho(r_1,r_2)).$$

2.2 RSBAC/Role Compatibility

Rule-Set Based Access Control (RSBAC) [3] — открытая (open-source) реализация программной компоненты обеспечения политик безопасности информационно-вычислительных систем под управлением ядра ОС Linux, основанная на архитектуре Generalized Framework for Access Control (GFAC). RSBAC включает в себя модули, реализующие отдельные модели логического разграничения доступа, такие как Mandatory Access Control (MAC) — составляющая многоуровневой политики, Role Compatibility (RC) — компонента управления доступом на основе ролевых политик [5], Access Control Lists (ACL) — компонента дискреционной политики, модуль Privacy Model (PM) — для реализации политики, ориентированной на конфиденциальность личных данных и некоторые другие.

В системе RSBAC не предусмотрены текстовые конфигурационные файлы, как это сделано в SELinux. Однако, опираясь на описание модуля ролевой политики Role Compatibility (RC) и на предоставляемые системой средства администрирования можно построить теоретико-множественную модель разграничения доступа в рамках рассматриваемой политики. Рассмотрим предложенные автором основные ее положения.

В ходе задания модели управления доступом (политики) описываются пользователи, роли и типы объектов. Множество пользователей будем обозначать через U, множество ролей через R. Для каждого пользователя $u \in U$ определено множество разрешенных ролей $AR(u) \subset R$ и роль по умолчанию $DR(u) \in R$. При этом всегда выполнено $DR(u) \in AR(u)$.

Множество типов обозначим через T, причем $T = FDTypes \sqcup DevTypes \sqcup ProcessTypes \sqcup IPCTypes \sqcup NetTypes \sqcup SCDTypes \sqcup UserTypes$. Здесь множества, входящие в дизъюнктное объединение, соответствуют множествам типов обычных файлов/каталогов, файлов устройств, процессов, объектов межпроцессного взаимодействия, сетевых объектов, системных объектов (таких как, например, таймер) и объектов системы управления пользователями в RSBAC. Здесь прослеживается аналогия с системой классов объектов доступа в SELinux. Вместе с тем, следует отметить, что разделение по классам в SELinux более детально.

Для каждого множества типов определен фиксированный набор прав доступа, обозначим их соответственно: $FDPermissions$, $DevPermissions$, $ProcessPremissions$, $IPCPermissions$, $NetPermissions$, $SCDPermissions$, $UserPermissions$. Как и в случае с SELinux, удобно ввести множество P общих прав доступа, как объединение указанных выше множеств.

В качестве примера приведем множество $FDPermissions$ согласно документации по системе RSBAC: $FDPermissions$ = {append_open, change_owner, chdir, close, create, delete, execute, get_perm_data, get_stat_data, link_hard, modify_access_data, modify_attribute, mount, read, read_attribute, read_write_open, read_open, rename, search, truncate, umount, write, write_open, map_exec}. Если сравнить содержание этой записи с типами доступа для классов `file` и `dir` в SELinux, то легко убедиться, что отличия довольно незначительны.

Для каждой роли $r \in R$ политика в RSBAC определяет множество совместимых ролей $CR(r) \subset R$, это является аналогом разрешенных переходов от одной роли к другой в SELinux.

Для каждой роли $r \in R$ в RSBAC определяется множество совместимых типов $CT(r) \subset T \times P$, это можно трактовать как аналог разрешенных доступов в SELinux.

Пусть S — множество субъектов (процессов) в системе, O — множество объектов. Для определения доступа процесса к процессу будем считать, что $S \subset O$. Для каждого $o \in O$ определен владелец $user(o) \in U$ и тип объекта $type(o) \in T$, причем $type(o)$ принадлежит соответствующему классу типов, например, $type(s) \in ProcessTypes$. Для каждого субъекта $s \in S$ определена его роль $role(s) \in R$. Для каждого объекта $o \in O$ определена назначаемая роль $FR(o) \in$

$R \sqcup \{role\_inherit\_user, role\_inherit\_process\}$. Здесь удобно считать, что для всех $o \in O \setminus \{o : type(o) \in FDTypes\}$ верно $FR(o) = role\_inherit\_process$, так как для таких объектов понятие назначаемой при выполнении роли не определено, поскольку такие объекты не могут быть исполняемыми. Назначаемая роль при выполнении файла представляет собой аналог точки входа (entrypoint) в SELinux.

Для каждого субъекта всегда должен выполняться предикат корректности, а именно $valid(s) = (role(s) \in AR(user(s)))$.

Предикат предоставления доступа P субъекта $s \in S$ к объекту $o \in O$ можно записать в виде

$$\begin{aligned}\Delta(s,o,p) = (type(o),p) \in CT(role(s)) \wedge (p = execute \implies \\ \implies (FR(o) \in CR(role(s)) \vee FR(o) = role\_inherit\_process \vee \\ \vee (FR(o) = role\_inherit\_user \wedge DR(user(s)) \in CR(role(s)))).\end{aligned}$$

Здесь заключение импликации — это предикат, обозначающий возможность перехода к новой роли в точке входа.

Аналогом правил `type_transition` из описания политики SELinux в модели RC являются типы создаваемых объектов, определяемые текущей ролью $r \in R$: $DefFDType(r) \in FDTypes$, $DefProcessType(r) \in ProcessTypes$, $DefIPCType(r) \in IPCTypes$ и также для остальных классов типов.

На самом деле изложенное выше описание теоретико-множественной модели RC не совсем соответствует реализации в системе RSBAC, в которой присутствует также изменение роли при смене пользователя и разделение прав администрирования системы, не имеющие аналогов в SELinux. Обработка смены пользователя при системном вызове типа setuid() в SELinux не оправдана, поскольку пользователи, определенные в политике SELinux, независимы от обычных пользователей в Linux. Отсутствие же раздельного администрирования политики, напротив, может оказаться существенным недостатком SELinux.

2.3 grsecurity/Role-Based Access Control

Система grsecurity [4] — новая активно развивающаяся программная компонента безопасности для ядра Linux, поддерживающая ролевую политику с возможностью обучения.

Модель управления доступом для ролевой политики безопасности, используемая в grsecurity, — пожалуй, самая примитивная, с точки зрения предоставляемых ею механизмов, из рассматриваемых в настоящей работе. В ней отсутствуют точки входа в роль, как в реализациях моделей TE и RC. Для изменения роли необходимо использование специального средства администрирования. Нет разделения объектов по типам, соответственно, не предусмотрено ограничения доступа, например, к IPC-объектам или сетевым объектам — рассматривается только доступ к объектам файловой системы. Приведем описание конфигурации политики, используемой системой grsecurity.

Описание политики состоит из последовательного определения ролей в системе, среди которых обязательно присутствует роль `default`. Каждое такое определение выглядит следующим образом:

```
role <имя роли> <флаги роли>
[role_transitions <список ролей>]
<список определений субъектов>
```

Здесь флаги роли представляют собой набор внутренних параметров роли в grsecurity, не имеющих прямого отношения к доступу субъектов к объектам. Например, флаг `'A'` обозначает администраторскую роль.

Строка конфигурации `role_transitions` позволяет определить роли, в которые можно переходить из текущей роли.

Далее, права каждого субъекта (процесса) определяются объектом — исполняемым файлом, из которого процесс запущен. Кроме того, существуют так называемые вложенные субъекты, позволяющие внести зависимость от порядка запуска процессов-субъектов друг из друга.

Определение субъекта в конфигурационном файле выглядит следующим образом:

```
subject { <объект> или <путь вложенного субъекта> } <флаги субъекта>
<список определений объектов>
<список разрешенных posix capabilities>
```

Здесь путь вложенного субъекта означает строку вида

```
<путь к объекту>:<путь к объекту>:...:<путь к объекту>
```

Заметим, что вместо определения доступов для каждого исполняемого файла в системе можно в качестве объекта указать каталог и использовать флаг `'i'`, означающий наследование всеми исполняемыми файлами в этом каталоге и его подкаталогах указанных прав, если в других описаниях субъектов не определено иначе. Другие флаги субъекта отвечают за управление системой PaX предотвращения выполнения произвольного кода, а также управляют разрешением или запрещением трассировки процесса-субъекта. Таким образом, как и ролевые флаги, флаги субъекта не имеют прямого отношения к доступу к объектам.

Каждое определение объекта описывается как

```
<путь к объекту> <флаги объекта>
```

Флаги объекта обозначают разрешенные права доступа к данному объекту и включают в себя:

`'r'` — право на чтение;

`'w'` — право на запись и дополнение;

`'a'` — право на дополнение;

`'c'` — право на создание файлов в каталоге;

`'d'` — право на удаление файлов в каталоге;

`'x'` — право на исполнение/поиск в каталоге;

`'l'` — право на создание ссылок;

`'m'` — право на создание setuid/setgid файлов;

`'i'` — флаг обозначающий наследование, так же как и во флагах субъекта.

Можно заметить, что по сравнению с наборами типов доступа в SELinux/TE и в RSBAC/RC перечисленных выше прав существенно меньше. Это свидетельствует о том, что с большой вероятностью при необходимости высокого уровня детализации типов доступа подходящую политику построить не удастся.

По умолчанию процессам пользователя root выставляется первая администраторская роль. Обычным пользователям при этом выставляется подходящая пользовательская роль, в случае ее отсутствия — подходящая групповая роль, а если и ее нет, то — роль `default`.

С учетом приведенного анализа функциональности системы и сделанных замечаний, опишем предложенную автором теоретико-множественную модель политики grsecurity/RBAC.

Пусть O — конечное непустое множество объектов. Содержательно, элементами этого множества являются объекты файловой системы. В множестве O выделено непустое подмножество исполняемых объектов $E \subset O$.

Определим множество ролей R и множество доступов A, которые также предполагаются конечными и непустыми. $A = \{r, w, a, c, d, x, l, m\}$ — по аналогии с флагами объектов.

Для множества ролей определен предикат $\rho(r_1, r_2)$ возможности перехода от одной роли к другой. Обозначим $\hat{\rho}(r_1, r_2)$ транзитивное замыкание $\rho(r_1, r_2)$ как отношения на $R \times R$.

Множество субъектов определим как конечное множество пар вида (роль, конечное слово над алфавитом E): $S = R \times E^*$. Пустое слово ε в этом случае соответствует «пустому» субъекту, действующему непосредственно после перехода в новую роль.

Также будем считать, что определено конечное непустое множество пользователей U, совпадающее с множеством пользователей системы. Обозначим $DR(u)$ — роль по умолчанию для пользователя.

В описании политики фактически задается предикат $\Delta(s, o, a)$ предоставления доступа $a \in A$ от субъекта $s = (r, e) \in S$ в роли $r \in R$ к некоторому объекту $o \in O$. Этот предикат с множества $S \times O \times A$ легко продолжается на $(R \times E^*) \times O \times A$, если принять $\Delta(s, o, a)$ ложным для $s \in (R \times E^*) \setminus S$.

Предикат $\Delta(s, o, a)$ определен для всех субъектов, однако не каждый субъект может быть реализован в некоторой фиксированной роли $r \in R$. Реализация субъекта означает последовательное выполнение всех составляющих субъекта. Возможность реализации определяется формулой $(s = \varepsilon) \vee (s = e_1 e_2 \dots e_n \in E^* \implies (\Delta((r, \varepsilon), e_1, x) \wedge (\forall i = 1, \dots, n-1 \implies \Delta((r, e_1 \dots e_i), e_{i+1}, x))))$.

3 Сравнение моделей разграничения доступа

В данном разделе приведем результаты решения двух задач выражения моделей логического разграничения доступа (ЛРД) в терминах конфигураций рассматриваемых подсистем управления ЛРД в ОС Linux. Полученные результаты позволяют судить о таком свойстве моделей ЛРД, как выразительность.

Под выразительностью будем понимать возможность реализации заданной конфигурации одной из подсистем управления доступом в терминах другой аналогичной подсистемы так, что с точки зрения возможных доступов заданная реализация модели разграничения доступа и вновь созданная эквивалентны. Выразительность — относительное понятие, то есть, практически могут быть интересны утверждения о том, что одна модель ЛРД не менее выразительна, чем другая, разумеется, при некоторых ограничениях на модели. Результаты такого вида могут быть полезны при унификации и интеграции различных средств управления доступом. Например, сформулированные в данном разделе теоретические результаты, учитывая их конструктивность, могут быть применены в объединении нескольких моделей ЛРД, используя модель SELinux/Type Enforcement как доминантную.

Будем считать, что задана некоторая политика в исходной модели (RSBAC/RC или grsecurity/RBAC), выполнены некоторые ограничения и необходимо построить политику в модели SELinux/TE, предоставление доступа в рамках которой совпадает с предоставлением доступа в соответствии с заданной исходной политикой.

3.1 Выражение политики RSBAC/RC через политику SELinux/TE

Приведем формулировку и конструктивное решение задачи о выражении политики Role Compatibility (RC) из системы RSBAC в терминах политики Type Enforcement (TE) из SELinux.

Допущения, принимаемые в этой задаче, формулируются в виде следующих утверждений.

- Рассматриваются политики RC без прав администрирования. Такое ограничение оправдано, особенно, если принять, что политики RC и TE сравниваются с точки зрения возможности выражения фиксированной политики логического разграничения доступа.
- Есть взаимно-однозначное соответствие между типами доступа в политиках, имея в виду, что политики сравниваются на тех типах доступа, где такое соответствие существует.
- Множества пользователей, субъектов и объектов в сравниваемых политиках понимаются одинаково.
- В модели RC нет объектов с назначаемой ролью, наследуемой от пользователя, что действительно представляет собой ограничение, ослабляющее модель.

Далее будем считать, что задана некоторая политика в модели RC, выполнены указанные выше ограничения и необходимо построить политику TE, предоставление доступа в рамках которой совпадает с предоставлением доступа в соответствии с заданной политикой RC. Поскольку обозначения сущностей в моделях RC и TE могут совпадать, в дальнейшем изложении пометим их соответственно префиксами «$RC.$» и «$TE.$».

Приведем конструкцию базовых множеств и предикатов математической модели политики TE с использованием элементов теоретико-множественной модели заданной политики RC.

Положим:

$TE.U := RC.U$,
$TE.R := \{u_i\_role | u_i \in RC.U\} \sqcup \{r_o\}$,
$TE.D := RC.R \times RC.ProcessTypes$,
$TE.T := TE.D \sqcup (RC.T \setminus RC.ProcessTypes)$.

С учетом построенных множеств приведем конструктивные определения базовых предикатов в модели TE.

$(TE.\mu(u_i, r) := (r = u_i\_role \vee r = r_o),\ u_i \in TE.U,\ r \in TE.R$
$(TE.\rho(u_i\_role, t) := t \in RC.ProcessTypes \wedge (t, CREATE) \in RC.CT(u_i\_role)),\ u_i \in RC.U,\ t \in RC.T$

$TE.\alpha(t_1,t_2,c,p) := (t_1 = (r_1,\tau_1) \in TE.D \wedge (t_2 \notin TE.D \implies (class(t_2) = c \wedge (t_2,p) \in RC.CT(r_1))) \wedge (t_2 = (r_2,\tau_2) \in TE.D \implies ((c = process \wedge (\tau_2,p) \in RC.CT(r_2)) \vee (c = process \wedge p = transition \wedge r_2 \in RC.CR(r_1)))))$, $t_1,t_2 \in TE.T$, $c \in TE.C$, $p \in TE.P$,

где $class$: $RC.T \to TE.C$ сопоставляет типу объекта из RC соответствующий класс доступа из TE.

$TE.\alpha_\rho(r_1,r_2) := (r_1 = r_2)$,

$\forall c \in TE.C, p \in TE.P, u_i \in TE.U, r_i \in TE.R, t_i \in TE.T, i = 1,2 : (TE.\chi_{c,p}(u_1,r_1,t_1;u_2,r_2,t_2))$.

Для получения утвердительного ответа на вопрос задачи о выражении одной политики через другую, необходимо показать, принимая во внимание приведенный выше формализм и конструктивные определения базовых понятий политики TE, что предикаты предоставления доступа ($TE.\Delta$ из вновь построенной политики и $RC.\Delta$ из заданной политики) содержательно совпадают.

Легко видеть, что в определении предиката $TE.\Delta$ часть $TE.\mu(u_1,r_1) \wedge TE.\mu(u_2,r_2) \wedge ((c,p) = (process, transition) \implies TE.\alpha_\rho(r_1,r_2))$ обеспечивает связь пользователей и искусственно введенных ролей. Часть $TE.\rho(r_1,t_1) \wedge TE.\rho(r_2,t_2)$ выражает привязку искусственных ролей к доменам, которые соответствуют ролям из $RC.R$ и типам процессов из $RC.ProcessTypes$. Ограничений вида `constrain` в модели TE нет, поскольку предикат $TE.\chi$ всегда истинен. Часть $TE.\alpha(t_1,t_2,c,p)$ содержательно выражает возможность доступа и возможность перехода в другую роль в модели RC.

Таким образом, приведенная конструкция, соответствующая политике TE, дает в точности те же доступы, что и исходная политика RC.

3.2 Выражение политики grsecurity/RBAC через политику SELinux/TE

Постановка задачи, исследуемой в этом разделе, по сути, схожа с постановкой предыдущей задачи за тем исключением, что заданной является политика grsecurity/RBAC и ограничивающих допущений в задаче меньше.

Допущения, принимаемые в задаче, следующие.

- Есть взаимно-однозначное соответствие между типами доступа в политиках. Будем считать, что политики сравниваются на тех типах доступа, где такое соответствие существует.
- Множества пользователей, субъектов и объектов в сравниваемых политиках понимаются одинаково.

Задача состоит в моделировании политикой SELinux/TE некоторой заданной политики grsecurity/RBAC. Обозначения сущностей в моделях grsecurity/RBAC и TE в дальнейшем изложении пометим соответственно префиксами «$GR.$» и «$TE.$».

Доказательство возможности выражения одной политики в рамках другой проведем аналогично решению задачи предыдущего раздела. То есть, конструктивно определим вновь создаваемую политику SELinux/TE и покажем содержательную эквивалентность предикатов предоставления доступа в сравниваемых моделях разграничения доступа.

Положим:

$TE.U := GR.U$,

$TE.R := GR.R \sqcup \{r_o\}$,

$TE.D := GR.S$,

$TE.T := TE.D \sqcup GR.O$.

Далее приведем построение базовых предикатов математической модели политики SELinux/TE.

$TE.\mu(u,r) := (GR.\hat{\rho}(GR.DR(u),r) \vee r = r_o)$,

$TE.\rho(r,t) := (t = (r,e) \in GR.S \vee r = r_o)$,

$TE.\alpha(t_1,t_2,c,p) := (t_1 = (r_1,e_1) \in TE.D \wedge ((t_2 \in TE.T \setminus TE.D \wedge GR.\Delta(t_1,t_2,access(c,p))) \vee (t_2 = (r_2,e_2) \in TE.D \wedge c = process \wedge p = transition \wedge (\exists e \in GR.E : e_2 = e_1 \cdot e))))$,

где $access(c,p) : TE.C \times TE.P \to GR.A$ — сопоставление типов доступа в TE и grsecurity/RBAC, а «$\cdot$» — операция конкатенации слов.

$TE.\alpha_\rho(r_1,r_2) := (r_1 = r_2)$,

$\forall c \in TE.C, p \in TE.P, u_i \in TE.U, r_i \in TE.R, t_i \in TE.T, i = 1,2 : (TE.\chi_{c,p}(u_1,r_1,t_1;u_2,r_2,t_2))$.

Покажем совпадение предикатов предоставления доступа. В рамках представленной конструкции можно констатировать следующее:

- в предикате $TE.\Delta$ часть $TE.\mu(u_1, r_1) \wedge TE.\mu(u_2, r_2) \wedge ((c, p) = (process, transition) \implies TE.\alpha_\rho(r_1, r_2))$ обеспечивает связь пользователей и ролей, аналогичную такой же связи в исходной политике;
- часть $TE.\rho(r_1, t_1) \wedge TE.\rho(r_2, t_2)$ выражает привязку ролей к субъектам исходной политики;
- ограничений типа `constrain` нет, поскольку предикат $TE.\chi$ всегда истинен;
- часть $TE.\alpha(t_1, t_2, c, p)$ непосредственно выражает возможность доступа в grsecurity/RBAC и возможность перехода в другой домен в TE для корректной обработки вложенных субъектов в grsecurity/RBAC.

С учетом изложенного, построенная политика SELinux/TE имитирует исходную политику.

4 Спецификация и проверка некоторых свойств информационных потоков

Для анализа некоторых свойств политики TE далее предлагается оригинальная модель, основанная на машине состояний и линейной темпоральной логике как языке спецификации. Похожие исследования проведены, например, в [8]. Представленная в данном разделе модель направлена также на исследование информационных потоков, однако при этом она шире модели, описанной в [8].

4.1 Описание модели

Упростим политику TE следующим образом: вместо множества $\Gamma \subset C \times P$ введем упрощенное множество доступов $A = \{read, write, transition\} \sqcup \{read\_by, written\_by\}$, где read_by, written_by — вспомогательные «обратные» доступы от типов к доменам, которые разрешены тогда и только тогда, когда разрешены соответствующие прямые доступы от доменов к типам.

Множество состояний определим как $S = U \times R \times T \times U \times R \times T \times A \times \{true, false\}$, последняя координата этого декартова произведения будет обозначать, что все состояния в некотором пути из переходов удовлетворяют предикату предоставления доступа, а само состояние означает переход от одного контекста безопасности к другому.

Определим следующим образом отношение перехода $\tau \subset S \times S$:

$$\begin{aligned}&((u_1, r_1, t_1, u_2, r_2, t_2, a, f), (u_1', r_1', t_1', u_2', r_2', t_2', a', f')) \in \tau \iff \\ &\iff (\Delta_a(u_1, r_1, t_1, u_2, r_2, t_2) \wedge \Delta_{a'}(u_1', r_1', t_1', u_2', r_2', t_2') \wedge \\ &\wedge u_1' = u_2 \wedge r_1' = r_2 \wedge t_1' = t_2 \wedge f' = f) \vee f' = false.\end{aligned}$$

Множество начальных состояний $\iota \subset S$ определим как

$$\iota = \{(u_1, r_1, t_1, u_2, r_2, t_2, a, f) \in S : \Delta_a(u_1, r_1, t_1, u_2, r_2, t_2 \wedge f = true\}.$$

Формулы линейной темпоральной логики включают в себя:

- все пропозициональные формулы от переменных состояния и LTL-формул;
- формулы вида «`X` f», где f — LTL-формула, `X` — оператор LTL, такая конструкция означает, что во всех непосредственно следующих за текущим состояниях верна формула f;
- формулы вида «`G` f», где f — LTL-формула, `G` — оператор LTL, означающий, что во всех состояниях на любом пути из текущего состояния верна формула f;
- формулы вида «`F` f», где f — LTL-формула, `F` — оператор LTL, означающий, что существует состояние на любом пути из текущего состояния, в котором верна формула f;
- формулы вида «f `U` g», где f,g — LTL-формулы, `U` — оператор LTL, означающий, что на любом пути из текущего состояния формула f верна во всех состояниях до некоторого, в котором верна формула g, причем такое состояние существует;

- формулы вида «f V g», где f,g — LTL-формулы, V — оператор LTL, означающий, что на любом пути из текущего состояния формула g верна во всех состояниях включительно до некоторого, в котором верна формула f, если такое состояние существует.

Представим следующие примеры выражения с помощью LTL свойств упрощенной модели логического разграничения доступа, основанной на SELinux/TE.

- Возможность потока от данных одного типа (src_t) к данным другого типа dst_t:

$$(t_1 = \texttt{src\_t}) \wedge ((a = read\_by \vee a = write)\ \texttt{U}\ (t_2 = \texttt{dst\_t} \wedge a = write \wedge f)).$$

- Доступ к домену domain2 возможен только из домена domain1, причем только с помощью перехода через домен domain3:

$$\neg(t_1 = \texttt{domain1} \wedge (a = transition\ \texttt{U}\ (a = transition \wedge f \wedge t_2 = \\ = \texttt{domain2})) \wedge \neg(t_1 = \texttt{domain1} \wedge a = transition \wedge t_2 = \texttt{domain3} \wedge \\ \wedge\ \texttt{X}\ (t_1 = \texttt{domain3} \wedge a = transition \wedge t_2 = \texttt{domain2} \wedge f))).$$

- Любой пользователь, действующий в роли user_r не получит доступа на запись к данным типа system_data_t через смены доменов:

$$(r_1 = \texttt{user\_r}) \wedge ((a = transition)\ \texttt{U}\ (t_2 = \texttt{system\_data\_t} \wedge a = write \wedge f)).$$

4.2 Пример проверки спецификации

Рассмотрим простейший пример применения средства анализа моделей NuSMV ([6], [7]) для решения задачи анализа политики модели SELinux/TE на соответствие заданной спецификации.

Описание модели приведем в упрощенном синтаксисе конфигурации SELinux/TE:

```
user system_u
role system_r
user_roles system_u system_r
role_types system_r server1_process_t server2_process_t
type server1_process_t
type server2_process_t
type server1_data_t
type server2_data_t
type shared_data_t
allow server1_process_t server1_data_t read
allow server1_process_t server1_data_t write
allow server1_process_t shared_data_t read
allow server2_process_t server2_data_t read
allow server2_process_t server2_data_t write
allow server2_process_t shared_data_t read
```

Содержательно, в приведенном примере описывается система, в которой функционируют процессы двух типов server1_process_t и server2_process_t, каждый из которых имеет доступ на чтение и на запись к своим данным, имеющим, соответственно типы server1_data_t и server2_data_t. Также в системе присутствуют данные типа shared_data_t, которые могут читать процессы обоих типов.

Спецификация невозможности потока из данных типа server1_data_t в данные типа server2_data_t выглядит следующим образом:

$$\neg((t_1 = \texttt{server1\_data\_t}) \wedge ((a = read\_by \vee a = \\ = write)\ \texttt{U}\ (t_2 = \texttt{server2\_data\_t} \wedge a = write \wedge f))).$$

Из примера очевидно, что эта спецификация верна.

Описание и спецификация преобразуются в формат NuSMV:

```
MODULE main

VAR
_u1 : { system_u };
_r1 : { _object_role, system_r };
_t1 : { server2_data_t, server1_data_t, server2_process_t,
server1_process_t, shared_data_t };
_u2 : { system_u };
_r2 : { _object_role, system_r };
_t2 : { server2_data_t, server1_data_t, server2_process_t,
server1_process_t, shared_data_t };
_a  : { _read, _write, _read_by, _written_by, _transition };
_flow : boolean;

DEFINE
_user_role_1 := case
_u1=system_u : _r1 in {system_r};
TRUE : FALSE;
esac;

_user_role_2 := case
_u2=system_u : _r2 in {system_r};
TRUE : FALSE;
esac;

_role_type_1 := case
_r1=system_r : _t1 in {server1_process_t, server2_process_t};
TRUE : FALSE;
esac;

_role_type_2 := case
_r2=system_r : _t2 in {server1_process_t, server2_process_t};
TRUE : FALSE;
esac;

_role_transition := case
_r1=system_r : FALSE;
TRUE : FALSE;
esac;

_allow := case
_t1=shared_data_t & _t2=server2_process_t & _a=_read_by : TRUE;
_t1=server1_process_t & _t2=shared_data_t & _a=_read : TRUE;
_t1=server2_process_t & _t2=shared_data_t & _a=_read : TRUE;
_t1=server1_data_t & _t2=server1_process_t & _a=_read_by : TRUE;
_t1=server1_data_t & _t2=server1_process_t & _a=_written_by : TRUE;
_t1=server2_data_t & _t2=server2_process_t & _a=_written_by : TRUE;
_t1=server2_data_t & _t2=server2_process_t & _a=_read_by : TRUE;
_t1=server1_process_t & _t2=server1_data_t & _a=_write : TRUE;
_t1=server1_process_t & _t2=server1_data_t & _a=_read : TRUE;
_t1=shared_data_t & _t2=server1_process_t & _a=_read_by : TRUE;
_t1=server2_process_t & _t2=server2_data_t & _a=_read : TRUE;
_t1=server2_process_t & _t2=server2_data_t & _a=_write : TRUE;
TRUE : FALSE;
esac;
```

```
_constrain := case
TRUE : TRUE;
esac;

_trans :=
(_role_type_1 | _r1=_object_role) &
(_role_type_2 | _r2=_object_role) &
(_user_role_1 | _r1=_object_role) &
(_user_role_2 | _r2=_object_role) &
_allow & _constrain &
(_a=_transition -> _role_transition);

INIT
_trans & _flow;

TRANS
(_trans & next(_trans) &
next(_u1) = _u2 & next(_r1) = _r2 & next(_t1) = _t2 &
next(_flow) = _flow) | (next(_flow) = FALSE);

LTLSPEC
! ((_t1=server1_data_t) & ((_a=_read_by | _a=_write)
U (_t2=server2_data_t & _a=_write & _flow)) );
```

Не вдаваясь в подробности синтаксиса NuSMV, в приведенном тексте в разделе `VAR` определяется множество состояний, в разделах `INIT` и `TRANS` — соответственно формулы, задающие множество начальных состояний и отношение перехода. В разделе `LTLSPEC` определяется анализируемая на соответствие спецификация.

В рассматриваемом примере выполнение проверки с помощью NuSMV дает следующий результат, подтверждающий выполнение спецификации:

```
-- specification !(_t1 = server1_data_t & ((_a = _read_by |
_a = _write) U ((_t2 = server2_data_t & _a = _write) &
_flow))) is true
```

Добавим в описание модели TE следующую строку:

```
allow server1_process_t shared_data_t write
```

В такой модели спецификация перестает быть верной, и с помощью NuSMV автоматически строится контрпример к спецификации, включающий путь из переходов, в котором не выполнена формула, задающая спецификацию:

```
-- specification !(_t1 = server1_data_t & ((_a = _read_by |
_a = _write) U ((_t2 = server2_data_t & _a = _write) &
_flow))) is false
-- as demonstrated by the following execution sequence
Trace Description: LTL Counterexample
Trace Type: Counterexample
-- Loop starts here
-> State: 1.1 <-
  _r1 = _object_role
  _t1 = server1_data_t
  _r2 = system_r
  _t2 = server1_process_t
  _a = _read_by
  _flow = 1
```

```
  _u1 = system_u
  _u2 = system_u
-> State: 1.2 <-
  _r1 = system_r
  _t1 = server1_process_t
  _r2 = _object_role
  _t2 = shared_data_t
  _a = _write
  _flow = 1
  _u1 = system_u
  _u2 = system_u
-> State: 1.3 <-
  _r1 = _object_role
  _t1 = shared_data_t
  _r2 = system_r
  _t2 = server2_process_t
  _a = _read_by
  _flow = 1
  _u1 = system_u
  _u2 = system_u
-> State: 1.4 <-
  _r1 = system_r
  _t1 = server2_process_t
  _r2 = _object_role
  _t2 = server2_data_t
  _a = _write
  _flow = 1
  _u1 = system_u
  _u2 = system_u
```

В приведенной трассе выполнения действий прослеживается путь, по которому возможен поток от типа `server1_data_t` к типу `server2_data_t` через тип `shared_data_t`. Рассмотренный пример, конечно, тривиален, но тем не менее показывает, как в автоматизированном режиме можно проверять некоторые свойства политики безопасности с помощью стандартных методик проверки моделей.

5 Заключение

Представленные в настоящей работе результаты по теоретико-множественным моделям описания политик безопасности различных реализаций систем управления доступом и имитации одной политики средствами другой позволяют аргументированно сравнивать выразительность используемых средств, выбирать наиболее подходящие из них для текущих целей.

Методика верификации политик с помощью представления функционирования системы в виде машины состояний и применения автоматических средств проверки моделей предоставляет возможность оценить свойства политики, и, при необходимости, найти слабое место в системе. Такая оценка может быть востребована, например, в процессе создания дистрибутивов защищенных операционных систем, где важны изоляционные возможности ролевых политик управления доступом, а из-за большого объема политик «ручная» верификация трудноосуществима.

Литература

[1] В. А. Васенин. Проблемы математического, алгоритмического и программного обеспечения компьютерной безопасности в Интернет // Материалы конференции МаБИТ-03, М.: МЦНМО, 2004, с. 111–143.

[2] National Security Agency. Security-enhanced Linux // `http://www.nsa.gov/selinux`.

[3] Rule-Set Based Access Control // `http://www.rsbac.org`.

[4] grsecurity. An Innovative Approach to Security // `http://www.grsecurity.net`.

[5] Amon Ott. The Role Compatibility Security Model // Nordic Workshop on Secure IT Systems (NordSec), 2002.

[6] NuSMV2 — A New Symbolic Model Checker // `http://nusmv.irst.itc.it`.

[7] A. Cimatti, E. Clarke, E. Giunchiglia, F. Giunchiglia, M. Pistore, M. Roveri, R. Sebastiani, A. Tacchella. NuSMV Version 2: An OpenSource Tool for Symbolic Model Checking // Proc. International Conference on Computer-Aided Verification (CAV 2002), July 2002, LNCS, vol. 2404.

[8] Joshua D. Guttman, Amy L. Herzog, John D. Ramsdell, Clement W. Skorupka. Verifying information flow goals in Security-Enhanced Linux // Journal of Computer Security, vol. 13, num. 1, 2005, p. 115–134.

Архитектура и модели для верификации политик безопасности

И. В. Котенко, А. В. Тишков, О. В. Черватюк

1 Введение

Разработка систем управления безопасностью вычислительных сетей, основанных на политиках, являются одним из наиболее актуальных направлений исследований в области защиты информации. В настоящее время общепринятым стандартом архитектуры управляющей системы, основанной на политиках, является рекомендация IETF [1]. Такая архитектура содержит централизованный репозиторий правил политики безопасности для всей системы, что делает политику доступной для анализа и верификации. В настоящей работе представлено развитие архитектуры и моделей функционирования системы верификации защиты вычислительной сети (СВЕРЗ), первоначально предложенной в [2] и реализуемой в соответствии с рекомендациями IETF.

В работе предлагается уточненная архитектура системы верификации политики, описываются механизмы работы с политиками трех уровней: верхнего уровня, приближенного к языку требований пользователя, среднего уровня, классифицирующего правила по нескольким категориям, и нижнего уровня, описывающего политику в формате Common Information Model (CIM). Рассматривается подход к реализации ядра СВЕРЗ, демонстрируется пример моделирования и обнаружения конфликтов в политике авторизации, а также делается обзор релевантных исследований.

2 Уточненная архитектура системы верификации

В системе СВЕРЗ язык описания политик имеет три уровня: верхний, средний и нижний (рис. 1).

Язык верхнего уровня (ЯВУ) (Upper-level Language (UL)) описывает задачу с обобщенной точки зрения. Формулировки допускают упоминание групп устройств и типов приложений («подсеть S не должна быть доступна с хоста H по протоколу P»). Для определения политик верхнего уровня используется язык скриптов и набор трансляторов с верхнего на средний уровень (ВС-трансляторы).

Правила верхнего уровня транслируются на язык *среднего уровня (ЯСУ)* (Intermediate level Language (IL)) в одну из шести категорий правил политики: аутентификация, авторизация, фильтрация, конфиденциальность, операционные правила и правила обнаружения уязвимостей. Для каждой из указанных категорий разработан ВС-транслятор, входом для которого является правило верхнего уровня, а выходом — XML-документы, валидные относительно XML-схемы соответствующей категории.

Одним из нетривиальных трансляторов с верхнего на средний уровень является ВС-транслятор, определяющий политику фильтрации. В контексте его задачи узлы компьютерной сети разделяются на два типа: осуществляющие фильтрацию и нефильтрующие (рис. 2).

При задании политики, защищающей нефильтрующий узел, на графе, представляющем топологию сети, решается задача использования фильтрующих узлов для предоставления или запрета доступа к защищаемому узлу, как задача о минимальном разрезе графа. На рис. 2 приведен пример создания ВС-транслятором четырех правил фильтрации, когда политика верхнего уровня требует запрета доступа между нефильтрующими узлами.

При расширении языка верхнего уровня новыми конструкциями, в систему должен быть подгружен набор новых ВС-трансляторов для каждой из категорий, вовлеченных в расширение. Допускаются только расширения, в результате которых не изменяется существующий подъязык. Таким способом

Работа выполнена при поддержке «Фонда содействия отечественной науке», РФФИ (проект № 04-01-00167), программы фундаментальных исследований ОИТВС РАН (контракт № 3.2/03) и программы FP6 Европейского Сообщества, как часть проекта POSITIF (контракт № IST-2002-002314).

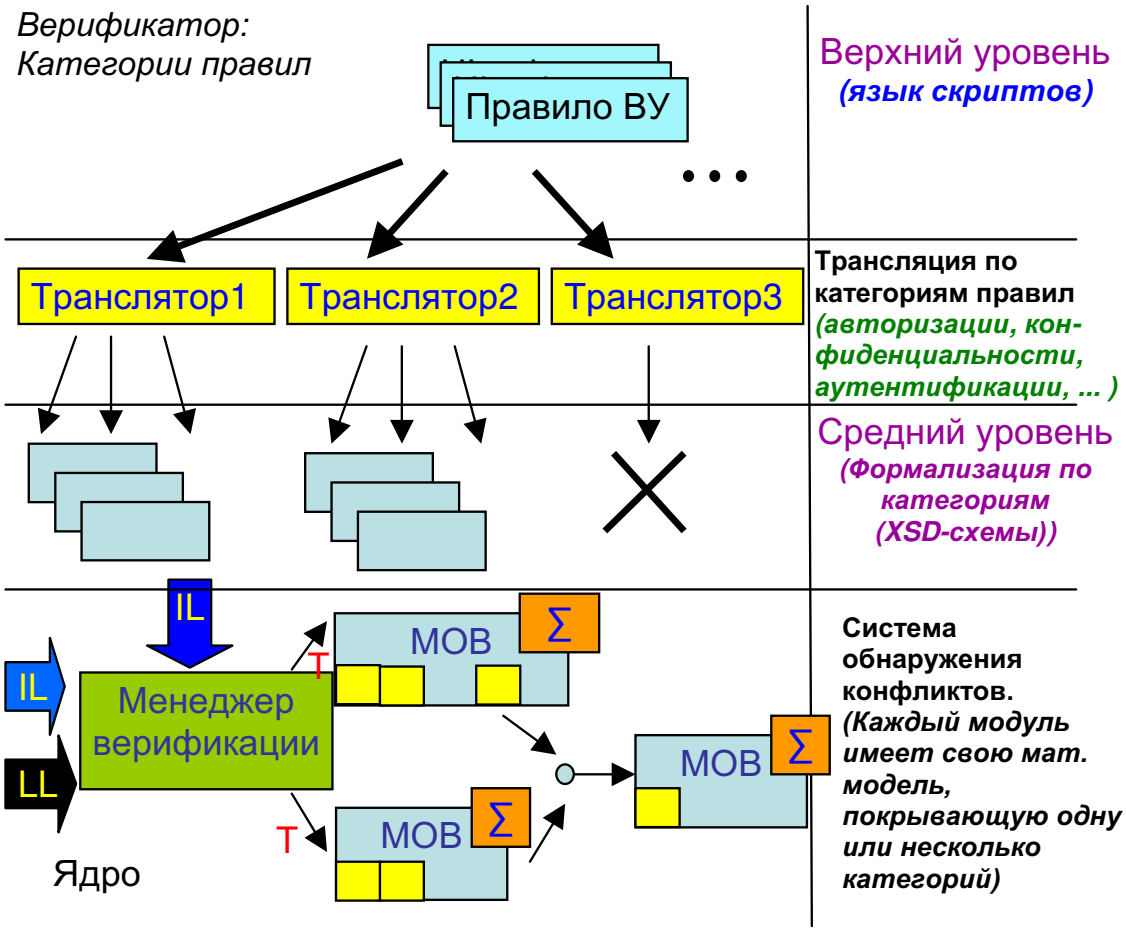


Рис. 1: Архитектура СВЕРЗ

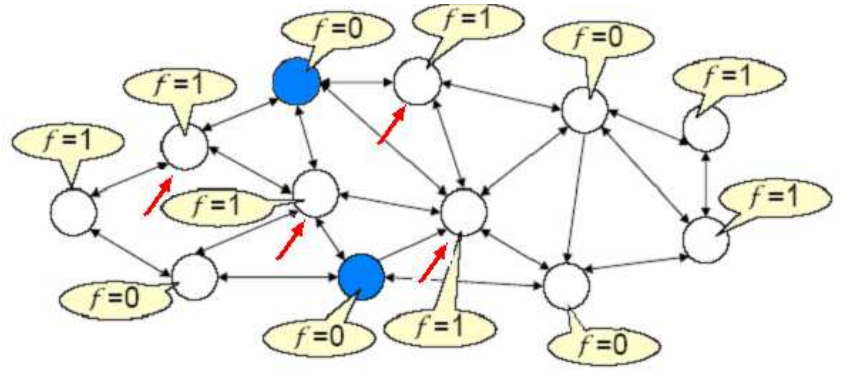


Рис. 2: Фильтрующие ($f = 1$) и нефильтрующие ($f = 0$) узлы

архитектура СВЕРЗ реализуется как открытая для интерпретации правил других языков, таких как Ponder [3] и других языков, определяемых пользователем.

Наконец, *язык нижнего уровня (ЯНУ)* (Low-level Language (LL)) представляет собой трансляцию правил среднего уровня в объектно-ориентированный формат Common Information Model (CIM).

Структура ядра верификатора содержит два типа базовых элементов: менеджер верификации и модуль верификации (МОВ). Каждый модуль имеет свою базу знаний (в виде аксиоматик, формул темпоральной логики, полурешеток действий и др.) и реализует собственный алгоритм проверки непротиворечивости политик и применимости к заданному описанию системы. Кроме того, каждый модуль объявляет о том, с какими категориями безопасности он работает. Менеджер верификации, получая на вход политики среднего и нижнего уровней, параллельно или последовательно вызывает модули верификации. Параллельная верификация возможна только для модулей, которые не изменяют набор правил. Модули, удаляющие, изменяющие или добавляющие правила, запускаются последовательно, принимая на вход политику, возможно измененную предыдущими модулями. Такой алгоритм работы ядра подразумевает итеративный вызов последовательности модулей. Итерации продолжаются до тех пор, пока набор правил не перестанет изменяться или до выполнения стоп-условия, в простейшем случае — явное ограничение числа итераций.

3 Категории безопасности

Как упоминалось выше, язык среднего уровня содержит XML-схемы для правил шести категорий безопасности.

Правило аутентификации содержит перечень субъектов (пользователей или ролей), объектов (сервисов, определенных на языке описания системы [1]), действий, которые можно выполнять над этими сервисами, метод аутентификации и уровень безопасности, к которому относится правило. Метод аутентификации соответствует классам-потомкам AuthenticationCondition, которые определены в CIM. Среди них: SharedSecretAuthentication, AccountAuthentication, BiometricAuthentication, NetworkingIDAuthentication, PublicPrivateKeyAuthentication, KerberosAuthentication, DocumentAuthentication, PhysicalCredentialAuthentication. Все правила сопровождаются меткой уровня безопасности, для того чтобы система могла переключиться с одного уровня на другой, например, при обнаружении атаки.

Правило авторизации формулируется как if-then правило. Условная часть содержит бескванторную предикатную формулу с использованием логических связок NOT, AND и OR, атомами которой являются определения субъекта, объекта и действия, а также условие на текущее состояние системы. В качестве условий на состояние системы используются состояние сервисов (запущен, не запущен, ожидание, занят), результаты выполнения политик аутентификации и авторизации (субъект аутентифицирован/авторизован для выполнения действия над объектом), а также состояния, определяемые пользователем. Основные используемые CIM-классы — Policy, AuthorizedSubject, AuthorizedObject, AuthorizedPriviledge, ComputerSystem, Role, Identity.

Правило фильтрации представляет собой общепринятую таблицу управления доступом, строка которой включает адрес и порт источника, адрес и порт приемника, разрешение/запрет и дополнительно уровень безопасности. Основные используемые CIM-классы — Policy, FilterList и FilterEntry.

Правила конфиденциальности в настоящий момент рассматриваются в узком смысле, позволяя задавать только механизмы защиты каналов передачи данных. Соответствующая схема позволяет задать SSL- или IPSec-протокол шифрования для канала. Основные используемые CIM-классы — Policy, IPSecRule, SSLRule.

Операционные правила определяются условием на состояние системы и действиями, совершаемыми над объектами системы. Схема содержит компоненты для определения на хостах сервисов, доступных другим узлам сети, и действий, которые возможно выполнить над сервисами. Используется CIM-класс Policy, в иерархию классов добавляются OperationalRule, StatusCondidtion и OperationalAction.

Правила обнаружения уязвимостей создаются на основе базы данных уязвимостей [4]. Правило содержит идентификационный номер уязвимости, ссылку на эксплоит, название и версию уязвимого продукта, информацию о программном обновлении, устраняющем данную уязвимость, и некоторую дополнительную информацию [5].

4 Реализация ядра

Базовыми классами ядра СВЕРЗ являются менеджер верификации и модуль верификации.

Менеджер верификации (VerificationManager) передает модулям верификации спецификацию системы на языке описания системы и фрагменты спецификации политик, в соответствии с категориями безопасности, за которые отвечает модуль верификации. Кроме этого, в предлагаемом представлении менеджер выдает информацию о результатах верификации, информацию о противоречиях, если они возникли, и достигнутый уровень защищенности. Данный класс реализует шаблон проектирования «одиночка» [6], поскольку менеджер верификации должен быть в системе в единственном экземпляре.

UML-представление для менеджера верификации изображено на рис. 3. Для каждого public-поля подразумевается наличие функций установки значения и получения значения (set и get).

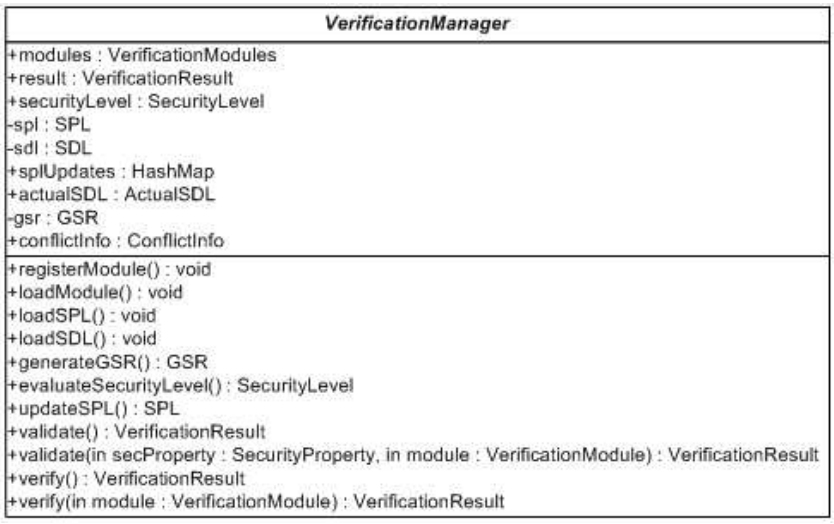


Рис. 3: Представление класса VerificationManager

В рамках данной работы поясним лишь некоторые основные поля и методы класса VerificationManager:

- Поле *HashMap splUpdates* содержит ссылки на объекты SPLUpdates, создаваемые в каждом модуле. Объекты SPLUpdates хранят перечень изменений, которые необходимо внести в набор правил политики для разрешения конфликтов, обнаруженных в процессе верификации.
- Поле *ActualSDL actualSDL* — пересмотренная топология сети, в которой некоторые сервисы блокированы политиками. ActualSDL содержит перечень блокированных сервисов.
- Поле *ConflictInfo conflictInfo* содержит информацию об обнаруженных в процессе валидации и верификации конфликтах.
- Метод *updateSPL*() служит для реализации изменений набора правил, предлагаемых модулями.
- Метод *validate*() без параметров производит проверку правил для каждой категории безопасности при помощи всех зарегистрированных и загруженных модулей, отвечающих за данную категорию безопасности.
- Метод *validate*() с параметрами выполняет обнаружение и разрешение конфликтов для правил в пределах одной категории безопасности. Категория безопасности и модуль, при помощи которого производится проверка, передаются через параметры метода.

- Метод *verify*() проверяет непротиворечивость всего набора правил и их применимость к заданному описанию системы при помощи определенного модуля.

Модуль верификации VerificationModule (рис. 4) осуществляет валидацию и верификацию правил категорий SecurityProperty, за которые он отвечает и которые перечислены в соответствующем поле.

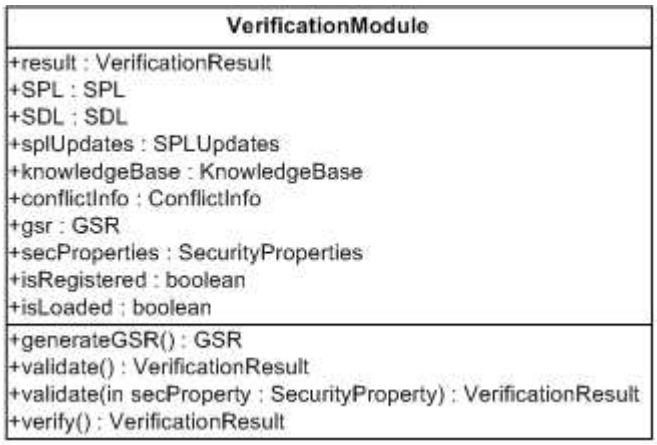


Рис. 4: Представление класса VerificationModule

Основными методами класса VerificationModule являются validate() и verify(). Через эти методы класс VerificationManager делегирует соответствующую функциональность классу VerificationModule.

5 Пример обнаружения конфликта

В настоящее время ведется работа над реализацией трех модулей верификации: (1) основанном на исчислении событий [7]; (2) базирующемся на технике проверки на модели (model checking) [8]; и (3) использующем построение полурешеток действий.

Опишем простой пример моделирования и обнаружения конфликта авторизации, реализованный на верификаторе моделей SPIN [9].

Конфликт авторизации возникает в том случае, когда один из пользователей приписывается к двум ролям R1 и R2, имеющим противоречивые привилегии на одно и то же действие: для одной роли разрешение, для другой — запрет.

Ключевыми блоками программы являются два процесса. Первый процесс случайным образом назначает и удаляет принадлежность пользователя к одной из двух ролей: R1 и R2. Представленный ниже код соответствует приписыванию пользователя к роли:

```
active proctype userRoleAssignment()
{
...
    :: (r.q<max_q_roles-1)->
        atomic {
            r.q++;
            if
            ::r.ar[r.q]=R1;
            ::r.ar[r.q]=R2;
            fi
            }
...
```

Второй процесс моделирует запросы на печать, посылаемые пользователем в случайные моменты времени. Процедура IsAssigned проверяет принадлежность пользователя к заданной роли.

Следующий код, при получении запроса на печать, присваивает значение true переменной deny (если пользователь в текущий момент принадлежит роли R1) или переменной allow (если пользователь принадлежит роли R2):

```
::printer_in?action,rr-> atomic
            {
            deny=false;
            allow=false
            IsAssigned(rr,R1,R1Res);
            IsAssigned (rr,R2,R2Res);
            if
            ::R1Res->deny=true
            ::else
            fi
            if
            ::R2->allow=true
            ::else
            fi
...
```

Возникновение конфликта заключается в нарушении следующего условия корректности состояния системы: allow и deny не могут выполняться одновременно:

```
assert((allow && !(deny)) || (!(allow) && deny))
```

6 Релевантные работы

Многие современные системы защиты вычислительных сетей, основанные на политиках, достаточно развиты, но охватывают не все категории безопасности, представленные в нашей работе, и имеют отличающиеся архитектуры.

Расширяемый язык разметки для управления доступом XACML [10] позволяет описывать политики авторизации. Трехуровневая структура описания политики (правило — политика, как множество правил — множество политик) позволяет построить гибкую систему контроля доступа с использованием формализованного понятия алгоритма разрешения конфликта на уровне политики и множества политик. В отличие от предлагаемого подхода, XACML не имеет отдельного языка для описания системы, описание узлов сети является частью описания правил.

Язык Ponder [3] содержит правила положительной и отрицательной авторизации, правила обязательного исполнения и правила делегирования. Авторы системы предложили несколько интересных подходов к разрешению конфликтов [11, 12], которые, однако, достаточно специфичны для введенного формализма политик.

«Гибкая система авторизации» (Flexible Authorization Framework, FAF) [13, 14] относится к системам управления доступом. К преимуществам FAF следует отнести подробное рассмотрение иерархий объектов, субъектов и привилегий при вычислении доступа. Используемый формализм позволяет специфицировать положительную и отрицательную авторизацию, включает понятия распространения привилегий по иерархиям, алгоритмы и стратегии разрешения конфликтов авторизации.

Существуют и другие подходы, представляющие различные техники обнаружения и разрешения конфликтов в политиках безопасности. Отметим подход, основанный на деонтической логике [15], динамическое обнаружение конфликта с применением темпоральной логики [16, 17], а также одну из базовых работ по классификации конфликтов политик безопасности [18].

7 Заключение

В настоящей работе предложено развитие архитектуры СВЕРЗ. Определена трехуровневая структура языка описания политик, от приближенного к натуральному языку верхнего уровня до объектно-

ориентированного представления политики в CIM-формате.

Описаны категории безопасности, на которые разделяются правила политики. Приведено UML-представление основных классов ядра СВЕРЗ, продемонстрирована идея реализации обнаружения конфликта в политике авторизации.

Дальнейшая работа связана с совершенствованием методик и алгоритмов верификации политик безопасности и разработкой рабочего прототипа СВЕРЗ на базе технологии веб-сервисов.

Литература

[1] IETF Policy Framework (policy) Working Group. http://www.ietf.org/html.charters/policy-charter.html.

[2] Тишков А. В., Котенко И. В. Спецификация и верификация политик безопасности защищенной вычислительной сети: использование исчисления событий // Математика и безопасность информационных технологий. Материалы конференции в МГУ. М.: МЦНМО, 2005.

[3] Ponder: A Policy Language for Distributed Systems Management. Department of Computing, Imperial College. http://www-dse.doc.ic.ac.uk/Research/policies/ponder.shtml.

[4] OSVDB: The Open Source Vulnerability Database. http://www.osvdb.org.

[5] Rohse M. Vulnerability naming schemes and description languages: CVE, Bugtraq, AVDL and VulnXML // SANS GSEC PRACTICAL, 2003.

[6] Гранд М. Шаблоны проектирования в Java. М.: Новое знание, 2004.

[7] Kowalski R.A., Sergot M.J. A Logic-Based Calculus of Events // New Generation Computing, No 4, 1986.

[8] Кларк Э.М., Грамберг О., Пелед Д. Верификация моделей программ: Model Checking. М.: МЦНМО. 2002.

[9] Holzmann G.J. The Spin Model Checker // IEEE Trans. on Software Engineering, Vol.23, No.5, 1997.

[10] OASIS: eXtensible Access Control Markup Language (XACML). http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml.

[11] Lymberopoulos L., Lupu E., Sloman M. Ponder Policy Implementation and Validation in a CIM and Differentiated Services Framework // IFIP/IEEE Network Operations and Management Symposium (NOMS 2004), Seoul, Korea, 2004.

[12] Bandara A., Lupu E., Russo A. Using Event Calculus to Formalize Policy Specifications and Analysis // IEEE 4th International Workshop on Policies for Distributed Systems and Networks, 2003.

[13] Jajodia S., Samarati P., Sapino M.L., Subrahmanian V. S. Flexible support for multiple access control policies. ACM Trans. Database Systems, Vol. 26, No.2, 2001.

[14] Jajodia S., Samarati P., Subrahmanian V.S. A Logical Language for Expressing Authorizations // IEEE Symposium on Security and Privacy, 1997.

[15] Cholvy L., Cuppens F. Analysing consistency of security policies // Proceedings of IEEE Symposium on Security and Privacy, 1997.

[16] Dunlop N., Indulska J., Raymond K. Methods for Conflict Resolution in Policy-Based Management Systems // Proceedings of the Seventh IEEE International Enterprise Distributed Object Computing Conference (EDOC'03), 2003.

[17] Dunlop N., Indulska J., Raymond K. Dynamic Conflict Detection in Policy-Based Management Systems // Proceedings of the Sixth IEEE International Enterprise Distributed Object Computing Conference (EDOC'02), 2002.

[18] Lupu E., Sloman M. Conflict Analysis for Management Policies // Fifth IFIP/IEEE International Symposium on Integrated Network Management IM'97, San Diego, 1997.

Телематические средства информационной безопасности на базе сетевых процессоров, функционирующих в скрытном режиме фильтрации

В. С. Заборовский

1 Введение

Современные телематические сети объединяют между собой широкий класс технических систем, используемых для передачи пакетного трафика, обработки результатов измерений, анализа и распределения навигационной информации и данных телеметрии. Во всех этих приложениях обмен данными осуществляется путем посылки и приема сетевых пакетов. Пакет, это специальная логическая последовательно-рекурсивная структура, которая формируется в узлах сети для организации обмена данными. Последовательная часть этой структуры состоит из двух полей - заголовка и данных. Рекурсивность пакета связана с тем, что данные могут быть другим пакетом со своей определенной структурой. Правила взаимодействия телематических приложений определяются данными об адресах, где располагаются приемники и источники сообщений, а конкретный путь передачи пакетов определяется протоколами маршрутизации в узлах сети. В линии связи никакой обработки пакета не происходит. Если в процессе обработки пакета принято решение не отправлять пакет в сеть, то считается, что либо пакет достиг заданного сетевого узла, либо пакет будет утерян. Базовая функциональность маршрутизатора определяется двумя последовательными стадиями обработки пакетов после их приема из линии связи, а именно *«запомнить — отправить»*.

В работе рассматривается новый подход к выбору архитектуры телематических устройств управления с учетом аспектов информационной безопасности, когда расширение функциональных требований к различным стадиям обработки пакетов осуществляется путем распределения процедур их реализации между сетевыми процессорами (СП) и каналами связи. Особенностью предлагаемого подхода является то, что применение специального типа СП, работающих в «стелс» режиме, не нарушает существующие адресные связи между узлами сети и не требует изменения политики маршрутизации. Адресная инвариантность позволяет встраивать системы информационной безопасности (ИБ) в существующие сетевые инфраструктуры без замены ранее установленного оборудования, так как масштабируемая производительность применяемых СП не снижает пропускную способность используемых каналов связи.

2 Тенденции развития систем телематики

По мере роста скорости передачи информации по коммутационным линиям и расширения спектра протоколов возрастают требования к производительности процессоров, используемых в узлах сети для обработки пакетов. Архитектура и особенности функционирования таких устройств – сетевых процессоров стала предметом большого числа исследований и разработок [1, 2, 3].

Известные в настоящее время решения можно разделить на два типа. Во-первых, это решения направленные на повышение производительности работы устройств маршрутизации. Основными параметрами, управляющими их работой, являются адреса приемника пакетов, поэтому применяемые решения направлены на ускорение поиска данных в таблицах маршрутизации. Второй типа решений использует различные процедуры классификации пакетов, обеспечения качество обслуживания за счет приоритетного распределения полосы пропускания. Такое разделение процессов обработки позволяет повысить интегральную производительность сетевого узла, но при этом часть данных может быть утеряна или должна быть передана повторно. Повысить эффективность обработки и качество функ-

ционирования сети можно на основе использования новой телематической парадигмы *«обработать - запомнить - отправить».*

3 Системы информационной безопасности

На практике применение новой парадигмы сводится к использованию модели взаимодействия открытых систем (ВОС), в которой в отличии от классических решений на базе протокола TCP/IP, вводится дополнительный канальный уровней управления. В результате объектом управления являются специальные структуры данных, которые представлены в формате сетевых пакетов с набором определенной адресной информации. Рассмотрим процесс генерации пакетов в процессе передачи данных через телематическую сеть. В этой процедуре можно выделить следующие стадии:

1) выделение определенного объема данных, подлежащих передаче через сеть;

2) формирование структуры, в которой производится количественный учет всего объема передаваемых данных;

3) присоединение к данным специального заголовка, содержащего набор параметров, на основании которых производится обработка пакета в узлах сети;

4) формирование кадра, структура которого соответствует требованиям каналообразующей аппаратуры;

5) передача кадра по коммутационной линии, которая связывает два узла сети.

В процессе передачи пакетов различают несколько типов сетевых узлов , а именно: узлы генерации, узлы в которых производится обработка только заголовков пакетов; узлы, в которых производится обработка заголовков и данных. Процесс маршрутизации или выбор интерфейса, куда передается пакет после обработки, носит локальный характер, то есть осуществляется в каждом узле сети, через который проходит пакет. Для маршрутизации используется адрес узла получателя пакета, который указан в соответствующем поле заголовка и таблица связи между адресами узлов сети и номерами интерфейсов маршрутизатора.

Описанный выше процесс весьма уязвим для различного рода воздействий, которые могут нарушить стандартную процедуру передачу пакетов или произвести подмену пакетов на этапе их следования от узла генерации до узла приема.

Основными мерами защиты являются:

1) организация специальной траектории движения пакетов через такие узлы сети, в которых реализуются специальные правила обработки, позволяющие не допустить прохождение через них пакетов с определенными адресами и параметрами заголовков;

2) использование режима туннеля, когда защищаемый пакет передается в поле данных другого сетевого пакета;

3) использование специальных режимов передачи пакетов, когда параметры заголовков защищены криптографическими средствами.

Практическая реализация всех этих мер защиты осуществляется несколькими способами, которые можно разделить на методы фильтрации и криптографической обработки данных. Первые методы защиты адресного пространства сети реализуются с помощью специальных устройства – межсетевых экранов, которые устанавливаются в тех сегментах сети, через которые проходят потоки пакетов. Обычно такие сегменты выбираются между защищаемой сетью и интерфейсом, связанного с этой сетью маршрутизатора. Для реализации вторых методов защиты создаются специальные сетевые шлюзы, в которых реализуется режим туннелирования, причем в этом режиме могут использоваться или не использоваться методы криптографической обработки пакетов. Если такие шлюзы наделены функциями маршрутизации, то одним из перспективных направлений их развития является использование сетевого протокола IPSec, что позволяет с помощью специальных криптографических средств, в частности, формирования электронной цифровой подписи, аутентифицировать заголовки всех сетевых пакетов и гарантировать целостность передаваемых данных.

В настоящее время индустрия телекоммуникаций, обладая избыточной пропускной способностью физических каналов связи, испытывает постоянно возрастающую потребность в эффективных средствах обработки пакетов для маршрутизации и защиты информации. Эти потребности стимулировали широкий спектр исследований по разработке специальных вычислительных устройств, используемых в процессе обработки пакетов. При разработке современных СП должны учитываться тенденции роста пропускных способностей линий связи при использовании оптических сред и применении технологий волнового мультиплексирования. Общие решения задачи повышения производительности вычислителей можно разделить на следующие группы: создание СП на базе параллельных процессоров, использующих разделяемую оперативную память; разработка конвейерных СП, с ресурсами оперативной памяти, распределенными между отельными фазами обработки; гибридные конвейерно-параллельные архитектуры, в которых стадии последовательной и параллельной обработки согласуются с количеством независимых потоков данных.

Эффективность таких решений целиком определяются алгоритмическими особенностями решаемых задач и способом доставки для них данных. В случае обработки сетевых пакетов существенное значение имеют следующие обстоятельства: потоковый характер данных, при котором число одновременно обрабатываемых соединений зависит от числа узлов с различными сетевыми адресами; последовательный способом передачи пакетов в потоке. Так как передача пакетов осуществляется в асинхронном режиме, то есть инициируется каждым узлом независимо, то количество проходящих через маршрутизаторы логических соединений является случайной величиной с фрактальной функцией распределения. Сложный характер процессов пакетной коммутации приводит к тому, что номинальное число параллельных процессоров в архитектуре СП не определяет полностью его производительность, а оптимальное количество стадий конвейерной обработки зависит от характера решаемой задачи и может меняться. Все эти обстоятельства являются предпосылкой для разработки новых подходов к организации процессов обработки сетевых пакетов.

4 Применение распределенных СП в системах защиты

Разработка СП для систем защиты может быть основана на разделении функций обработки пакетов по принципу базовых и дополнительных операций. К базовым операциям относится маршрутизация пакетов, а к дополнительным – остальные операции с пакетами, связанные с расширением функциональности СП, например фильтрация пакетов. Предлагаемое разделение позволяет рассматривать сетевой узел как часть специальной сети обработки пакетов. Топология соединения устройств обработки должна быть такой, чтобы в процессе передачи между ними пакетов не использовались адреса узлов, входящих в таблицу маршрутизации. Применительно к задачам информационной безопасности применение данного подхода позволяет использовать технологию сетевого управления, основанную на использовании системного принципа «безопасность через защиту средств защиты». Этот принцип уравнивает значимость двух ключевых аспектов информационной безопасности, положенных в основу ГОСТ 15408, а именно *функциональность и доверие*. Следование этому принципу означает, что средства, используемые для защиты информации в компьютерных сетях, должны обладать эффективными механизмами обеспечения собственной безопасности, как на стадии разработки (контроль за отсутствием недекларируемых возможностей – НДВ), так и на стадии оперативного функционирования. Для этого на нескольких уровнях модели взаимодействия открытых систем (ВОС) должны использоваться меры, обеспечивающие невозможность локализации места размещения в сети устройств защиты методами удаленного мониторинга. Такая скрытность функционирования приводит к изменению модели защиты, так как большинство существующих средств организации сетевых атак и разрушающих воздействий построена на удаленной нейтрализации устройств, используемых для защиты информационных ресурсов в сети. Создать средства защиты на базе распределенных СП с использованием режима «стелс» возможно потому, что устройства защиты в большинстве режимов функционирования не являются источниками или конечными получателями сетевых пакетов. Поэтому сетевые интерфейсы этих устройств могут не иметь физических и логических адресов, следовательно, характер прохождения через них IP пакетов или MAC кадров аналогичен прохождению их через сетевые концентраторы (HUB) и сегменты кабельных линий, используемых в процессе межсетевого обмена. Применение данного метода сокрытия сетевого адреса расположения средств информационной безопасности с одной стороны обеспечивает выполнение функций защиты, а с другой из-за отсутствия адресов у сетевых интерфейсов устройств обработки пакетов не требует изменения то-

пологии сетевых связей и ранее принятой политики маршрутизации пакетов. Устройства защиты, использующие технологию «стелс обладают рядом преимуществ не только с точки зрения скрытного характера своего функционирования, но также в аспекте масштабирования производительности и повышения уровня надежности работы. Повышение производительности основано на использовании последовательно-параллельного характера сетевого трафика, когда независимые логические соединения формируются путем последовательной передачи отдельных пакетов с определенными адресами источников и приемников сообщений. В результате появляется возможность уменьшения задержек пакетов в цепочке операций:

«фильтрация, обработка, передача пакета в сеть»,

«прием пакетов из сети, фильтрация и обработка»

за счет объединения СП в специализированный вычислительный кластер. Применение режима «стелс для любых типов сетевых устройств, использующих технологию IEEE 802.3 Ethernet,» позволяет организовать процесс обработки пакетов в ядре встраиваемой операционной системы без использования стека протоколов TCP/IP. Такой способ обработки снижает уровень флуктуаций задержек пакетов при их буферизации, что также способствует дополнительному сокрытию места расположения устройств защиты.

5 Заключение

Применение сетевых процессоров с распределенной архитектурой позволят существенно расширить возможности использования средств защиты информации в телематических сетях. Скрытных характер функционирования устройств защиты позволят встроить дополнительные процедуры обработки пакетов в стандартный процесс коммутации без изменения политики маршрутизации. Применение технологи «стелс» снижает затраты на модернизацию сети, так при ее внедрении требуемая вычислительная мощность распределяется между различными сетевыми устройствами. Использование технологии кластеризации СП позволяет эффективно масштабировать производительность сетевых узлов и повышать надежность предлагаемых технических решений.

Литература

[1] Vladimir Zaborovsky. Multiscale Network Processes: Fractal and p-Adic analysis. Proceedings of 10th International Conference on telecommunications (ICT'2003), University of Haute Alsace, Colmar, France, 2003.

[2] Вильчевский Н. О., Заборовский В. С., Клавдиев В. Е., Шеманин Ю. Е. Методы оценки эффективности управления и защиты транспортных соединений в высокоскоростных компьютерных сетях. Материалы конференции «Математика и безопасность информационных технологий (МаБИТ-03)», МГУ им. М.В.Ломоносова, 23-24 октября 2003 г.

[3] Vladimir Zaborovsky, Yuri Shemanin, Jim A. McCombs, Alex Sigalov. Firewall Network Processors: Concept, Model and Platform. Proceedings of International Conference on Networking (ICN'04), Guadeloupe, 2004.

Об одной задаче анализа устойчивости мобильных сетей передачи данных

В. В. Величко, В. К. Попков, О. Д. Соколова, А. Н. Юргенсон

1 Введение

Вопрос оценки устойчивости работы мобильных сетей передачи данных требует комплексного анализа различных взаимосвязанных технологий, обеспечивающих функционирование данного вида связи. Возникающие при этом задачи достаточно сложны из-за специфики мобильной связи - необходимо использовать не привычные стационарные модели и объекты, а рассматривать параметры каждого объекта как функции, зависящие от времени.

Рассмотрим архитектуру мобильной сети связи на примере сети UMTS [1]. С функциональной точки зрения элементы сети объединяются в сеть радиодоступа, которая выполняет необходимые функции, и в базовую сеть, которая осуществляет переключение и маршрутизацию вызовов, а также подключение данных к внешним сетям. Кроме того, в состав сети входит оборудование пользователя. На рис. 1 представлена системная архитектура сети UMTS с указанием принятых открытых интерфейсов.

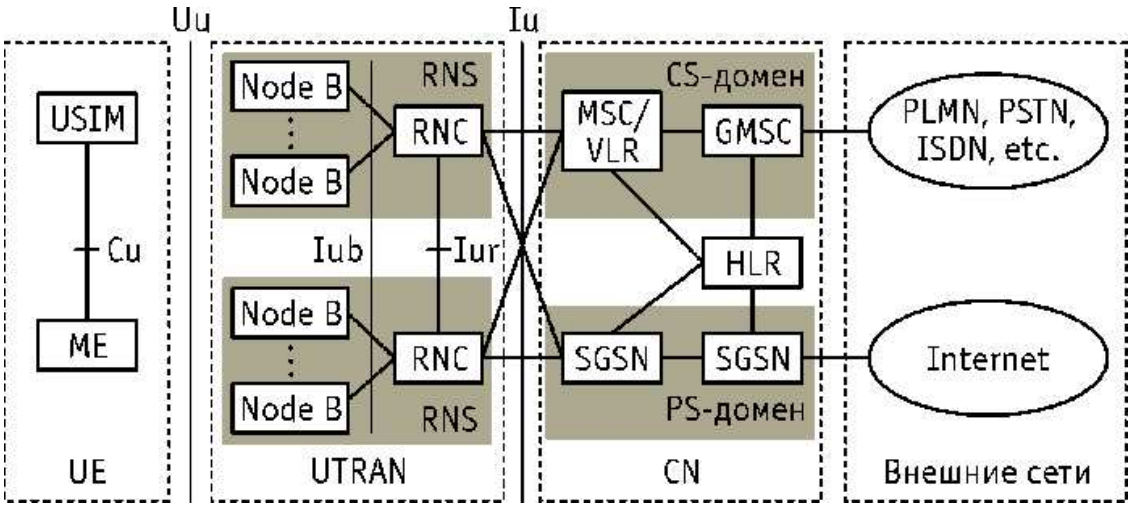


Рис. 1: Архитектура сети UMTS

Функциональное назначение основных структурных элементов сети состоит в следующем. Базовые станции Node B осуществляют организацию радиоканалов по вызовам мобильных абонентов или по своей инициативе при поступлении внешнего вызова. Основной функцией Node B является реализация радиоинтерфейса (обработка радиосигнала, модуляция/демодуляция с расширением/сжатием спектра сигнала, кодирование/декодирование и др.), в том числе, выполнение некоторых операций по распределению радиоресурсов сети.

Контроллер сети радиодоступа осуществляет управление базовыми станциями и взаимодействует с центром коммутации сети. Его основными функциями являются управление распределением радиоканалов, контроль соединений, регулирование их очередности, удаленная динамическая коммутация, а также контроль за распределением абонентской нагрузки.

Для исследования живучести мобильной сети связи в условиях разрушающих информационных воздействий удобно рассмотреть композицию нескольких моделей подсистем более низкого уровня:

абонентское оборудование, система радиодоступа, базовая сеть, шлюз для коммутации с внешними сетями. Поскольку основные процессы разрушения и восстановления в подсистемах происходят независимо друг от друга, становится возможным декомпозиция общей задачи, что позволяет использовать математический аппарат для детального исследования живучести подсистем.

Заметим, что проблема исследования живучести базовой сети закрывается многочисленными исследованиями живучести стационарных сетей [2]. Таким образом, с целью учета специфики живучести мобильной сети, имеет смысл сделать акцент на анализе модели системы радиодоступа, и в частности на мониторинге потоков передаваемой информации.

2 Модель передачи данных абонентами мобильной сети

Для моделирования процесса передачи данных нестационарными абонентами предлагается использовать гиперсеть, в которой вершины, ребра и ветви имеют параметры, зависящие от времени. Такая нестационарная гиперсеть как математический объект была специально разработана для анализа сетей связи с подвижными объектами [3].

Базовую сеть мобильной связи можно представить графом $G(X, R)$, в котором вершинам соответствуют базовые станции сети, а ребрам - линии связи. Абоненты могут быть связаны с любой базовой станцией по соответствующему радио-каналу. Вершине z инцидентно $n(z)$ различных гиперребер, если соответствующая базовая станция может работать на $n(z)$ радио-каналах. Таким образом, первичная сеть гиперсети задается гиперграфом, вторичная сеть гиперсети задается мультиграфом.

Определение. Нестационарная гиперсеть $NS(t) = (X, V, R)$ включает в себя:

- $X = (x_1, \ldots, x_n)$ - множество вершин (базовые станции);
- $V = (v_1, \ldots, v_g)$ - множество ветвей (выделенные линии связи, соединяющие базовые станции);
- $R = (r_1, \ldots, r_m)$ - множество ребер (виртуальные каналы, по которым ведется передача данных).

При передаче данных по мобильной сети центр коммутации выделяет виртуальные каналы таким образом, чтобы рационально использовать пропускную способность каждого канала для удовлетворения одновременно нескольких запросов. Каждой ветви $v \in V$ сопоставлена $\alpha \geqslant 0$ пропускная способность ветви (выделенной линии связи). Каждому ребру $r \in R$ сопоставлена $\delta(t) \geqslant 0$ пропускная способность ребра, в соответствии с которой данное ребро может предоставляться для передачи объема информации в момент времени t.

Процедура передачи информации: В каждый момент времени возникает пара абонентов (мобильных терминалов), между которыми происходит процесс передачи пакетов данных. Базовая станция после получения запроса выделяет подходящее ребро гиперсети по остаточному принципу на данный момент времени (т.е. оставляется резерв для активных абонентов).

Процедура перехода: В каждый момент времени базовая станция может передать абонента соседней базовой станции (в случае перехода абонента в зону действия другой станции).

Процедура окончания передачи информации: После сеанса передачи информации соответствующее активное ребро вторичной сети становится неактивным (ресурс ребра восстанавливается).

3 Моделирование процесса атак

Базовые станции могут быть выведены из строя различными способами - физическое уничтожение, включение широкополосной помехи, вывод некоторых каналов из строя узкополосной радио помехой. Кроме того, современные сети мобильной связи могут быть атакованы разрушающими информационными воздействиями (РИВ). Следовательно, при моделировании воздействий на сеть необходимо вычислять различные показатели: пропускная способность, время задержки пакетов или сообщений в сети и т. д.

При моделировании атаки на мобильные терминалы рассматривается дополнительный поток (назовем его «ложной» информацией), посылаемый зараженным абонентом на другие терминалы, находящиеся в данный момент времени на связи с исходным терминалом. По некоторому алгоритму, зависящему от типа атаки, происходит заражение связанных терминалов. Введем характеристики для

осуществления мониторинга потоков в мобильной сети. На каждом интервале времени отслеживаем следующие параметры: заявки по передаче информации, реальный поток (вместе с пакетами «ложной» информации), количество пар абонентов закончивших передачу информации, количество пар абонентов которым было отказано в обслуживании. При распространении атаки во времени растет число зараженных терминалов, растет поток передаваемой информации (за счет большого количества «ложных» пакетов) и, следовательно, падает качество обслуживания в сети, т. к. появляются отказы на заявки обслуживания из-за загруженности каналов. В качестве примера работы модели мобильной сети в условиях атаки на один терминал рассмотрена нестационарная гиперсеть с числом базовых станций - 20, количеством абонентов - 4 000. На графиках рис. 2 показано увеличение передаваемого по сети потока за счет растущего объема «ложных» пакетов информации.

Рис. 2: Моделирование потоков информации в мобильных сетях

Можно усложнить задачу, рассматривая одновременную атаку нескольких мобильных терминалов, что более адекватно реальным ситуациям. Таким образом, применяя нестационарную гиперсеть в качестве модели мобильной сети передачи данных, можно показать, как с помощью мониторинга различных характеристик (например, передаваемых потоков информации) отслеживать несанкционированный доступ в сеть. Дальнейшие исследования в этой области позволят получить показатели живучести и качества обслуживания для мобильных сетей связи, разработать методики применения результатов теории для оценки этих показателей на практике.

Литература

[1] UTRA (UE) TDD; Radio transmission and Reception. - 3GPP TS 25.102 v3.4.0 . 2000.

[2] Попков В. К. Математические модели живучести сетей связи. Новосибирск, 1990.

[3] Попков В. К. Математические модели связности. Новосибирск, 2002.

Язык описания моделей разграничения доступа и его реализация в ядре операционной системы Linux

О. О. Андреев

Введение

В связи с широким использованием информационно-вычислительных комплексов для решения практически значимых задач все больший интерес в последнее время проявляется к политикам их информационной безопасности и, в частности, к моделям разграничения доступа. Модель разграничения доступа занимает одно из центральных мест в числе других компонент политики безопасности, включающей также средства идентификации/аутентификации, шифрования и иные компоненты, многие из которых встроены в операционные системы (ОС) и поставляются вместе с ними [4].

В современных ОС, таких как Linux и Windows, используются механизмы разграничения доступа, основанные на моделях, разработанных еще в 70-х годах, таких как дискреционная [1] или мандатная [2]. Дискреционная модель представляет собой достаточно примитивную, хотя и простую для описания модель логического разграничения доступа, политики безопасности на основе которой получаются громоздкими в представлении, сложноверифицируемыми и неудобными для управления. Мандатная модель разграничения доступа на основе упорядоченных меток безопасности является, наоборот, слишком жесткой с точки зрения условий, которые она реализует, удобной в верификации, простой в представлении и настройке, однако пригодной для реального применения в политиках безопасности весьма узкого класса информационных систем. Реализация политик безопасности, использующих современные и более сложные модели разграничения доступа, такие как ролевая [6], сопряжена с большими трудностями внедрения дополнительных механизмов в ядра операционных систем и их использованием в составе программных комплексов. В UNIX-подобных системах, например, существуют дополнительные проблемы, связанные с пользователем *root*. В таких системах доступ *root* требуется для выполнения большого объема действий по администрированию системы. При этом, наличие данного доступа дает неограниченные привилегии его владельцу, что формирует «грубую» модель по принципу «все или ничего». Для решения отмеченной проблемы были созданы дополнительные средства безопасности, такие как `sudo` — команда, позволяющая непривилегированному пользователю исполнять некоторое множество программ с повышенными привилегиями. Однако, эти средства не решают проблему в полном объеме, так как имеют низкую гранулярность разграничения доступа — в пределах одной программы (процесса, порожденного запуском исполняемого файла и всех создаваемых им процессов).

Отмеченные выше и ряд других недостатков эксплуатирующихся в настоящее время систем стимулируют работы по созданию новых логико-языковых средств описания моделей разграничения доступа, таких как eXtended Access Control Language [5], Enterprise Privacy Authorisation Lanuage [7]. Данные средства предоставляют пользователю, отвечающему за информационную безопасность (в дальнейшем именуемому *офицером безопасности*) возможность самому определить модель разграничения доступа, лучше всего подходящую под нужды защищаемого информационно-вычислительного комплекса, или выбрать требуемую модель среди предложенных другими разработчиками. Кроме отмеченных выше, активно проводятся работы по созданию механизмов, позволяющих модифицировать стандартные модели контроля доступа в популярных операционных системах, подобных RSBAC Linux [8].

Целью настоящей работы является исследование подходов и разработка на их основе перспективных логико-языковых средств описания моделей разграничения доступа, их реализация в операционной системе Linux. Основными ее результатами являются разработанный автором язык описания моделей разграничения доступа и надстройка над ОС Linux, позволяющая на основе предложенного языка задавать политики разграничения доступа к локальным файлам и устройствам. Следует отметить,

что данный язык является универсальным и может использоваться при формировании и реализации политик безопасности в распределенных информационно-вычислительных комплексах на основе ОС Linux.

1 Формальное описание

В настоящем разделе представлено формальное описание класса моделей, которые могут быть заданы с помощью предложенного в настоящей работе языка описания моделей разграничения доступа.

Под политикой безопасности (ПБ) информационно-вычислительного комплекса понимается ограничение на функционирование этого комплекса в соответствии с набором неформально задаваемых правил. Одним из аспектов ПБ является контроль за доступом со стороны пользователей комплекса к его ресурсам. Этот контроль осуществляется подсистемами разграничения доступа каждого из компонентов комплекса на основе некоторой модели логического разграничения доступа. Модель логического разграничения доступа и конфигурация механизмов разграничения задается в ПБ. Одним из важнейших ресурсов в информационно-вычислительных комплексах являются локальные файловые системы. Доступ к ним регламентируется с помощью механизмов ОС. Предлагаемый язык позволяет задавать семейство моделей и определяет возможности по конфигурированию основывающихся на них механизмов разграничения доступа.

Как и большинство других, модели разграничения доступа, задаваемые данным языком, базируются на трех основных понятиях: *субъект*, *объект* и *доступ*. Основной задачей подсистемы разграничения доступа является выдача ответов на запросы «может ли конкретный субъект получить желаемый доступ к данному объекту». Дополнительными (смежными) задачами, решение которых может поддерживаться в процессе эксплуатации системы, являются протоколирование удавшихся или неудавшихся попыток доступа и (или) предоставление данных системе обнаружения вторжений (IDS — Intrusion detection system) и другие.

Решение вопроса о разрешении или запрещении доступа подсистемой его разграничения может основываться на различных данных. В случае дискреционной политики такое решение принимается на основе значений идентификаторов субъекта, объекта и доступа. В случае многоуровневой политики — оно является результатом анализа меток безопасности (так называемых уровнях секретности в модели Белла — Лападула или уровнях целостности в модели Биба) субъекта и объекта и типе доступа. Предложенный язык описывает класс моделей, являющихся, в некотором смысле, расширением принципов, положенных в основу многоуровневой модели разграничения доступа.

Основным понятием, на котором базируются модели разграничения доступа, задаваемые предлагаемым языком, является понятие *атрибута*. Каждый объект и субъект может иметь некоторые задаваемые пользователем или системой атрибуты. Доступ разрешается или запрещается в соответствии со значением, выдаваемым заданной в конкретной модели булевозначной функции, называемой *функцией доступа* от атрибутов субъекта, объекта, запрашиваемого доступа и определенных системных переменных. Таким образом, для задания модели разграничения доступа требуется определение множества атрибутов и задание булевозначной функции от этих атрибутов, а для задания конкретной политики, основанной на этой модели — значений атрибутов для конкретных объектов и субъектов. Примерами атрибутов могут быть данные о том, в какой должности находится пользователь, каков уровень секретности у запрашиваемого объекта или время последнего обращения пользователя к объекту. Таким образом, в качестве атрибута может записываться некоторое свойство объекта или субъекта, которое в соответствии с принятой политикой безопасности влияет на разрешение доступа, функция доступа является переложением на формальный язык правил разграничения доступа, присутствующих в политике.

Традиционные модели разграничения доступа являются статичными, то есть неизменными во времени. Это не позволяет офицеру безопасности задавать сложные модели и базирующиеся на них политики безопасности, в которых, скажем, доступ может зависеть от истории предыдущих доступов пользователя. Представляемый в данной работе язык дает возможности по заданию такого рода политик. Для этого вводится понятие *пост-действия*. Это действия, которые должны быть выполнены при каждой попытке доступа. В предлагаемом языке пост-действия могут изменять атрибуты участвующих в попытке доступа субъекта и объекта. Примером возможного пост-действия является увеличение атрибута «количество неудавшихся попыток доступа» на единицу при каждой неудачной попытке доступа. Такой атрибут может использоваться, скажем, для полного запрещения доступа при

превышении некоторого порогового значения.

Представим следующее формальное определение описанного класса моделей.

Определение. Зададим:

- S, множество субъектов;
- O, множество объектов;
- A, множество доступов;
- $Attr = Attr_S \sqcup Attr_O$, множество атрибутов субъектов ($Attr_S$) и объектов ($Attr_O$);
- $Value$, множество значений атрибутов;
- $V_S\colon S \times Attr_S \to Value$, $V_O\colon O \times Attr_O \to Values$, функции, выдающие значения атрибутов;
- $P\colon Value^{|Attr|} \times A \to \{True, False\}$, функция доступа;
- $Success\colon Attr \times Value \to (Attr_S \sqcup Attr_O) \times Value$, функция, устанавливающие значения атрибутов при разрешенном доступе;
- $Fail\colon Attr \times Value \to (Attr_S \sqcup Attr_O) \times Value$, функция, устанавливающие значения атрибутов при запрещенном доступе;

Тогда, если субъект s хочет получить доступ a к объекту o, то система разграничения доступа вычисляет значение $P(V_S(s, attr_S^1), \ldots, V_O(o, attr_O^1), \ldots, V_A(a, attr_A^1), \ldots)$ и если это значение $True$, то доступ разрешается, а если значение $False$, то доступ запрещается. При разрешении доступа атрибуты s и o переустанавливаются в $Success(V_S(s, attr_S^1), \ldots, V_O(o, attr_O^1), \ldots, a)$, а при отказе в доступе — в $Fail(V_S(s, attr_S^1), \ldots, V_O(o, attr_O^1), \ldots, a)$.

Предлагаемый язык не позволяет задавать произвольные функции P, $Success$ и $Fail$. Вместо этого, для задания функции P пользователю предоставляется некоторый класс булевозначных функций от атрибутов типа «неравно» или «больше» и позволяется задавать функцию доступа как суперпозицию булевой функции и функций из фиксированного класса. Для задания функций $Success$ и $Fail$ также предоставляется некий (весьма узкий) класс функций от атрибутов и констант. Функции пост-действия задаются как последовательности присваиваний атрибутам значений функций из этого класса от атрибутов субъекта и объекта. Каждый шаг пост-действия может быть выполнен в зависимости от истинности или ложности задаваемого для него условия, аналогичного P.

Такое ограничение класса функций доступа и функций пост-действия позволяет не только упростить синтаксис языка, но и представляет возможность создания автоматизированных средств для анализа моделей логического разграничения доступа, заданных с помощью предложенного языка. Такие средства особенно необходимы при создании динамических моделей разграничения доступа, в которых решение вопроса, «сможет ли указанный субъект *когда либо* получить доступ к данному ресурсу» представляется неочевидным. Создание автоматизированных средств позволит облегчить труд офицера безопасности по проверке соответствия заданных моделей разграничения доступа и конфигураций программных средств, реализующих их, правилам политики безопасности. Кроме этого, необходимость в таких средствах может возникнуть при интегрировании политик безопасности и, как следствии, моделей разграничения доступа. В этом случае требуется проверить противоречат ли политики друг другу, является ли одна политика более сильной чем другая. Такого рода вопросы также должны решаться с помощью автоматизированных средств анализа.

2 Семантика языка

В данном разделе будет показано как класс моделей разграничения доступа, описанный в предыдущем разделе, может быть реализован с использованием предлагаемого в настоящей работе языка.

Представление модели в данном языке состоит из структурных единиц — *подмоделей*. Каждая из подмоделей представляет собой отдельную модель разграничения доступа, применимую к части или ко всем запросам на доступ. Решение о выдаче или отказе в доступе принимается на основании

анализа результатов, полученных при применении каждой из подмоделей, для которых запрашиваемый доступ находится в области применимости. Такое объединение нескольких подмоделей в единую облегчает установку и настройку системы, в которой будет применяться предложенный язык, позволяя по мере необходимости добавлять или удалять подмодели для ужесточения или ослабления общей политики разграничения доступа. Поставщики системы могут снабжать ее заданными подмоделями разграничения, из которых впоследствии офицером безопасности будет конструироваться единая модель.

В свою очередь, каждая подмодель подразделяется на *правила*. Такое подразделение, помимо прочих преимуществ, облегчает написание подмодели и повышает ее восприятие при чтении. Каждое правило состоит из *цели*, *условия* и четырех пост-действий. Цель определяет границы применимости правила, условие определяет результат его применения, а пост-действия определяют действия, выполняемые при его применении. Цель и условие задают пару функций того же типа (с тем же набором аргументов и получаемым значением), что и P, а пост-действия — функции того же типа, что и $Success$ и $Fail$.

Вычисление функции доступа подсистемой его разграничения, реализующей данный язык, происходит следующим образом.

1. Для каждой подмодели в составе подсистеме выполняются пункты 2–4.
2. Для каждого правила в подмодели вычисляется цель.
3. Если цель истинна, то вычисляется условие, иначе переходим к следующему правилу.
4. Если условие истинно, то добавляем правило в список истинных правил, иначе добавляем правило в список ложных правил.
5. Если после обхода всех политик список ложных правил оказывается не пуст, то доступ запрещается. Если список ложных правил оказывается пуст, а список истинных правил — не пуст, то доступ разрешается. В случае, когда оба списка пусты, принимается решение по умолчанию.

Решение вопроса по умолчанию зависит от реализации подсистемы разграничения доступа. Например, это решение может прописываться в некотором конфигурационном файле. Решение этого вопроса в представленной в данной работе реализации языка будет изложено в разделе 4.

Выполнение пост-действий происходит после создания списка ложных и истинных правил. Четыре пост-действия соответствуют следующим случаям.

1. Правило истинно при положительном результате.
2. Правило ложно при положительном результате.
3. Правило истинно при отрицательном результате.
4. Правило ложно при отрицательном результате.

Реализация языка должна гарантировать, что пост-действия правил будут выполняться в том порядке, в котором правила записаны в подмодели, однако она не должна гарантировать какую-либо очередность выполнения правил из разных политик.

У каждого объекта и субъекта существует стандартный, определяемый реализацией языка набор атрибутов. Примерами могут служить имя файла и его владелец. Эти атрибуты не могут переустанавливаться пользователем или офицером безопасности. Кроме этого, пользователю или офицеру безопасности разрешается устанавливать дополнительные атрибуты с произвольными именами. Именем атрибута называется строка символов произвольной длины. Множество типов атрибутов является объединением булевого, вещественного и строкового типов, а также специально выделенного значения nil. Это значение используется, если атрибут не установлен. Для установленных офицером безопасности атрибутов может быть определено, разрешено ли изменять их непривилегированному пользователю.

Системные переменные определяются реализацией языка и не могут быть переустановлены.

Семантика языка позволяет задавать сложные модели разграничения доступа, оставляя при этом получающиеся описания простыми для чтения и понятными. Разделение на подмодели позволяет переложить существенную часть работы по созданию моделей на плечи поставщиков ОС, одновременно позволяя офицеру безопасности специализировать модель для конкретной системы и политики безопасности.

3 Синтаксис языка

Рассмотрим синтаксис языка и выражение с его помощью описанных в предыдущем разделе конструкций.

Обсуждаемый в настоящей работе язык представляет собой подмножество XML. Один XML-документ описывает одну подмодель разграничения доступа. Такое разделение позволяет четко обозначить разделение единой модели на подмодели.

Корневым элементом всегда должен являться элемент `<policy>`. Он может содержать элементы:

- `<rule>`,
- `<description>`.

Подмодель записывается следующим образом:

```
<policy>
    <description>
        ...
    </description>
    <rule>
        ...
    </rule>
    ...
    <rule>
        ...
    </rule>
</policy>
```

Элемент `<description>` носит роль комментария и используется для словесного описания подмодели или одного из ее правил. Он игнорируется при разборе.

Элемент `<rule>` описывает правило подмодели. Он может содержать элементы:

- `<description>`,
- `<target>` (не более одного элемента),
- `<condition>` (не более одного элемента),
- `<rule-success-policy-success-action>` (не более одного элемента),
- `<rule-fail-policy-success-action>` (не более одного элемента),
- `<rule-success-policy-fail-action>` (не более одного элемента),
- `<rule-fail-policy-fail-action>` (не более одного элемента).

Если элемент `<condition>` отсутствует, то условие считается всегда истинным. Если элемент `<target>` отсутствует, то цель считается всегда истинной. Если один из элементов `<*-action>` отсутствует, то соответствующее действие считается ничего не выполняющим.

Общий вид одного правила:

```
<rule>
    <description>
        ...
    </description>
    <target>
        ...
    </target>
    <condition>
        ...
    </condition>
```

```
        <rule-success-policy-success-action>
            ...
        </rule-success-policy-success-action>
        <rule-fail-policy-success-action>
            ...
        </rule-fail-policy-success-action>
        <rule-success-policy-fail-action>
            ...
        </rule-success-policy-fail-action>
        <rule-fail-policy-fail-action>
            ...
        </rule-fail-policy-fail-action>
    </rule>
```

Элементы <target> и <condition> должны содержать булевозначные выражения. Эти выражения могут являться элементами:

- <and>, задающий функцию «и»,
- <or>, задающий функцию «или»,
- <not>, задающий функцию «не»,
- <less>, задающий функцию «меньше»,
- <greater>, задающий функцию «больше»,
- <lequal>, задающий функцию «не больше»,
- <gequal>, задающий функцию «не меньше»,
- <equal>, задающий функцию «равно»,
- <nequal>, задающий функцию «неравно»,
- <substr>, задающий функцию «является подстрокой»,
- <strprefix>, задающий функцию «является началом строки»,
- <strpostfix>, задающий функцию «является концом строки»,
- <exists>,
- <access-is>,

Элементы <and>, <or> и <not>, находящиеся внутри <target> или <condition>, являются логическими связками и должны содержать два (в случае <and> и <or>) или одно (в случае <not>) булевозначное выражение.

Приведем пример записи цели:

```
    <target>
        <and>
            <not>
                ...
            </not>
            <or>
                <not>
                    ...
                </not>
                <or>
                    ...
                </or>
```

```
            </or>
        </and>
    </target>
```

Элементы `<less>`, `<greater>`, `<lequal>`, `<gequal>`, `<equal>`, `<nequal>`, `<substr>`, `<strprefix>`, `<strpostfix>` являются булевозначные функциями от двух аргументов, которые могут задаваться элементами `<subject>`, `<object>`, `<system>`, `<bool>`, `<integer>`, `<string>`, `<float>`.

Функции «меньше», «больше», «не больше», «не меньше», «равно» и «неравно» определены на всем множестве значений атрибутов, а функции «является подстрокой», «является началом строки» и «является концом строки» определены на строковых значениях.

Если одно или оба значения не лежат в классе допустимых значений, функция выдает «ложно». Если одно или оба значения равны nil, то все эти функции, кроме «неравно» выдают «ложно», а функция «неравно» выдает «истинно».

Следующее выражение проверяет, оканчивается ли значение атрибута субъекта `e-mail` на `@molvania.com`:

```
    <strpostfix>
        <subject>e-mail</subject>
        <string>@molvania.com</string>
    </strpostfix>
```

Элемент `<access-is>` задают функции, проверяющие равенство типа доступа заданному значению. Он должны содержать строку.

Данное выражение проверяет, запрашивается ли доступ `execute`:

```
    <access-is>execute</access-is>
```

Элемент `exists` задает функцию, проверяющую существование у субъекта или объекта заданного атрибута (т. е. равенство значения этого атрибута nil). Он должен содержать один элемент `<subject>` или `<object>`.

Следующая функция проверяет, существует ли у субъекта атрибут `e-mail`:

```
    <exists>
        <subject>e-mail</subject>
    </exists>
```

Элементы `<subject>`, `<object>` и `<system>` задают атрибуты субъекта, объекта или системные переменные, такие как время. Они должны содержать строку.

Данный элемент обозначает значение атрибут `domainname` субъекта:

```
    <system>domainname</system>
```

Элементы `<bool>`, `<integer>` `<string>`, `<float>` задают булевы, целочисленный, строковые и вещественные константы соответственно. Они должны содержать строку.

Такая запись обозначает булевское значение $True$:

```
    <bool>true</bool>
```

Элементы `<*-action>` задают пост-действия. Они должны содержать не менее одного элемента `<set>`.

Общий вид пост-действия таков:

```
    <success-action>
        <set>
            ...
        </set>
        ...
        <set>
            ...
        </set>
    </success-action>
```

Элемент <set> указывает системе установить атрибут в заданное значение. Атрибуты задаются с помощью элементов <subject> и <object>, а значения — с помощью следующих элементов:

- <subject>,
- <object>,
- <system>,
- <bool>,
- <string>,
- <float>,
- <and>,
- <or>,
- <xor>,
- <not>,
- <add>, определяющего функцию «сложить»,
- <subtract>, определяющего функцию «вычесть»,
- <concat>, определяющего функцию «присоединить строку»

Данная запись обозначает установку значения атрибута субъекта last-access в значение атрибута объекта id:

```
<set>
    <subject>last-access</subject>
    <object>id</object>
</set>
```

Элементы <and>, <or>, <not> (находясь внутри элементов <*-action>), <xor>, <add>, <subtract> и <concat> задают функции от двух аргументов, задаваемых элементами <subject>, <object>, <system>, <bool>, <string>, <float>, <and>, <or>, <xor>, <not>, <add>, <subtract> и <concat>.

Если подать одной из функций на вход несоответствующее ей значение, весь шаг действия, определяемый элементом <set> не будет выполнен.

В данном примере, атрибут num-accesses увеличивается на единицу:

```
<set>
    <subject>num-accesses</subject>
    <add>
        <subject>num-accesses</subject>
        <float>1</float>
    </add>
</set>
```

В качестве примера, подтверждающего выразительность и эффективность представленного языка, приведем запись модели, реализующей простейший вариант многоуровневой политики. Разрешены две операции: чтение и запись. При этом, чтение разрешается тем субъектам, метка которых не ниже, чем метка объекта, а запись — тем, у которых метка не выше, чем метка объекта.

```
<policy>
    <rule>
        <description>
            ss-property
        </description>
        <target>
```

```
            <and>
                <access-is>read</action-is>
                <or>
                    <equal>
                        <object>type</object>
                        <string>dir</string>
                    </equal>
                    <equal>
                        <object>type</object>
                        <string>file</string>
                    </equal>
                </or>
            </and>
        </target>
        <condition>
            <gequal>
                <subject>level</subject>
                <object>level</object>
            </gequal>
        </condition>
    </rule>
    <rule>
        <description>
            *-property
        </description>
        <target>
            <and>
                <access-is>write</action-is>
                <or>
                    <equal>
                        <object>type</object>
                        <string>dir</string>
                    </equal>
                    <equal>
                        <object>type</object>
                        <string>file</string>
                    </equal>
                </or>
            </and>
        </target>
        <condition>
            <lequal>
                <subject>level</subject>
                <object>level</object>
            </lequal>
        </condition>
    </rule>
</policy>
```

Как видно из приведенного примера, синтаксис языка несложен и достаточно понятен даже человеку, незнакомому с деталями языка. Выразительные возможности его высоки и при этом он не перегружен излишними деталями.

4 Детали реализации языка в ОС Linux

Вновь разработанный язык реализован на основе подсистемы RSBAC, представляющей собой патч (patch) к ядру ОС Linux и набор дополнительных программ, и включенной в некоторые дистрибутивы Linux. К таким дистрибутивам относятся, например, Mandrake/Mandriva Linux, ALT Linux Castle. Данная подсистема позволяет контролировать в отдельности любой доступ к объектам, находящимся под контролем системы, включающим файлы, директории, устройства, средства межпроцессного взаимодействия, процессы, сетевые каналы, системные данные (время, журнал событий и другие параметры). При этом созданы средства для мгновенного вступления в силу изменений в политике доступа (то есть, права доступа проверяются для каждой операции чтения и модификации объекта в отличие от классического Linux, что уменьшает вероятность появления так называемых race conditions). Кроме того, система предоставляет пользователю возможность загрузки собственных модулей контроля доступа (decision modules) в дополнение к уже присутствующим в ядре.

Для ускорения процесса загрузки политик, который происходит после каждой загрузки системы, политики преобразуются из первоначального XML-представления в бинарный формат. Это преобразование осуществляется с помощью программы `xmlplc`. Бинарный формат представляет собой участок памяти с добавленной информацией о перемещениях (relocations). Такой выбор формата обусловлен прежде всего скоростью и простотой обработки. После загрузки файла в память необходимо лишь изменить некоторые адреса в соответствии с relocations.

После преобразования политика может быть загружена в ядро с помощью утилиты `xmlplset`. Эта же утилита позволяет пользователю установить атрибуты субъектов и объектов. Установленные атрибуты сохраняются между перезагрузками системы в отличие от политик.

Стандартным атрибутом объекта является его тип, значениями этого атрибута может быть «file», «directory» или «device». Стандартные атрибуты для файлов и директорий включают имя файла, имя владельца и группы-владельца файла, для устройств — minor и major номера. Стандартными атрибутами субъекта является имя исполняемого файла и имя пользователя, запустившего процесс. Системные переменные включают текущее время, имя системы.

Для каждого атрибута может быть установлен флаг `inherited`, который указывает на условие, будет ли данный атрибут унаследован потомком вместе со своим значением, вместе с указанным значением или не будет наследоваться вовсе. Установка этого флага имеет смысл, например, для директорий. Пользователь может изменить значение этого флага, если он может изменить значение самого атрибута. Для каждого атрибута может быть установлен флаг `user-modified`, позволяющий пользователю, являющемуся владельцем объекта, модифицировать данный атрибут. Установка этого флага, равно как и модификация атрибутов, для которых этот флаг не установлен, доступна только офицеру безопасности.

В упомянутом в разделе2 случае, если запрашивается решение по доступу, который не подходят ни под одно правило, модуль вернет `DO_NOT_CARE`, что предоставит решение вопроса о предоставлении доступа другим RSBAC-подсистемам. Следует иметь ввиду тот факт, что если данный модуль является единственным принимающим решения и стандартные средства дискреционного разграничения доступа отключены, то доступ будет разрешен.

Заключение

Предложенный язык представляет собой эффективное средство для описания достаточно сложных моделей разграничения доступа. Разделение модели на структурные единицы облегчает написание моделей, позволяет легко добавлять по необходимости новые компоненты и устранять лишние. Возможность самостоятельной установки значений атрибутов избавляет пользователя от необходимости обращаться к офицеру безопасности для настройки доступа к своим данным. Например, пользователь может устанавливать на данные атрибут, определяющий ценность информации для этого пользователя, присваивая важным документам высший уровень ценности, а на музыкальные или видеофайлам, например, — низший.

Реализация данного языка в ядре ОС Linux позволяет легко и без существенных изменений перевести существующие системы и политики безопасности на новую модель разграничения доступа, одновременно реализуя дискреционной контроль доступа к большей части файлов и модель, основанную на этом языке, для контроля доступа к выделенной части важных данных.

Рассматривая пути продолжения данной работы, хотелось бы остановиться на ряде перспективных аспектов. Как уже было упомянуто в разделе1, одной из основных задач, стоящих перед разработчиками языка является создание автоматизированных средств проверки политик на соответствие некоторым свойствам. Такие средства возможно создать именно благодаря намеренному ограничению функциональности. При этом, автору представляется, что такое снижение является весьма разумной платой за появление таких возможностей. Как уже было упомянуто, автоматизированные средства позволяют существенно упростить работу офицера безопасности и избежать угроз информационной безопасности, связанных с неправильной конфигурацией средств логического разграничения доступа.

Другими направлениями развития работы могут быть добавление в язык дополнительных типов, таких как множество значений, и дополнительных функций и действий. Также предполагается внедрение данного языка в другие системы разграничения доступа, например, в Web-сервер.

Литература

[1] A guide to understanding discretionary access control in trusted systems, NCSC-TG-003. National Computer Security Center, 1987.

[2] Bell D., LaPadula L. Secure computer systems: Unified exposition and multics interpretation, Technical Report MTR-2997. Mitre Corporation, 1976.

[3] Андреев О. О. Сравнение ролевой и дискреционной модели разграничения доступа. Материалы конференции МаБИТ-04. М.: МЦНМО, 2005, с. 284–291.

[4] Васенин В. А. Проблемы математического, алгоритмического и программного обеспечения компьютерной безопасности в Интернет. Материалы конференции МаБИТ-03. Москва, М.: МЦНМО, 2004, с. 111–143.

[5] eXtensible Access Control Markup Language (XACML) Commitee Specification. OASIS Open, 2003.

[6] Ferrariolo D., Kuhn R. Role-Based Access Controls. 15th National Computer Security Conference, 1992.

[7] Enterprise Privacy Authorization Language. IBM Research Report, 2003.

[8] `http://www.rsbac.org`.

Проактивная безопасность и самокорректирующиеся среды

О. В. Казарин

1 Введение

Проактивная безопасность компьютерной системы (КС) — это органичное структурное свойство КС, которое позволяет ей сохранять функциональность и защищенность своих информационных ресурсов от кибератакующих действий, как на этапе разработки, так и на этапе эксплуатации КС.

Более того, если такая гипотетическая проактивно безопасная КС будет создана, то потенциальный пользователь (эксплуатирующая организация) в большинстве случаев может «просто» не заботиться о том, подвергается ли его КС кибератакам или нет? Эта КС, в таком случае, «просто проглотит» кибератаку, сохраняя при этом свою функциональность.

Основные элементы методологии создания безопасных КС рассмотрены в работе [1], в том числе и в ее проактивной части. В работе [2] в качестве основного алгоритмического инструмента для создания проактивно безопасных распределенных КС предлагается использовать распределенные алгоритмы (протоколы) конфиденциальных вычислений [3, 4] или, в более общем случае, протоколы, которые реализуют свою целевую функцию, даже если некоторые из участников протокола (некоторые из процессоров распределенной вычислительной системы) отклоняются от предписанных протоколом действий (например, протоколы, устойчивые к сбоям, протоколы византийских соглашений, протоколы консенсуса и др.).

В данной работе предлагается использовать для создания проактивно безопасных КС алгоритмический инструментарий, использующий методологию самотестирования и самокоррекции программ [3, 4], который наряду с самокорректирующимися схемами (например, см. определения из работ [5, 6]), может стать одним из фундаментальных «кирпичиков» для создания таких КС.

2 Основные элементы методологии создания самотестирующихся и самокорректирующихся программ

2.1 Общая постановка задачи

Пусть необходимо написать программу P для вычисления функции f так, чтобы $P(x) = f(x)$ для всех значений x. Традиционные методы тестирования программ не позволяют убедиться с вероятностью 1 в корректности результата выполнения программы, хотя бы потому, что тестовый набор входных данных, как правило, не перекрывают весь их возможный спектр. Один из методов решения данной проблемы заключается в создании так называемых самокорректирующихся и самотестирующихся программ, которые позволяют оценить вероятность некорректности результата выполнения программы, то есть, что $P(x) \neq f(x)$ и корректно вычислить $f(x)$ для любых x, в том случае, если сама программа P на большинстве наборах своих входных данных (но не всех) работает корректно.

Чтобы добиться корректного результата выполнения программы P, вычисляющей функцию f, необходимо написать такую программу T_f, которая позволяла бы оценить вероятность того, что $P(x) \neq f(x)$ для любых x. Такая вероятность будет называться *вероятностью ошибки* выполнения программы P. При этом T_f может обращаться к P как к своей подпрограмме.

Обязательным условием для программы T_f является ее принципиальное *временное отличие* от любой корректной программы вычисления функции f, в том смысле, что время выполнения программы T_f, не учитывающее время вызовов программы P, должно быть значительно меньше, чем время выполнения любой корректной программы для вычисления f. В этом случае, вычисления согласно T_f

некоторым количественным образом должны отличаться от вычислений функции f и *самотестирующаяся программа* может рассматриваться как независимый шаг при верификации программы P, которая предположительно вычисляет функцию f. Кроме того, желательное свойство для T_f должно заключаться в том, чтобы ее код был насколько это возможно более простым, то есть T_f должна быть *эффективной* в том смысле, что время выполнения T_f даже с учетом времени, затраченного на вызовы P должно составлять константный мультипликативный фактор от времени выполнения P. Таким образом, самотестирование должно лишь незначительно замедлять время выполнения программы P.

Пусть π означает некоторую вычислительную задачу и/или некоторую задачу поиска решения. Для x, рассматриваемого в качестве входа задачи, пусть $\pi(x)$ обозначает результат решения задачи π. Пусть P — программа (предположительно предназначенная) для решения задачи π, которая не останавливается (например, не имеет зацикливаний) на всех входах задачи π. Будем говорить, что P *имеет дефект*, если для некоторого входа x задачи π имеет место $P(x) \neq \pi(x)$.

Определим *(эффективный) программный чекер* C_π для задачи π следующим образом. Чекер $C_\pi^P(I,k)$ — является произвольной вероятностной машиной Тьюринга, удовлетворяющей следующим условиям. Для любой программы P (предположительно решающей задачу π), выполняемой на всех входах задачи π, для любого элемента I задачи π и для любого положительного k (параметра безопасности) имеет место:

- если программа P не имеет дефектов, т. е. $P(x) = \pi(x)$ для всех входов x задачи π, тогда $C_\pi^P(I,k)$ выдаст на выходе ответ «Норма» с вероятностью не менее $1 - 1/2^k$;
- если программа P имеет дефекты, т. е. $P(x) \neq \pi(x)$ для всех входов x задачи π, тогда $C_\pi^P(I,k)$ выдаст на выходе ответ «Сбой» с вероятностью не менее $1 - 1/2^k$.

Самокорректирующаяся программа — это вероятностная программа C_f, которая помогает программе P скорректировать саму себя, если только P выдает корректный результат с низкой вероятностью ошибки. Данная оценка означает, что для любого x, C_f вызывает программу P для корректного вычисления $f(x)$, в то время как собственно сама P обладает низкой вероятностью ошибки.

Самотестирующейся/самокорректирующейся программной парой называется пара программ вида (T_f, C_f). Предположим, что пользователь может взять любую программу P, которая целенаправленно вычисляет f и тестирует саму себя при помощи программы T_f. Если P проходит такие тесты, тогда по любому x, пользователь может вызвать программу C_f, которая, в свою очередь, вызывает P для корректного вычисления $f(x)$. Даже если программа P, которая вычисляет значение функции f некорректно для некоторой небольшой доли входных значений, ее в данном случае все равно можно уверенно использовать для корректного вычисления $f(x)$ для любого x. Кроме того, если удастся в будущем написать программу P' для вычисления f, тогда некоторая пара (T_f, C_f) может использоваться для самотестирования и самокоррекции P' без какой-либо ее модификации. Таким образом, имеет смысл тратить определенное количество времени для разработки самотестирующейся/самокорректирующейся программной пары для прикладных вычислительных функций.

Перед тем как перейти к более формальному описанию определений самотестирующихся и самокорректирующихся программ необходимо дать определение вероятностной оракульной программе (по аналогии с вероятностной оракульной машиной Тьюринга). Вероятностная программа M является *вероятностной оракульной программой*, если она может вызывать другую программу, которая является исполнимой во время выполнения M. Обозначение M^A означает, что M может делать вызовы программы A.

Пусть P — программа, которая предположительно вычисляет функцию f. Пусть I является объединением подмножеств I_n, где $n \in \mathbf{N}$ и пусть $D^p = \{D_n \mid n \in N\}$ есть множество распределений вероятностей D_n над I_n. Далее, пусть $err(P, f, D_n)$ — это вероятность того, что $P(x) \neq f(x)$, где x выбрано случайно в соответствии с распределением D_n из подмножества I_n. Пусть β есть некоторый параметр безопасности. Тогда $(\varepsilon_1, \varepsilon_2)$-самотестирующейся программой для функции f в отношении D_p с параметрами $0 \leqslant \varepsilon_1 < \varepsilon_2 \leqslant 1$ называется вероятностная оракульная программа T_f, которая для параметра безопасности β и любой программы P на входе n имеет следующие свойства:

- если $err(P, f, D_n) \leqslant \varepsilon_1$, тогда программа T_f^P выдаст на выходе ответ «Норма» с вероятностью не менее $1 - \beta$;
- если $err(P, f, D_n) \geqslant \varepsilon_2$, тогда программа T_f^P выдаст на выходе «Сбой» с вероятностью не менее $1 - \beta$.

Оракульная программа C_f с параметром $0 \leqslant \varepsilon < 1$ называется *ε-самокорректирующейся программой* для функции f в отношении множества распределений D^p, которая имеет следующее свойство по входу n, $x \in I_n$ и β. Если $err(P, f, D_n) \leqslant \varepsilon$, тогда $C_f^P = f(x)$ с вероятностью не менее $1 - \beta$.

$(\varepsilon_1, \varepsilon_2, \varepsilon)$-самотестирующейся/самокорректирующейся программной парой для функции f называется пара вероятностных программ (T_f, C_f) такая, что существуют константы $0 \leqslant \varepsilon_1 \leqslant \varepsilon_2 \leqslant \varepsilon < 1$ и множество распределений D^p при которых T_f есть $(\varepsilon_1, \varepsilon_2)$-самотестирующаяся программа для функции f в отношении D^p и C_f есть ε-самокорректирующаяся программа для функции f в отношении распределения D^p.

Свойство случайной самосводимости. Пусть $x \in I_n$ и пусть $c > 1$ — целое число. Свойство случайной самосводимости заключается в том, что существует алгоритм A_1, работающий за время пропорциональное $n^{O(1)}$, посредством которого функция $f(x)$ может быть выражена через вычислимую функцию F от x, $a_1, \ldots, a_c$ и $f(a_1), \ldots, f(a_c)$ и алгоритм A_2, работающий за время пропорциональное $n^{O(1)}$, посредством которого по данному x можно вычислить $a_1, \ldots, a_c$, где каждое a_i является случайно распределенным над I_n в соответствии с D^p.

2.2 Устойчивость, линейная и единичная состоятельность

Пусть свойство I выражается уравнением $I(x_1, \ldots, x_k) = 0$, где кортеж $\langle x_1, \ldots, x_k \rangle$ выбирается с распределением E из пространства D_k. Пара (I, E) характеризует семейство функций F, где $f \in F$ тогда и только тогда, когда для всех $\langle x_1, \ldots, x_k \rangle$ с ненулевой выборкой элементов кортежа из E, $I^f(x_1, \ldots, x_k) = 0$. Базовой техникой самотестирования является идентификация свойства устойчивости для семейства функций F. Неформально (D, D')-устойчивость пары (I, E) для семейства функций G реализует, что если для программы $P \in G$, свойство $I^P(x_1, \ldots, x_k) = 0$ удовлетворяется с высокой вероятностью, когда $\langle x_1, \ldots, x_k \rangle$ выбрано с распределением E из D^k, тогда существует функция $g \in F \cap G$, которая согласуется с P на большей части входов из D'.

Рассмотрим некоторое свойство линейности (I, E), где свойство $I^f(x_1, x_2, x_3)$ тождественно $f(x_1) + f(x_2) = f(x_3)$ и E означает $(x_1 \in_{\mathrm{R}} Z_p, x_2 \in_{\mathrm{R}} Z_p, x_1 + x_2)$. Пара (I, E) характеризует $F = \{f(x) = cx \mid c \in Z_p\}$ — множество всех линейных функций над Z_p. В этом примере G — тривиальное множество всех функций и пара (I, E) устойчива для G.

Как только удалось убедиться посредством рандомизированных попыток, что программа R удовлетворяет свойству устойчивости, можно переходить к тестированию программы на линейную и единичную состоятельность.

Существует два базовых теста для самотестирующихся программ — *тест линейной состоятельности* и *тест единичной состоятельности* [8]. Продемонстрируем построение таких тестов на примере следующей тривиальной модулярной функции. Пусть x, R — положительные целые, тогда $f_R(x) \equiv x \pmod R$, где R фиксировано.

Пусть x_1 и x_2 случайно, равновероятно и независимо от других событий выбраны из Z_{R2^n} и x принимает значение: $x :\equiv x_1 + x_2 \pmod{R2^n}$. Необходимо отметить, что $f_R(x) \equiv [f_R(x_1) + f_R(x_2)] \equiv \pmod R$ — линейная функция по всем входам (аргументам). Тогда тест линейной состоятельности заключается в выполнении или не выполнении равенства: $P_R(x) \equiv [P_R(x_1) + P_R(x_2)] \pmod R$, а *ошибка линейной состоятельности* есть вероятность того, что данный тест не выполнился.

Пусть z случайно выбрано из Z_{R2^n} в соответствии с распределением и z принимает значение: $z' :\equiv z + 1 \pmod{R2^n}$. Отметим также, что $f_R(z') \equiv [f_R(z) + 1] \pmod R$. Тогда тест единичной состоятельности заключается в выполнении или не выполнении равенства: $P_R(z') \equiv [P_R(z) + 1] \pmod R$, а ошибка *единичной состоятельности* есть вероятность того, что данный тест не выполнился.

3 Прикладные результаты

3.1 Короткое замечание

В самом начале 90-х годов при создании библиотеки базовых криптографических функций «CRYPTOOLS 1.0» [11, 12] автор данной работы в составе коллектива разработчиков библиотеки, не сознавая того, использовал идеи самотестирования и самокоррекции (как само собой разумеющееся) для отладки кодов программ при вычисления теоретико-числовых функций, используемых в

интересах криптографии. А именно в это время и началось формирование методологии создания самотестирующихся и самокорректирующихся программ и их сочетаний [7, 8, 9, 10]. Позже, при создании более поздних версий библиотеки авторы имели, по существу, уже хороший математический аппарат, аргументирующий подобный процесс отладки [12].

3.2 Метод верификации расчетных программ на основе ST-пары функций

В качестве расчетной программы рассматривается любая программа, решающая задачу получения значения некоторой вычислимой функции. При этом под верификацией расчетной программы понимается процесс доказательства того, что программа будет получать на некотором входе истинные значения исследуемой функции. Иными словами, верификация расчетной программы направлена на доказательство отсутствия преднамеренных и (или) непреднамеренных программных дефектов в верифицируемой программе.

В данном случае предлагается метод создания самотестирующихся программ для верификации расчетных программных модулей [12]. Данный метод не требует вычисления эталонных значений и является независимым от языка программирования, используемого при написании расчетной программы, что существенно повышает оперативность исследования программы и точность оценки вероятности отсутствия в ней программных дефектов. Следует в то же время отметить, что предлагаемый метод можно использовать для программ, вычисляющих функции особого вида, а именно функции, обладающие свойством случайной самосводимости.

Пусть для функции $Y = f(X)$ существует пара функций $(g_c, h_c)^Y$ таких, что $Y = g_c(f(a_1), \ldots, f(a_c))$ и $X = h_c(a_1, \ldots, a_c)$.

Легко увидеть, что если значения a_i выбраны из I_n в соответствии с распределением D^p, тогда пара функций $(g_c, h_c)^Y$ обеспечивает выполнение для функции $Y = f(X)$ свойства случайной самосводимости. Пару функций $(g_c, h_c)^Y$ будем называть ST-парой функций для функции $Y = f(X)$.

Предположим, что на ST-пару функций можно наложить некоторую совокупность ограничений на сложность программной реализации и время выполнения. В этом случае, пусть длина кода программ, реализующих функции g_c и h_c , и время их выполнения составляет константный мультипликативный фактор от длины кода и времени выполнения программы P.

Предлагаемый метод верификации расчетной программы P на основе ST-пары функций для некоторого входного значения вектора X^* заключается в выполнении следующего алгоритма. Всюду далее, если осуществляется случайный выбор значений, этот выбор выполняется в соответствии с распределением вероятностей D^p.

Алгоритм ST

1. Определить множество $A^* = \{a_1^*, \ldots, a_c^*\}$ такое, что $X^* = h_c\{a_1^*, \ldots, a_c^*\}$, где $a_1^*, \ldots, a_c^*$ выбраны случайно из входного подмножества I_n.
2. Вызвать программу P для вычисления значения $Y_0^* = f(X^*)$.
3. Вызвать c раз программу P для вычисления множества значений $\{f(a_1^*), \ldots, f(a_c^*)\}$.
4. Определить значения $Y_1^* = g_c(f(a_1^*), \ldots, f(a_c^*))$.
5. Если $Y_0^* = Y_1^*$, то принимается решение, что программа P корректна на множестве значений входных параметров $\{X^*, a_1^*, \ldots, a_c^*\}$ в противном случае данная программа является некорректной.

Таким образом, данный метод не требует вычисления эталонных значений и за одну итерацию позволяет верифицировать корректность программы P на $(n+1)$ значении входных параметров. При этом время верификации можно оценить как $T = \sum_{i=1}^{c} t_i + t_x + t_g + t_{h^{-1}}$, где t_i и t_x — время выполнения программы P при входных значениях a_i, $i = 1, \ldots, c$ и X^* соответственно; t_g и $t_{h^{-1}}$ — время определения значения функции g_c и множества A^* соответственно: $T_P(X)$ — временная (не асимптотическая) сложность выполнения программы P; $K_{gh}(X, c)$ — коэффициент временной сложности программной реализации функции g_c и определения A^* по отношению ко временной сложности программы P (по предположению он составляет константный мультипликативный фактор от $T_P(X)$, а его значение

меньше 1). Для традиционного вышеуказанного метода тестирования время выполнения и сравнения полученного результата с эталонным значением составляет:

$$T_0 = \sum_{i=1}^{c} t_i + t_x + \sum_{i=1}^{c} t_i^e + t_x^e > 2T_P(X)(1+c),$$

где t_i^e и t_x^e — время определения эталонных значений функции $Y = f(X)$ при значениях a_i и X^* соответственно (в общем случае, не может быть меньше времени выполнения программы).

Следовательно, относительный выигрыш с точки зрения оперативности предложенного метода верификации (по отношению к методу тестирования программ на основе ее эталонных значений):

$$T_0 = \frac{T}{T_0} = \frac{\sum_{i=1}^{c} t_i + t_x + t_g + t_{h^{-1}}}{\sum_{i=1}^{c} t_i + t_x + \sum_{i=1}^{c} t_i^e + t_x^e} < \frac{1+c+K_{gh}}{2(1+c)} = 1/2 + \frac{K_{gh}}{2(1+c)}.$$

Так как коэффициент $K_{gh} < 1$, а $c \geqslant 2$, то получаем относительный выигрыш по оперативности испытания расчетных программ указанного типа (обладающих свойством случайной самосводимости) более чем в 1.5 раза.

3.3 Исследования процесса верификации расчетных программ

В качестве примера работоспособности предложенного метода рассмотрим верификацию программы вычисления функции дискретного возведения в степень: $y = f_{AM}(x) = A^x \bmod M$.

Выбор данной функции обусловлен тем, что она является одной из основных функций в различных теоретико-числовых конструкциях, например, в схемах электронной цифровой подписи и аутентификации, системах открытого распределения ключей и др. Отмеченный факт, в свою очередь, демонстрирует возможность применения предложенного метода при исследовании расчетных программ, решающих конкретные прикладные задачи. Кроме того, очевидно, что данная функция обладает свойством случайной самосводимости, а исходя из результатов работы [8] можно показать, что для данной функции существует $(1/288, 1/8)$-самотестирующаяся программа.

Для экспериментальных исследований была выбрана программа EXP из библиотеки базовых криптографических функций CRYPTOOLS [12], которая реализует функцию дискретного возведения в степень (размерность переменных и констант не превышает 128 байтов). Экспериментальные исследования состояли из определения временных характеристик процесса верификации на основе использования ST-пары функций и определения возможности обнаружения предложенным методом преднамеренно внесенных программных дефектов.

Для этого были определены следующие ST-пары функций:

$$\begin{aligned}
g_2(a_1, a_2) &= [f_{AM}(a_1) \cdot f_{AM}(1)] \pmod M, \\
h_2(a_1, a_2) &= a_1 + 1; \\
g_3^1(a_1, a_2, a_3) &= [f_{AM}(a_1) \cdot f_{AM}(a_2) \cdot f_{AM}(a_3)] \pmod M, \\
h_3^1(a_1, a_2, a_3) &= \sum_{i=1}^{3} a_i; \\
g_3^2(a_1, a_2, a_3) &= [f_{f_{AM(a_1)}}(a_2) \cdot f_{AM}(a_3)] \pmod M, \\
h_3^2(a_1, a_2, a_3) &= a_1 \cdot a_2 + a_3.
\end{aligned}$$

В процессе исследований менялась используемая ST-пара функций и варьировалась размерность параметров A, M и аргумента X. Результаты экспериментов полностью подтвердили приведенные выше временные зависимости (технические результаты исследований автор в данной работе опускает).

Исследование возможности обнаружения предложенным методом преднамеренно внесенных изменений заключалось в написании программы EXPZ. Спецификация для программ EXP и EXPZ одна и та же, отличие же заключается в том, что программа EXPZ содержит программную закладку деструктивного характера. Преднамеренно внесенная закладка при исследованиях гарантировала сбой работы программы вычисления значения функции $y = f_{AM}(x) = A^x \bmod M$ (то есть обеспечивала

получение неправильного значения функции) примерно на каждой восьмой части входных значений экспоненты x.

Все входные значения, на которых произошел сбой программы, были обнаружены, что в дальнейшем подтвердилось проверочными тестами, основанными на использовании малой теоремы Ферма и теореме Эйлера. Этот факт, в свою очередь, экспериментально показал, что программа, реализующая алгоритм ST, является $(1/8, 1/288)$-самотестирующейся программой.

Таким образом, предложенный метод позволяет в значительной степени сократить время испытания расчетных программ на предмет выявления непреднамеренных и преднамеренных программных дефектов. При этом по результатам испытаний можно получить экспериментальные оценки вероятности наличия программных дефектов в верифицируемой расчетной программе.

Более того, эксперимент сводился еще и к тому, чтобы разработать алгоритм SK, который позволял эффективно вычислять $y = A^x \bmod M$, не смотря на имеющиеся программные закладки. Таким образом, в итоге была получена $(1/8, 1/288, 1/8)$-самотестирующаяся/самокорректирующаяся программная пара для вычисления функции дискретного экспоненцирования.

4 Заключение

В итоге на некоем иллюстративном примере покажем фундаментальную сущность проактивно безопасной КС. Такие образные зарисовки на ранних этапах исследований являются более информативными, чем формальные языковые средства.

Попытаемся сравнить КС с живым организмом, а проактивно безопасную КС с генетически сконструированным живым организмом (моральную и социальную сторону проблемы мы опускаем). Вообще, использование медицинской терминологии при рассмотрении проблем информационной безопасности не является новым. Достаточно вспомнить компьютерную вирусологию и термины, используемые при этом: «вакцинация», «прививка», «инкубационный период», «вирусная эпидемия» и ряд других [3, 4].

Образные аналогии проактивно безопасных КС с генетически защищенными живыми организмами приведены в следующей таблице.

№	**Объекты анализа**	**Генетически защищенные организмы**	**Проактивно безопасные системы**
1	*Объекты*	Живые организмы	Компьютерные системы
2	*Создатели*	Отец, мать, прародители по отцовской и материнской линии (могут иметь генетические аномалии; при создании не имеют злоумышленных целей)	Разработчик, коллектив разработчиков (некоторые из них могут быть «нерадивыми» или иметь злоумышленные цели
3	*Свойства организма*	Физические и умственные свойства (сила, выносливость, долголетие интеллект, устойчивость к патологиям и болезням)	Качество, надежность,безопасность КС
4	*Концентрация на свойствах*	Устойчивости к патологиям, болезням, внешним травмам и др.	Защищенности информационных и функциональных ресурсов КС
5	*Субъекты нападения на этапе создания*	Врожденные генетические патологии (генетические дефекты обмена веществ, злокачественные новообразования и др.)	Априорные программные закладки, аппаратные закладные устройства
6	*Субъекты нападения на этапе жизни*	Внешние причинные факторы (ожоги, раны, обморожения и др.), вирусные инфекции	Внешние деструктивные воздействия, разрушающие программные средства (апостериорные программные закладки, компьютерные вирусы)
7	*Цель защиты*	Инициализация защитно-компенсаторных процессов организма	Инициализация процесса защиты информационных и функциональных ресурсов КС

8	*Основная задача защиты*	Целенаправленное включение новой информации в клетки высших организмов (эта информация должна быть направлена на защищенность организмов, а также на решение медико-биологических проблем, связанных с исправлением генетических дефектов обмена веществ, лечением вирусных заболеваний и злокачественных новообразований)	Целенаправленное включение средств защиты с хорошими пространственно-временными характеристиками в создаваемую КС, что позволяет оперативно «скорректировать» поведение КС в случае злоумышленного воздействия
9	*Инструментарий*	Генетическая (генная) инженерия — конструирование материального вещества наследственности, т. е. рекомбинантных ДНК (ДНК с заданным сочетанием генов)	Математический в т. ч. криптографический), инженерно-технический
10	*Методы защиты*	Нахождение векторных ДНК, в молекулы которых можно легко встроить отрезок, чужой ДНК или заменить фрагментом чужой ДНК, не нарушая при этом способности самой векторной ДНК к размножению (репликации) в клетке «хозяина»	Схемы, устойчивые к сбоям, (n, t)-пороговые схемы, проверяемые схемы разделения секрета, конфиденциальные вычисления, самокорректирующиеся среды
11	*Традиционные действия, обычный сценарий (реактивная безопасность)*	«Человек с кучей таблеток и вакцин, обвешанный медицинским оборудованием и инструментами»	«Навешивание на КС громоздких средств защиты, привлечение дополнительных системных функций, программно-аппаратных и “людских” ресурсов»
12	*Получаемый эффект*	Стабильное функционирование живого организма, несмотря на негативные внешние воздействия	Нормальное функционирование КС, которая «игнорирует» действие и последействие средств деструктивного характера

На основании изложенного, можно констатировать, что проактивно безопасные КС (если они будут созданы) выглядят исключительно привлекательно для пользователей КС, которые, по большому счету, могут «даже не заботиться» о защите своих информационных и функциональных ресурсов от кибератак. В данной работе предлагается лишь один из потенциально эффективных инструментов для создания подобных систем, — самокорректирующиеся программно-аппаратные среды. К тому же, как видно из данной работы и работ [3, 4], самокорректирующиеся программы уже сегодня имеют достаточно широкую прикладную сферу применения.

Литература

[1] Васенин В. А. *Проблемы математического, алгоритмического и программного обеспечения компьютерной безопасности в Интернет* // Математика и безопасность информационных технологий (МаБИТ-03). Материалы конференции в МГУ. М.: МЦНМО, 2004. С. 111–141.

[2] Казарин О. В. *Проактивная безопасность вычислительных систем* // Математика и безопасность информационных технологий (МаБИТ-04). Материалы конференции в МГУ. М.: МЦНМО, 2005. С. 306–320.

[3] Казарин О. В. *Безопасность программного обеспечения компьютерных систем.* М.: МГУЛ, 2003, 212 с.

[4] Казарин О. В. *Теория и практика защиты программ.* 2004, 450 с. `http://www.cryptography.ru`.

[5] Редькин Н. П. *Асимптотически минимальные самокорректирующиеся схемы для одной последовательности булевых функций* // Дискретный анализ исследование операций. Серия 1. 1996. Т. 3. № 2. С. 62–79.

[6] Редькин Н. П. *О самокорректирующихся схемах и о тестах для инверсных неисправностей элементов* // Материалы VIII Международного семинара «Дискретная математика и ее приложения», Изд-во механико-математического факультета МГУ, Москва, 2004. С. 4–8.

[7] Blum M., Kannan S. *Designing programs that check their work* // Proc. 21st ACM Symposium on Theory of Computing (STOC'89). P. 86–97.

[8] Blum M., Luby M., Rubinfield R. *Self-testing/correcting with applications to numerical problems* // Proc. 22nd ACM Symposium on Theory of Computing (STOC'90). P. 73–83.

[9] Gemmel P., Lipton R., Rubinfield R., Sudan M., Wigderson A. *Self-testing/correcting for polynomials and for approximate functions*// Proc. 23rd ACM Symposium on Theory of Computing (STOC'91). P. 32–42.

[10] Kumar R. S., Sivakumar D. *Efficient self-testing/self-correcting of linear recurrences* // Proceedings of the 37th IEEE Symposium on Foundations of Computer Science (FOCS'96). P. 602–611.

[11] *Библиотека базовых криптографических функций CRYPTOOLS* // Авторское свидетельство РосАПО N940518 от 16.12.94 г.

[12] Казарин О. В., Скиба В. Ю. *Об одном методе верификации расчетных программ* // Безопасность информационных технологий. 1997. № 3. С. 40–43.

Гарантированно защищенные базы данных, построенные на недоверенных с точки зрения безопасности элементах

А. А. Грушо, Е. Е. Тимонина

1 Введение

Построение надежной защиты информационных систем на недоверенной компьютерной базе является актуальной открытой задачей. В работе [3] нам удалось построить гарантированно защищенный интерфейс между защищенным сегментом сети и глобальной сетью при связи двух одноуровневых (в смысле политики MLS [2]) сегментов. При этом использовалась минимальная доверенная вычислительная база для нейтрализации программно-аппаратных агентов нарушителя безопасности в защищенных сегментах, оборудовании и программном обеспечении интерфейса и в глобальной сети.

В данной статье рассматривается задача построения модели гарантировано защищенной базы данных. В наших условиях считаем, что не все пользователи являются доверенными лицами и поступающая к ним информация на запросы других пользователей может считаться утечкой информации, а производимые ими самими действия являются легальными и невраждебными(из-за опасности быть обнаруженными при анализе аудита использования базы данных). Однако возможности враждебных пользователей становятся большими при использовании враждебных программно-аппаратных агентов в компьютерной среде, базах данных и коммуникационном оборудовании. Политика разграничения доступа к базе данных представима в виде ролевой политики безопасности, при которой роль связана с рабочей станцией клиента (хотя возможны другие правила политики безопасности, включая реализацию многоуровневой политики безопасности [2]). Механизмы разграничения доступа реализованы криптографическими методами. В частности пользователь может получить только ту информацию, для которой имеет ключ к расшифрованию ответа на свой запрос. Связь пользователя с базой данных может проходить через незащищенную территорию, например через Интернет. Таким образом противник может находиться на одной из рабочих станций, но интересоваться не только данными относящимися к своей роли. Противник может также находиться в сети и при сговоре с легальным пользователем- нарушителем безопасности подслушивать запросы и взаимодействовать с программно-аппаратными агентами в доступной для скрытых каналов среде базы данных. Так как пользователей базы данных может быть много, то описанная в [3] гарантированная защита связи в случае с единой базой и многих пользователей не защищает от скрытых каналов, например, от каналов с модуляцией потоков адресов [4].

В рассматриваемой задаче предполагается, что в программно-аппаратных платформах и СУБД существуют программно-аппаратные агенты противника, «невидимые» для штатных средств защиты. Для выполнения задания нарушителя безопасности программно-аппаратные агенты нарушителя безопасности должны получать инструкции извне системы, должны координировать свои действия и должны передавать ценную или конфиденциальную информацию нарушителю вне системы. Поскольку связь между станциями и серверами системы может эффективно контролироваться, то взаимодействие агентов между собой, а также взаимодействие с внешней средой может происходить только с помощью скрытых каналов.

Используя программно-аппаратных агентов и скрытые каналы для связи с противником в глобальной сети, опираясь при этом на возможности легального пользователя-нарушителя безопасности, противник старается решать свои задачи. При работе с базами данных противник может ставить следующие задачи:

Работа поддержана грантом РФФИ, проект 04-01-00089.

- нахождение нужной ценной информации;
- передача ценной информации противнику в глобальной сети;
- имитация себя как легального пользователя (отличного от нарушителя);
- модификация данных;
- уничтожение информационного ресурса.

Мы также считаем, что в наших силах построить доверенную программно-аппаратную среду с очень ограниченным функционалом, в которой выполнение этого функционала будет доверенным.

2 Математическая модель защиты баз данных

База данных (БД) состоит из таблиц, называемых отношениями [1]. Строки таблиц содержат значения базы данных, а столбцы называются атрибутами или полями. Уникальный идентификатор строк таблицы называется первичным ключом. Для любого данного атрибута определяется область его значений. Домен — множество допустимых значений атрибутов.

Контроль целостности в таблице позволяет, например, проводить проверку однозначной идентифицируемости строк по ключу и принадлежность домену.

В данной статье мы рассматриваем упрощенную модель базы данных, состоящую из одной или двух таблиц и операций select, update, insert, delete, причем последние две операции легко доопределяются, если определены первые две.

Приведем формальное определение операции select.

Select(R, F) — строит новое отношение (таблицу), состоящее из всех векторов (строк) R, удовлетворяющих F, где F — формула вида « $A_i \Theta V$» или «$A_i \Theta A_j$», где Θ — отношение сравнения ($=, \leqslant$ и т. д.) и V — значение из области D_i атрибута A_i (Этот оператор также называют «Θ — выбор» или выборка отношения).

Распределение информации в любом осмысленном сообщении (в любом языке) подчиняется следующему лингвистическому закону [Пиотровский, («Текст, человек, машина», М.: 1972).]. Согласно этому закону любое осмысленное сообщение структурно разбивается на две непересекающиеся части: тема и рема. Тема определяет контекст информационного сообщения и отвечает на вопрос, о каком типе ремы идет речь. Рема содержит значения отличительной информационной составляющей в данном контексте и отвечает на вопрос: что нового получит получатель сообщения в данной теме (в данном контексте). Известно, что без специальных методов сжатия тема в естественном языке занимает 80 процентов места, а ее информационная нагрузка составляет 20 процентов. Соответственно, рема занимает около 20 процентов места, но ее информационная составляющая в сообщении составляет 80 процентов. При этом тема без ремы не несет никакой информации (кроме случаев вопросов). Точно также рема без темы (без заданного контекста) не несет никакой информации. Поэтому в отдельности значения темы и значения ремы почти не несут самостоятельной информации.

Данное утверждение можно проиллюстрировать следующим образом. Пусть выписан словарь слов из некоторого литературного произведения. По содержанию этого словаря можно иногда оценить приблизительно тему произведения, но нельзя ничего сказать о содержании произведения (то есть фактах, идеях и мыслях, которые в нем заложены).

В базах данных содержательными информационными элементами являются строки и множества строк. Можно считать, что строка есть содержание сообщения, которое передается пользователю по запросу select. Можно сказать, что строка определена набором названий атрибутов и это тема для каждой строки, а набор значений атрибутов — это рема строки. Поэтому в строке присутствуют тема и рема, несущие информацию, полезную для пользователя, причем эта информация представлена в виде значений атрибутов и строчной связи (названия атрибутов в строке), которая состоит в том, что данные элементы составляют сроку базы данных. На основании лингвистического закона, приведенного выше, мы можем сформулировать следующую аксиому, которая лежит в основе построения защиты базы данных.

Аксиома. В каждом сообщении (строке базы данных) существуют тема и рема и они не пересекаются в том смысле, что одной теме могут соответствовать много рем, а одна рема имеет смысл только в сочетании с некоторой темой(при этом сама рема тему не задает).

При этом строчная связь (перечень названий атрибутов в строке) представляет собой тему сообщения, а ремой сообщения чаще всего являются конкретные значения атрибутов в данной строке. Аксиома утверждает, что связей между значениями в строках БД вне темы не существует.

Следуя этой лингвистической логике, мы можем сказать, что значения БД, т. е. значения ремы, взятые отдельно без контекста, не дают информации пользователю базы данных. Точно также как строка без заполнения элементами ремы не несет никакой информации.

Рассмотрим базу данных, состоящую из одного отношения (таблицы):

$$R(A_1,\dots,A_r)=\begin{pmatrix} A_1 & . & . & . & A_r \\ a_{i1} & . & a_{ij} & . & a_{ir} \\ . & . & . & . & . \end{pmatrix} \tag{1}$$

В этом отношении A_1 считаем ключевым атрибутом. Исходя из этого отношения, построим следующее семейство отношений (таблиц):

$$\begin{gathered} R(B_1,A_1)=\begin{pmatrix} B_1 & A_1 \\ b_{i1} & a_{i1} \\ . & . \end{pmatrix}, \\ \dots \\ R(B_r,A_r)=\begin{pmatrix} B_r & A_r \\ b_{ir} & a_{ir} \\ . & . \end{pmatrix} \end{gathered} \tag{2}$$

В этих таблицах b_{ij} — это независимые случайные или псевдослучайные последовательности достаточной длины, чтобы говорить об уникальности их значений. В каждой таблице атрибут B является ключом. Отношения $R(B_i,A_i), i=1,\dots,r$, могут быть расширены «шумом». То есть могут быть добавлены любые не встречающиеся в исходном отношении $R(A_1,\dots,A_r)$ значения атрибутов A в сочетании с уникальными значениями атрибута B.

С помощью «шума» можно добиться любой точности приближения к аксиоме в том смысле, что «шум» снижает информационную связь между значениями различных атрибутов вне строк БД.

Рассмотрим таблицу:

$$R(B_1,\dots,B_r)=\begin{pmatrix} B_1 & . & . & . & B_r \\ b_{i1} & . & b_{ij} & . & b_{ir} \\ . & . & . & . & . \end{pmatrix} \tag{3}$$

В дальнейшем нам потребуется следующие леммы.

Лемма 1. *Не существует алгоритма, восстанавливающего отношение $R(A_1,\dots,A_r)$ по (2).*

Доказательство. Для доказательства достаточно показать, что не может быть восстановлена строка отношения $R(A_1,\dots,A_r)$ исходя из знания $R(B_i,A_i), i=1,\dots,r$. Из противного следует, что существует строка $(a_{i1},\dots,a_{ir})$, которая может быть восстановлена из (2). При этом значения атрибутов b_{ij} являются независимыми случайными числами и не могут нести информации о данной строке. Тогда строка $(a_{i1},\dots,a_{ir})$ восстанавливается за счет связей значений отдельных значений $a_{i1},\dots,a_{ir}$, то есть только ремы. Значит это и тема и рема одновременно, что противоречит аксиоме. Лемма доказана. □

Лемма 2. *1. Существует алгоритм, восстанавливающий отношение $R(A_1,\dots,A_r)$ по (2) и (3) (даже при наличии «шума»).*

2. Алгоритм, восстанавливающий $R(A_1,\dots,A_r)$ по расширенным с помощью «шума» (2) и (3), инвариантен относительно любых изменений значений атрибутов $A_1,\dots,A_r$.

Доказательство. Для доказательства п. 1 построим алгоритм решения этой задачи. Рассмотрим произвольную строку таблицы (3) $(b_{i1},\dots,b_{ir})$. Выберем в таблице $R(B_1,A_1)$ значение первого атрибута i-ой строки a_{i1}. Соответственно в таблице $R(B_j,A_j)$ выберем b_{ij} и по нему соответствующее значение a_{ij}. Данная строка и все остальные восстанавливаются однозначно. Строки с «шумом» просто не учитываются в этом процессе, так как они отсутствуют в (3).

Утверждение п. 2 следует из описания приведенного алгоритма. Лемма доказана. □

Из определения операции select в базах данных следует, что операцию select можно реализовать в два этапа с использованием таблиц (2) и (3).

1-й этап. Выбираем в (2) значения атрибутов, удовлетворяющих формуле $A\Theta V$ или формуле $A_i\Theta A_j$. Для выделенных значений определяется множество значений атрибутов B_i и пар (B_i, B_j).

2-й этап. По выбранным значениям атрибутов B_i и пар (B_i, B_j) восстанавливаются строки в отношении (3). По этим строкам согласно лемме 2 однозначно восстанавливаются строки отношения $R(A_1, \ldots, A_r)$, удовлетворяющие данному запросу select.

Эти рассуждения можно суммировать в виде леммы 3.

Лемма 3. *Существует алгоритм выполнения операции select по* (2) *и* (3) *без восстановления отношения* $R(A_1, \ldots, A_r)$.

Рассмотрим оператор update. Его можно реализовать следующим образом. Пусть необходимо заменить существующее значение в поле атрибута A_j в строке с ключевым значением атрибута a_{i1} на значение a. Выполнение этой операции в исходной таблице $R(A_1, \ldots, A_r)$ следующее. Выбирается строка в $R(A_1, \ldots, A_r)$ со значением ключевого атрибута a_{i1}. В этой строке однозначно определяется значение j-го атрибута. После этого значение j-го атрибута заменяется на a.

Лемма 4. *Существует алгоритм выполнения операции update по* (2) *и* (3) *без восстановления отношения* $R(A_1, \ldots, A_r)$.

Доказательство. Пусть необходимо заменить существующее значение в поле атрибута A_j в строке с ключевым значением атрибута a_{i1} на значение a. С помощью (2) по значению ключевого атрибута a_{i1} восстанавливается значение ключевого атрибута b_{i1}. Тогда в (3) находится строка с ключевым значением b_{i1}, в которой берется значение b_{ij}. Затем в (2) в таблице $R(B_jA_j)$ находится значение ключевого атрибута b_{ij}, и в этой строке значение атрибута a_{ij}заменяется на значение a. Лемма доказана. □

В основе построения защищенной базы данных лежит возможность раздельного выполнения операций select, update с использованием (2) и (3), расположенных на различных узлах распределенной вычислительной системы. Причем согласно лемме 1 знание одной из баз (2) или (3) не достаточно для восстановления какой-либо строки $R(A_1, \ldots, A_r)$. Но согласно леммам 2, 3, 4 совместное использование этих баз данных однозначно определяет $R(A_1, \ldots, A_r)$.

Теорема. *Для исходной базы данных* $R(A_1, \ldots, A_r)$ *могут быть построены распределенные базы данных* (2) *и* (3), *которые позволяют выполнять операции select и update без восстановления базы данных* $R(A_1, \ldots, A_r)$, *причем каждая из баз* (2) *или* (3) *в отдельности не несет никакой информации о базе данных* $R(A_1, \ldots, A_r)$.

Доказательство. Из леммы 3 и 4 следует, что select и update могут быть выполнены без восстановления базы данных $R(A_1, \ldots, A_r)$. Из леммы 1 следует, что в (2) или в (3) по отдельности нет достаточной информации для восстановления $R(A_1, \ldots, A_r)$. Лемма 2 утверждает, что «шум» не мешает этим операциям. Теорема доказана. □

Из вышесказанного следует, что в реляционной базе данных информация сконцентрирована в значениях базы данных и связях между этими значениями (строчные связи, межтабличные связи). В наших условиях будем считать, что основная информация содержится в строчных и межтабличных связях и в меньшей степени в самих значения атрибутов. То есть значения любого атрибута (рема) без привязки к другим значениям атрибутов в данной строке не является конфиденциальной информацией. За счет «шума» в (2) можно немного отступить от этого постулата. Необходимо полностью исключить ситуацию, когда существует хотя бы одно значение какого-либо атрибута, несущее самостоятельно ценную информацию.

В защищенном исполнении база данных должна разрешать операции select, update, insert, delete.

Выполнение аксиомы на практике означает, что трудоемкость восстановления информативных связей между значениями атрибутов (например, восстановление строки по известным множествам значений атрибутов в столбцах) недопустимо высока для вычислительных возможностей противника.

3 Архитектурное решение

Любую операцию select можно разбить на две части. Первая связана с выбором значений, удовлетворяющих запросу пользователя, вторая — с выбором строк, в которых находятся выбранные значения. Обязательность первой части определяет невозможность шифрования баз данных и связана с тем, что нарушаются возможности эффективного поиска информации и другого использования этих баз данных. Поэтому схемы защиты, связанные с шифрованием данных, оказываются не эффективными. Вместе с тем, в реляционной базе данных значение каждого атрибута может сопровождаться ключом, который представляет собой криптографически стойкую псевдослучайную последовательность. Тогда основные операции по поиску данных, пополнения и модификации баз данных можно разделить на две основные части. Первая часть — выбор записей, которые удовлетворяют запросам среди множества значений таблиц, каждая из которых является столбцом значений одного атрибута. Вторая часть — это стыковка выделенных записей на основе значений псевдослучайных ключей этих таблиц. Эти вычисления можно сделать разделенными и разнесенными в распределенной базе данных, использующей различные экземпляры СУБД. В результате формируется множество строк, удовлетворяющих запросу пользователя. Из этого множества можно сделать обзор и в зашифрованном виде передать легальному пользователю.

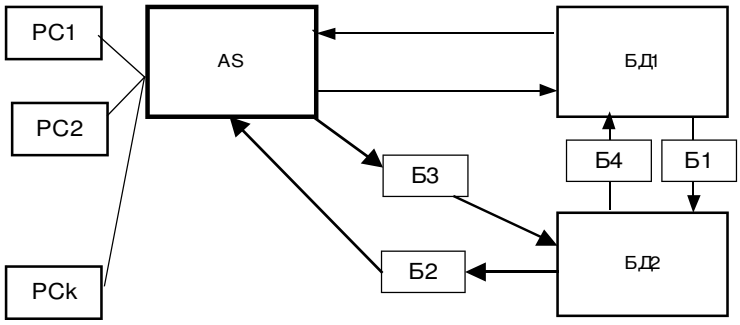


Рис. 1: Архитектурное решение

На схеме, изображенной на рис. 1, все связи пользователей с базой данных реализуются через сервер приложений AS. На сервере приложений происходит аутентификация пользователей штатными средствами защиты и разграничение запросов пользователей с точки зрения защиты запросов одних пользователей от других пользователей. Для связи пользователей с AS может использоваться криптография, например SSL. Однако мы предполагаем, что программно-аппаратные агенты противника могут присутствовать в AS. Из этого следуют следующие выводы. С помощью скрытых каналов противник в глобальной сети может узнать информацию о ключах SSL и пароли любых пользователей. Если применяются протоколы рукопожатий, то агент может вставить любой запрос в любую сессию от имени легального пользователя этой сессии. Таким образом, противник знает все на сервере приложений, а запросы пользователей нельзя считать защищенными от такого противника. Однако SSL является хорошей защитой от более «простых» противников.

Запросы типа select из AS поступают в БД1, где выделяются все значения атрибутов, требуемых в запросе, а также пары значений различных атрибутов, равные между собой, тройки и т. д. Данные в БД1 хранятся в виде, описанным в форме (2). Между AS и БД1 существует незащищенная двусторонняя скоростная связь. Поэтому агент нарушителя безопасности в БД1 имеет связь с агентом в AS. Отсюда следует, что все таблицы БД1 известны противнику в глобальной сети, хотя с некоторой задержкой, связанной с ограниченной пропускной способностью скрытых каналов. Однако, как было отмечено выше, по значениям атрибутов противник без знания строчных связей не может получить содержательной информации, особенно при наличии «шума». Противник может пытаться модифицировать какие-либо значения атрибутов. Однако далее мы покажем, что построенная система способна выявлять модификации и исправлять их не в реальном времени. Правда, нелегальные изменения значений атрибутов могут негативно сказаться на результативности поиска в БД. Эта проблема, а также проблема инициации сбоев с помощью агентов в данной работе не рассматриваются.

Выбранные значения в БД1, как строчки соответствующих таблиц, передаются в виде последовательности байт в БД2 через узел защиты Б1. Данный узел осуществляет линейное преобразование

всех байт данных с помощью невырожденной матрицы M_1, а также вычисляются коды аутентификации с помощью матрицы M_2 как это было предложено в [3]. В базе данных 2 (БД2) осуществляется стыковка выбранных с помощью операции select значений таким образом, чтобы восстановить строчки преобразованных записей исходной таблицы $R(A_1, \ldots, A_r)$ и осуществить поиск по межтабличным связям в условиях, когда все данные представлены в преобразованном виде, а восстановление строк происходит с помощью специальных ключевых атрибутов B, сопровождающих каждое выбранное значение. Таким образом в результате обработки в БД2 возникают все строки, выделенные по значениям атрибутов в БД1 на основании SQL запросов пользователя, но в преобразованном виде.

Полученные данные отделяются от вспомогательных атрибутов и передаются в AS для передачи пользователю через узел Б2 проверки и снятия кода аутентификации, обратного преобразования M_1^{-1} и шифрования. Таким образом, у пользователя появляются все восстановленные строки исходных таблиц, значения которых удовлетворяют запросу select данного пользователя при условии, что он знает ключ шифрования.

При шифровании можно использовать два ключа: индивидуальный ключ пользователя и ключ, соответствующий категории ценности строки. Тогда чтение возможно при наличии у пользователя соответствующих полномочий.

Необходимые действия над восстановленными данными в БД3 формируются в БД2 с помощью метода шаблонов. Отметим, что истинные данные, отвечающие строчным межтабличным связям, возникают у пользователя в объеме, требуемом для удовлетворения запроса пользователя.

При получении сервером приложений легального запроса на модификацию данных update через БД1 проходит запрос на выбор ключевого атрибута и номера атрибута, который подлежит модификации, и его значения. В БД2 происходит восстановление строки с заданным значением ключевого атрибута и значения атрибута, который нужно модифицировать. Значение модифицированного атрибута поступает в БД2 из сервера приложений через отдельный канал и узел защиты Б3, в котором происходит расшифрование этого значения на ключе полномочного пользователя. В связи с тем, что в БД2 восстанавливается преобразованная с помощью M_1 и M_2 строка из исходной базы данных и получено новое значение исправляемого атрибута также в преобразованном виде по M_1 и M_2 (через узел Б3) такое исправление становится возможным. Уведомление об исправлении направляется в шифрованном виде пользователю через Б2, а исправленная строка в БД2 разлагается на отдельные значения, к которым приписываются псевдослучайные атрибуты, и затем поступает в БД1 через Б4 (обратные преобразования для M_1 и снятие кода аутентификации) в той форме, в которой хранятся данные в этой базе. Операции insert, delete осуществляются по аналогичной схеме.

4 Представление данных в БД1

Хранение данных в БД1 осуществляется в форме таблиц (2), имеющих 2 атрибута. Пусть исходная база данных содержит таблицы T1 и T2, атрибуты таблицы T1 — $A_1, \ldots, A_r$, где A_1 — ключевой атрибут; таблица T2 имеет атрибуты $C_1, \ldots, C_m$, где C_1 — ключевой атрибут. Таблицы БД1 строятся в БД2 по следующему правилу. Для каждого атрибута A_i вводится дополнительный атрибут B_i и для каждого атрибута C_i вводится атрибут D_i, в результате чего формируется таблицы (B_i, A_i) и (D_i, C_i). Значения атрибутов B_i и D_i являются уникальной последовательностью с характеристиками случайного числа, а значения атрибутов A_i и C_i остаются в БД1 в открытом виде. Таким образом если в SQL запросе присутствуют значения атрибутов $A_{i1}, \ldots, A_{ik}$ и $C_{j1}, \ldots, C_{js}$, то в таблицах выбираются соответствующие истинные значения атрибутов, присутствующие в SQL запросе, а также названия соответствующих атрибутов в явном виде. Если существуют межтабличные связи, указывающие на выбор значений атрибута C_i при выборе атрибута A_i, заданные в явном виде в SQL запросе, то такая операция также осуществляется в БД1. Далее выбранные значения (вместе со значениями атрибутов B и D) поступают в БД2 через узел защиты, в котором каждый байт данных преобразуется по линейному преобразованию M_1 и добавляется код аутентификации по M_2.

5 Описание операций в БД2

В БД2 хранятся представления таблиц T1 и T2, выраженные (см. (3)) через значения атрибутов B_i и D_i. Таким образом по значениям выбранных атрибутов B и D восстанавливаются преобразованные

строчки таблиц T1 и T2. Вместе с этими строками воссоздаются преобразованные по линейному преобразованию M_1 истинные строки исходных таблиц, выбранных в соответствии с SQL запросом. В восстановленных строках исключаются значения атрибутов B и D, после чего полученные данные передаются в узел защиты Б2. Узел защиты Б2 снимает со всех байт преобразования M_1 и M_2 с помощью сложения с кодом аутентификации и умножения байта на матрицу M_1^{-1} , в результате чего в доверенной среде возникают истинные строки исходных таблиц, выбранных в соответствии с SQL запросом пользователя. Эти строки шифруются и передаются на сервер приложений, а оттуда пользователю, осуществившему исходный запрос.

Рассмотрим генерацию атрибутов B и D. Пусть выполняется операция создания строки $(a_1, \ldots, a_r)$ с атрибутами $A_1, \ldots, A_r$ в системе. В БД2 эта строка появляется в преобразованном виде. Предположим, что имеется массив случайных и равновероятных байт. Для каждого значения атрибута $(M_1, M_2)(a_i)$ берется очередной кусок последовательности байт, длина которого равна длине записи преобразованного по M_1 значения атрибута с кодами аутентификации плюс фиксированное число m байт. Значение преобразованного атрибута вместе с кодами аутентификации шифруется с помощью этой последовательности имитозащищенным шифром. При этом первые m байт не участвуют в шифровании, а приписываются к шифртексту в явном виде. Эти m байт вместе с шифртекстом образуют последовательность x. Эта последовательность подвергается преобразованию, состоящему в вычислении и приписывании каждому байту кода аутентификации. Полученная последовательность образует преобразованное по (M_1, M_2) значение атрибута B_i для значения a_i. Данная процедура позволяет по значению атрибута B (по первым m байтам) отыскивать ключ для шифрования значения атрибута A, который в явном виде может храниться в БД2, вычислять преобразованное значение атрибута A и сравнивать его с преобразованным значением атрибута A для контроля целостности. В случае совпадения расшифрованное значение может быть передано Б2. Операция update осуществляется также с разницей, что вместо старого значения атрибута A и его ключа B в строку записывается новое значение ключа.

Одной из опасностей частичного восстановления данных с передачей этой информации противнику является привязка ключевых значений в базе данных БД1 при модификации отдельного значения атрибута. В этом случае в SQL запросе фигурирует значение ключа для модифицируемого значения, поэтому появление измененного значения в БД1 может быть однозначно связанно с данным значением ключа. Это позволит противнику, получившему через агента право модифицировать данные, восстанавливать строчные связи. Чтобы избежать этой утечки предлагается использовать «шум», то есть формировать как минимум одно ложное значение атрибута после модификации. Это значение в таблице БД1 будет иметь свой ключ и может не совпадать с истинным значением модифицированного атрибута. Появляется неоднозначность привязки значений ключевого атрибута к значениям модифицированного атрибута. Чтобы приращение количества ложных значений в каждой таблице не увеличивалось постоянно, предлагается уничтожать вместе выполняемой операцией delete соответствующее ложное значение атрибута. Введение достаточного количества ложных значений в таблице БД1 снижает информативность истинных значений атрибутов при получения информации о значениях в базах данных противником.

6 Обоснование безопасности

Сервер приложений связан с БД1 двухсторонней связью, поэтому возможности взаимодействия агентов в сервере приложений и агентов в БД1 будем считать не лимитированными. Агент в сервере приложений связан с различными пользователями, поэтому согласно [4] агенты в сервере приложений имеют связь с нарушителями безопасности, несмотря на шифрование и межсетевые экраны. Между БД1 и БД2 имеет двухсторонняя связь, при этом данные поступающие из БД1 преобразовываются по секретной матрице M_1 и вычисляются коды аутентификации. Все данные, поступающие из БД2 в БД1, освобождаются от преобразования M_1. Таким образом все данные в БД2 обрабатываются в преобразованном с помощью M_1 виде. Однако для всех преобразований данных в БД2 требуются знания только атрибутов B и D и может быть равенство значений атрибутов A или C при контроле целостности. Все эти функции могут быть выполнены без знания преобразования M_1. Таким образом программно-аппаратный агент в БД2 не может воспользоваться методами стеганографии при передаче данных из БД2 в БД1 и из БД1 в БД2. В случае, когда агент попробует передавать все подряд, коды аутентификации, как и раньше, предотвратят утечку. Остальные скрытые каналы перекрываются с

помощью специального интерфейса между БД1 и БД2. Таким образом в БД2 обрабатываются все строчные связи, при этом агент в БД2 не имеет директив по анализу этих связей и не имеет возможностей осуществлять поиск по этим связям. Истинные значения атрибутов известны в БД1, но агент в БД1 не знает строчных и межтабличных связей этих значений. Передача инструкций агента от БД1 в БД2 и наоборот перекрыта преобразованием M_1, которое не может быть восстановлено с помощью решения линейных систем уравнений в БД2 в силу их отсутствия, но может быть восстановлена с помощью эвристики, которая заведомо отсутствует у этого агента и не может быть передана от агента в глобальной системе через агента на сервере и агента в БД1, так как агент в БД2 не знает M_1. Данные в явном виде возникают только в Б2, при этом агентов в Б2 не может быть. На сервер приложений поступают шифрованные данные и далее, так как они шифруются на ключе пользователя, противник не имеет доступа к информации. Аналогично с помощью специальных интерфейсов исключаются скрытые каналы из БД2 в сервер приложений. Канал из сервера приложений в БД2, через которые передаются значения модифицированных атрибутов, является шифрованным на ключах пользователя и роли. Расшифрование этих данных осуществляется на специальном узле Б3, расположенном на канале от сервера приложений в БД2. С помощью метода создания специального интерфейса скрытые каналы от сервера приложений к БД2 также уничтожаются. Таким образом, закладка в сервере приложений не может взаимодействовать с закладкой в БД2. Отсюда следует, что, несмотря на наличие программно-аппаратных агентов во всех БД и сервере приложений, восстановление и передача этих данных нарушителю в глобальной сети невозможна.

Литература

[1] Глушаков С. В., Ломотько Д. В. Базы данных: Учебный курс. Харьков: Фолио; М.: ООО «Издательство АСТ», 2001. 504 с.

[2] Грушо А. А., Тимонина Е. Е. Теоретические основы защиты информации. М.: Агентство «Яхтсмен», 1996. с. 494.

[3] Грушо А. А., Володин А. В., Тимонина Е. Е. Безопасный интерфейс с глобальной сетью из ненадежных в смысле безопасности элементов. Труды XXVIII международной конференции «Информационные технологии в науке, образовании, телекоммуникации, бизнесе. IT+SE'2001 (майская сессия)», Ялта — Гурзуф, Украина, 20–29 мая 2001 г.

[4] Тимонина Е. Е. Анализ угроз скрытых каналов и методы построения гарантированно защищенных распределенных автоматизированных систем. М.: Диссертация на соискание ученой степени доктора технических наук, 2004.

К развитию механизмов разграничения доступа в распределенных информационных системах

А. А. Иткес, В. Б. Савкин

Введение

В течение последних лет сфера применения сложных, территориально распределенных информационных систем неуклонно расширяется. Одновременно с этим возрастает и величина потенциального ущерба, который может быть вызван сбоями в их работе. Такие сбои могут возникать как случайно, так и в результате злонамеренных действий посторонних лиц, либо неправильных действий работников, ответственных за те или иные аспекты сопровождения системы. Диапазон типов потенциальных нарушителей с точки зрения их целей и возможностей простирается от подростков, ищущих приключений, до террористических организаций, имеющих целью на длительное время вывести из строя жизненно важные для страны информационные системы, причинив как можно больший ущерб. Отказы или сбои в работе таких систем систем могут привести к экологическим или техногенным катастрофам, человеческим жертвам и, учитывая тенденции развития общества и темпы развития телекоммуникаций, в будущем риски подобных событий увеличатся.

Специалисты, анализирующие методы защиты сложных информационных систем, часто говорят о «правиле слабого звена». Оно заключается в том, что система защищена настолько надежно, насколько можно гарантировать защиту самой слабой её компоненты. Во многом указанное утверждение справедливо, однако, не следует также пренебрегать защитой системы в целом в том случае, если часть её оказывается под контролем злоумышленников. Существует несколько причин для возникновения подобной постановки задачи. Первая из них состоит в том, что как бы хорошо ни были защищены компоненты распределенной информационной системы от внешней угрозы, злоумышленник может получить контроль над одной из них в каком-то смысле «случайно». Вторая причина заключается в том, что зачастую организации сталкиваются с нападениями не только внешних, но и внутренних нарушителей. Не следует также упускать из вида «комбинированные» угрозы, исходящие как извне, так и изнутри организации. Если противник — террористическая организация или крупная преступная группировка, то вполне вероятно, что перед нападением злоумышленники попытаются подкупить кого-либо из служащих организации, имеющей прямое отношение к процессу эксплуатации системы. Третья причина, по которой необходимо разрабатывать меры безопасности, предназначенные для функционирования в том числе и в условиях, когда некоторые из компонент системы контролируются злоумышленником, состоит том, что многие крупные распределенные информационные системы создаются путём объединения более мелких систем. До объединения эти системы могут иметь различные уровни защищенности, и некоторые из систем, по крайней мере на начальном этапе их совместной работы, могут быть защищены недостаточно надежно.

С учетом изложенного, представляет практический интерес разработка способов построения политик безопасности сложных, территориально распределенных информационных систем таким образом, чтобы, с одной стороны, минимизировать взаимозависимость компонент такой системы и уменьшить возможный ущерб от несанкционированного доступа к некоторым из них, и, с другой стороны, оставить компоненты системы достаточно сильно связанными для того, чтобы система могла успешно выполнять свои основные функции. Необходимо также уделить внимание защите гетерогенных систем, отдельные компоненты которых имеют различные, никаким образом не согласованные друг с другом политики безопасности.

Предлагаемая работа состоит из двух частей. Первая часть описывает один из подходов к формализации понятия доверия между узлами распределенной информационной системы. В этой части предлагается метод построения политики безопасности распределенной системы по имеющимся политикам безопасности ее отдельных компонент и оценивается эффективность предложенного метода.

Вторая часть описывает ведущиеся в настоящее время работы по созданию программных средств, позволяющих на уровне ядра операционной системы задавать и контролировать выполнение политики безопасности для распределенной информационной системы. Здесь и далее под политикой безопасности понимается только описание разрешенных доступов к ресурсам, и не затрагиваются такие аспекты безопасности, как аутентификация субъектов, шифрование, аудит и ряд других. Такая политика может быть основана на одной из известных моделей логического разграничения доступа. Принимая во внимание упомянутые ограничения, будем под термином «ролевая политика безопасности», например, подразумевать политику, при которой управление доступом происходит на основе ролевой модели логического разграничения доступа.

1 Отношения доверия

В данном разделе рассмотрен один из способов формализации понятия доверия одного узла распределенной информационной системы к другому. Центральным понятием в приведенных рассуждениях является отношение доверия.

Прежде, чем определить отношения доверия, необходимо ввести обозначения. Если A представляет собой информационную систему, то $O(A)$ обозначает множество объектов системы A, а $S(A)$ — множество субъектов системы A.

Определение 1. Отношением доверия между информационными системами A и B называется множество $T_{A,B} \subset S(A) \times S(B)$. При этом, если $(S_A, S_B) \in T_{A,B}$ будем говорить, что S_B доверяет S_A. Смысл данного отношения в том, что если субъект S_B доверяет субъекту S_A, то S_A может получить доступ к объектам системы B посредством S_B.

Одним из наиболее показательных примеров отношения доверия является доверие между WEB-клиентом и WEB-сервером. При этом WEB-клиент может получить доступ на чтение к объектам удаленной системы посредством WEB-сервера, однако, подобное отношение не дает клиенту доступа на запись к удаленным объектам (за исключением особых случаев), к которым имеет доступ процесс-сервер. Ограничения доступа указанного вида не отражаются в моделях, основанных на определенном выше понятии отношения доверия. Ниже введено понятие ограниченного отношения доверия, позволяющего формализовать системы, в которых один субъект может выполнять по запросам другого лишь некоторые операции над объектами. Однако, модели, основанные на ограниченных отношениях доверия, также имеют недостатки, которые перечислены ниже.

Иногда имеет смысл рассматривать локальные отношения доверия, то есть множества вида $T_{A,A} \subset S(A) \times S(A)$. Обычно доверие между субъектами монолитной информационной системы необходимо в том случае, если политика безопасности системы не может быть записана в терминах модели, поддерживаемой установленной ОС.

Рассмотрим пример. Модель безопасности, принятая в UNIX-подобных ОС, не позволяет разрешить какому-либо пользователю добавление данных к некоторому файлу, запретив при этом модификацию ранее записанных туда данных. По этой причине обычно доступ на запись к файлам системных журналов запрещен всем пользователям, кроме пользователя `root`, а для добавления информации в системный журнал пользовательские приложения обращаются к программе `syslogd`. Таким образом, пользовательская программа может получить доступ к объекту — системному журналу, посредством приложения `syslogd`, то есть `syslogd` доверяет пользовательской программе.

Еще один пример локального отношения доверия связан с использованием локальных портов с помощью системного вызова `bind`. В ОС Linux, только приложения пользователя `root` могут использовать порты от 1 до 1023, называемые привилегированными. С учетом того, что сетевые сервисы подвергаются внешним атакам чаще, чем любые другие приложения, для повышения безопасности системы в целом права доступа сетевых сервисов должны быть минимальны. Возможность запуска сетевых сервисов (за исключением средств удаленного администрирования) предоставляется в ОС Linux с помощью программы `bindd`. Указанное приложение, выполняющееся с привилегиями пользователя `root`, получает от другого приложения дескриптор сетевого канала (сокета), и связывает его с определенным локальным адресом и портом, если данная операция допускается политикой безопасности. При этом приложение, пославшее запрос к `bindd`, может связать сокет с привилегированным портом, не имея привилегий пользователя `root`. Таким образом, различные субъекты могут получить

доступ к локальным адресам и портам посредством приложения `bindd`, а это означает, что `bindd` доверяет некоторым другим приложениям.

Изложенные выше соображения свидетельствуют, что отношения доверия позволяют сделать выразительные свойства локальной политики безопасности более богатыми. Основываясь на введенном понятии отношения доверия, можно строить политики безопасности для распределенных систем в условиях, когда политики безопасности для компонент заранее заданы.

1.1 Объединение политик безопасности с помощью отношений доверия

В данном разделе рассмотрены примеры построения общей политики безопасности для распределенной информационной системы с использованием отношений доверия между ее компонентами. Для простоты все суждения проведены для случая распределенной системы, состоящей из двух монолитных систем A и B. Если система C — это объединение систем A и B, то скажем, что политика безопасности системы C построена с помощью пары отношений доверия $T_{A,B}$ и $T_{B,A}$, если каждый субъект S_A системы A получает те доступы к объектам системы B, которые имеет хотя бы один субъект S_B системы B, доверяющий S_A. Аналогично определяются права доступа субъектов системы B к объектам системы A. При объединении политик безопасности с помощью отношений доверия в системы во многих случаях будут добавляться новые субъекты, и подобное действие не следует считать противоестественным, так как на практике, чтобы интегрировать замкнутую монолитную систему в распределенную среду, необходимо добавить в указанную систему сетевые приложения.

1.1.1 Объединение ролевых политик безопасности

В данном разделе описано применение отношений доверия при построении ролевой политики безопасности для распределенной системы, состоящей из двух подсистем A и B, в которых заданы ролевые политики безопасности. Это означает, что в каждой из систем задано множество P элементарных привилегий и каждому субъекту S посредством некоторых отношений между субъектами, сеансами, пользователями и ролями приписано множество $R_S \subset P$, называемое ролью субъекта S. Ввиду того, что ролевая модель разграничения доступа разрабатывается относительно недавно, в некоторых источниках можно встретить описания, отличающиеся от использованного в данном разделе. В [5], например, указано несколько вариантов усовершенствования ролевой модели разграничения доступа. Ограничимся, однако, рассмотрением вышеописанного простейшего варианта.

Определение 2. Пусть информационные системы A и B имеют ролевые политики безопасности. Объединением систем A и B называется система C, субъектами которой являются субъекты систем A и B, а элементарными привилегиями — элементарные привилегии систем A и B. Скажем, что ролевые политики безопасности систем A и B объединены корректно, если в системе C всем субъектам приписаны роли таким образом, что каждый субъект S_A системы A имеет в системе A те же привилегии, которые он имеет в соответствии с политикой безопасности системы A, и, аналогично, каждый субъект S_B системы B имеет в системе B те же привилегии, которые он имеет в соответствии с политикой безопасности системы B.

Теорема 1. *Пусть информационные системы A и B имеют ролевые политики безопасности, причем множества $P(A)$ и $P(B)$ привилегий систем A и B задаются администратором и не изменяются в процессе функционирования систем. Тогда любое корректное объединение ролевых политик безопасности систем A и B может быть выражено с помощью отношений доверия между системами A и B.*

Доказательство. Пусть $r_{A,1}, r_{A,2}, \ldots r_{A,N_A}$ — элементарные привилегии системы A. Добавим в систему A роли $R_{A,1} = \{r_{A,1}\}, \ldots R_{A,N_A} = \{r_{A,N_A}\}$ и субъекты $S_{A,1}, S_{A,2}, \ldots S_{A,N_A}$. Припишем субъекту $S_{A,j}$ роль $R_{A,j}$ для $j = 1, 2, \ldots N_A$. Далее, пусть каждый субъект $S_{A,j}$ доверяет тем и только тем субъектам системы B, которые должны иметь в системе A привилегию $r_{A,j}$. То есть строится отношение доверия $T_{B,A}$.

Таким образом, субъекты системы B имеют необходимые привилегии в системе A. Аналогично строится отношение доверия $T_{A,B}$, дающее субъектам системы A привилегии в системе B. □

1.1.2 Объединение многоуровневых политик безопасности

Итак, любое корректное объединение ролевых политик безопасности может быть выражено с помощью пары отношений доверия. Однако, аналогичное утверждение неверно для многоуровневых политик безопасности, то есть политик, при использовании которых каждому объекту системы приписан уровень секретности, и права доступа субъектов к объектам устроены так, чтобы информационные потоки с более высокого уровня секретности на более низкий были невозможны. Это значит, что доступ субъекта к объекту на чтение разрешается в том и только том случае, если уровень секретности субъекта не ниже уровня секретности объекта. Доступ на запись разрешается, наоборот, если уровень секретности субъекта не выше уровня секретности объекта.

Определение 3. Пусть информационные системы A и B имеют многоуровневые политики безопасности. Объединением систем A и B называется система C, множество объектов которой является дизъюнктным объединением множеств объектов систем A и B, а множество субъектов — дизъюнктивным объединением множеств субъектов систем A и B, то есть $O(C) = O(A) \sqcup O(B)$ и $S(C) = S(A) \sqcup S(B)$. Скажем, что многоуровневые политики безопасности систем A и B объединены корректно, если в системе C задана многоуровневая политика безопасности, такая, что:

- Субъектам системы A разрешены те и только те доступы к объектам системы A, которые разрешает политика безопасности системы A.
- Субъектам системы B разрешены те и только те доступы к объектам системы B, которые разрешает политика безопасности системы B.
- Субъекты системы A не могут передавать информацию «сверху вниз» по решетке ценностей системы A, используя объекты системы B.
- Субъекты системы B не могут передавать информацию «сверху вниз» по решетке ценностей системы B, используя объекты системы A.

Теорема 2. *Пусть системы A и B имеют многоуровневые политики безопасности и при этом в обеих системах на каждом уровне секретности имеется по крайней мере один субъект. Пусть корректное объединение многоуровневых политик безопасности систем A и B может быть выражено через отношения доверия между субъектами систем A и B. Тогда решетки ценностей систем A и B изоморфны между собой и решетка ценностей системы C, являющейся объединением систем A и B, также изоморфна им.*

Для доказательства данного утверждения потребуется несколько лемм.

Лемма 1. *Пусть S_1 и S_2 — произвольные субъекты системы A. Тогда, если в системе C субъекты S_1 и S_2 лежат на одном уровне секретности, то и в системе A указанные субъекты лежат на одном уровне секретности.*

Доказательство. Докажем указанный факт от противного. Пусть субъекты S_1 и S_2 лежат на разных уровнях секретности в системе A. Без ограничения общности будем считать, что если уровни секретности субъектов S_1 и S_2 сравнимы, то S_1 лежит выше S_2. Тогда в системе A существует объект O, доступ к которому на чтение разрешен субъекту S_1, но запрещен субъекту S_2. Но объект O является также объектом системы C, в которой субъекты S_1 и S_2 имеют такие же права доступа к нему, как и в системе A. Значит, в системе C субъекты S_1 и S_2 лежат на разных уровнях секретности. Полученное противоречие доказывает лемму. □

Лемма 2. *Пусть S_1 и S_2 — субъекты системы A, лежащие в системе A на одном уровне секретности. Тогда в политике безопасности системы C права доступа ко всем объектам системы B одинаковы для субъектов S_1 и S_2.*

Доказательство. Предположим противное. Пусть O_B — объект системы B. Пусть субъект S_1, в соответствии с политикой безопасности системы C, имеет доступ на чтение к объекту O_B, а субъект S_2 такого права доступа не имеет. Пусть O_A — объект системы A, лежащий на одном уровне секретности с S_1 и S_2. Субъект S_1 имеет доступ на чтение к объекту O_B, следовательно, в решетке ценностей системы C субъект S_1 либо лежит выше объекта O_B, либо на одном уровне с ним. Субъект S_1 имеет

доступ на запись к объекту O_A, и поэтому объект O_A либо лежит выше субъекта S_1, либо на одном уровне с ним. И, наконец, субъект S_2 имеет доступ на чтение к объекту O_A. Это значит, что S_2 лежит в решетке ценностей системы C либо выше объекта O_A, либо на одном уровне с ним. Пусть функция $L(O)$ задает уровень секретности объекта O в системе C. Тогда

$$L(O_B) \leqslant L(S_1) \leqslant L(O_A) \leqslant L(S_2)$$

а, значит, $L(O_B) \leqslant L(S_2)$. Поэтому субъект S_2 обязан иметь доступ на чтение к объекту O_B.

Аналогично рассматривается случай, когда субъект S_1, в соответствии с политикой безопасности системы C, имеет доступ на запись к объекту O_B, а субъект S_2 такого доступа не имеет. □

Лемма 3. *Пусть S_1 и S_2 — субъекты системы A. Тогда субъекты S_1 и S_2 имеют один и тот же уровень секретности в системе A тогда и только тогда, когда они имеют один и тот же уровень секретности в системе C.*

Доказательство. По лемме 1, если S_1 и S_2 лежат на одном уровне секретности в системе C, то они лежат на одном уровне секретности и в системе A. Необходимо доказать обратное утверждение. Пусть субъекты S_1 и S_2 лежат на одном уровне секретности в системе A. Тогда все права доступа к объектам системы A одинаковы для субъектов S_1 и S_2. Но, в силу леммы 2, все права доступа ко всем объектам системы B также одинаковы для субъектов S_1 и S_2. Поэтому субъекты S_1 и S_2 обладают одинаковыми правами доступа в системе C, то есть лежат на одном уровне секретности в системе C. □

Лемма 4. *Решетки ценностей систем A и B вложены в решетку ценностей системы C.*

Доказательство. Докажем лемму для решетки ценностей системы A. Обозначим уровни секретности системы A символами $l_1 \dots l_n$, а уровни секретности системы C, являющейся объединением систем A и B, символами $L_1 \dots L_m$. Обозначим $F\colon \{l_1 \dots l_n\} \to \{L_1 \dots L_m\}$ — отображение, такое, что если субъект S_1 лежит на уровне l_j в системе A, то тот же субъект лежит на уровне $L_k = F(l_j)$ в системе C. Из леммы 3 следует, что отображение F определено корректно и, кроме того, данное отображение инъективно. Осталось показать, что если уровень l_1 выше, чем l_2, то и $F(l_1)$ выше, чем $F(l_2)$, и, кроме того, что если уровни l_1 и l_2 несравнимы, то и уровни $F(l_1)$ и $F(l_2)$ также несравнимы.

В самом деле, пусть уровень секретности l_1 выше, чем l_2. Пусть субъект S_1 лежит на уровне l_1, а субъект S_2 — на уровне l_2. В системе A существует объект O, такой, что субъект S_1 имеет доступ на чтение к O, а S_2 — доступ на запись к O. То же самое верно и в политике безопасности системы C, а значит, $F(l_1)$ либо лежит в решетке ценностей системы C выше, чем $F(l_2)$, либо $F(l_1)$ и $F(l_2)$ совпадают. Но $F(l_1)$ и $F(l_2)$ не могут совпадать в силу инъективности F, следовательно, $F(l_1)$ либо лежит выше, чем $F(l_2)$.

Теперь пусть l_1 и l_2 несравнимы в решетке ценностей системы A. Пусть объект O и субъект S_1, входящие в систему A, лежат на уровне l_1, а субъект S_2 системы A лежит на уровне l_2. Субъект S_1 имеет к объекту O доступ как на чтение, так и на запись, и поэтому в системе C объект O лежит на уровне $F(l_1)$, как и субъект S_1. Если бы уровень $F(l_2)$ был сравним с уровнем $F(l_1)$, то субъект S_2 имел бы в системе C, а значит, и в системе A, доступ к объекту O либо на чтение, либо на запись, но S_2 не имеет никаких прав доступа к O. Возникло противоречие. □

Доказательство теоремы 2. Пусть многоуровневые политики безопасности систем A и B объединены с помощью отношений доверия между субъектами систем A и B. Обозначим решетки ценностей систем A, B и C, соответственно, L_A, L_B и L_C. По лемме 4, $L_A \subset L_C$. Необходимо доказать, что $L_A = L_C$. Пусть уровень $l_0 \in L_C$ не соответствует никакому уровню из L_A. Тогда l_0 соответствует какому-то уровню из L_B, так как в противном случае l_0 не мог бы содержать объектов ни одной из систем A и B, то есть не содержал бы объектов вообще. Пусть субъект S_B системы B лежит на уровне l_0.

Пусть уровень l_0 в решетке L_C сравним с каким-нибудь уровнем секретности системы A. Тогда субъект S_B имеет какие-то права доступа к какому-то объекту O системы A, то есть в системе A найдется субъект S_A, доверяющий S_B. Тогда субъект S_A также лежит в решетке ценностей системы C на уровне l_0. В самом деле, если объект O лежит в системе A на том же уровне, что и субъект S_A, то S_A имеет доступ к O как на чтение, так и на запись. В системе C субъект S_B имеет такие же права доступа к объекту O, следовательно, субъекты S_B, S_A и объект O лежат на одном уровне

секретности, а именно, на уровне l_0. В системе A уровню l_0, таким образом, соответствует уровень секретности, на котором лежит S_A. Если же субъект S_A лежит выше или ниже объекта O, то вместо O в приведенных рассуждениях можно использовать какой-либо объект O', лежащий в системе A на том же уровне секретности, что и субъект S_A.

Таким образом, теорема доказана для случая, когда уровень l_0 в решетке L_C сравним с каким-нибудь уровнем секретности системы A. Предположим, однако, что это не так. В этом случае субъекты всех уровней L_C, находящихся выше l_0, не имеют доступа на запись ни к одному из субъектов системы A. Но в решетке L_C существует наибольший элемент $l_{\max}$. Но тогда $l_{\max}$ также не соответствует никакому уровню секретности системы A, и субъекты системы B, находящиеся на уровне $l_{\max}$, не имеют также прав доступа на чтение ни к одному объекту системы A. Но тогда ни один субъект системы C не имеет прав доступа на чтение ни к одному объекту системы A. Но субъекты системы A также являются субъектами системы C, поэтому данный случай невозможен. □

Таким образом, доказано, что многоуровневые политики безопасности на двух информационных системах A и B могут быть объединены с помощью отношений доверия только в том случае, когда решетки ценностей систем A и B изоморфны. Однако, многоуровневые политики безопасности можно, вообще говоря, объединить и без отмеченного условия. Например, можно всем субъектам системы A разрешить только чтение всех объектов системы B, а всем субъектам системы B — только запись во все объекты системы A. Тогда, если системы A и B имеют многоуровневые политики безопасности, то и «объединенная» политика безопасности будет многоуровневой, при любой структуре решеток ценностей систем A и B. Таким образом, не любые политики безопасности можно объединить с помощью отношений доверия.

Заметим, что в данной теореме существенно то условие, что на каждом уровне секретности должен быть по крайней мере один субъект. Если это условие не выполнено, то с помощью отношений доверия в некоторых случаях возможно объединить многоуровневые политики безопасности, имеющие не изоморфные решетки ценностей.

Рассмотрим пример. Пусть в системе A два уровня секретности, обозначенные как H_A и L_A, при этом уровень H_A более высокий, чем L_A. В системе B три уровня секретности: H_B, M_B, L_B, при этом $L_B < M_B < H_B$, и на уровне M_B нет ни одного субъекта. В этом случае построим отношения доверия следующим образом. Пусть каждый субъект системы A, находящийся на уровне L_A, доверяет каждому субъекту системы B, находящемуся на уровне L_B. Аналогично, пусть каждый субъект системы A, находящийся на уровне H_A, доверяет каждому субъекту системы B, находящемуся на уровне H_B. Таким образом формируется отношение доверия $T_{B,A}$ субъектов системы A к субъектам системы B. Отношение доверия субъектов системы B к субъектам системы A определяем как $T_{A,B} = T^T_{B,A}$, то есть $(S_A, S_B) \in T_{A,B}$ тогда и только тогда, когда $(S_B, S_A) \in T_{B,A}$. В этом случае система C, являющаяся объединением систем A и B, имеет многоуровневую политику безопасности, имеющую три уровня секретности: H_C, M_C, L_C. Легко видеть, что H_C будет содержать объекты систем A и B, лежащие соответственно на уровнях H_A и H_B, аналогично, L_C будет содержать объекты систем A и B, лежащие соответственно на уровнях L_A и L_B, а уровень M_C будет содержать объекты системы B, лежащие на уровне M_B.

1.2 Ограниченные отношения доверия

Рассмотренный выше метод объединения политик безопасности, использующий отношения доверия, имеет ряд недостатков. Один из недостатков состоит в том, что при анализе защищенности информационных систем не учитывается природа субъектов, доверяющих другим субъектам. Так, в примере с программой `syslogd` субъект, имеющий доступ на запись к системным журналам, доверяет всем остальным субъектам. Таким образом, из описания отношений доверия не следует, что произвольный субъект не может очистить журнал регистрации событий, в то время, как подобные действия должны быть запрещены политикой безопасности.

Второй недостаток данного метода заключается в том, что отношения доверия позволяют объединить, вообще говоря, не любые политики безопасности. Например, выше было показано, что многоуровневые политики безопасности не всегда могут быть объединены с помощью отношений доверия.

Для решения этих проблем введем понятие *ограниченного*, или *нагруженного* отношения доверия. Смысл его в том, что субъект S_B может доверять другому субъекту S_A, лишь частично. Это значит,

что S_A может совершать посредством S_B не все операции над объектами, которые может совершать S_B, а только некоторые из них.

Определение 4. Пусть M некоторая решетка, называемая решеткой доверия. Ограниченным, или нагруженным отношением доверия между информационными системами A и B называется пара $(T_{A,B}, F)$, где $T_{A,B}$ — это отношение доверия между A и B, а F — отображение множества $T_{A,B}$ в M. Тогда $F(S_A, S_B)$ называется уровнем доверия S_B к S_A.

При этом, пусть каждой операции, которую может выполнить S_B также соответствует некоторый уровень доверия $m_0 \in M$. Тогда S_A может выполнить некоторую операцию посредством S_B, если $(S_A, S_B) \in T_{A,B}$ и $m_0 \leqslant F(S_A, S_B)$.

В общем случае, решетка доверия M может зависеть от конкретного субъекта S_B системы B. Тогда значение $F(S_A, S_B)$ должно принадлежать $M(S_B)$.

Рассмотрим пример. Пусть S_2 — это программа `bindd`, описанная выше. Тогда в случае, если система, на которой функционирует сервер `bindd`, имеет один локальный адрес, M — это множество всех подмножеств множества портов $P = 1, 2, \ldots 65535$. Тогда для каждого субъекта S_1, использующего локальные порты с помощью S_2, уровень доверия $F(S_1, S_2)$ представляет собой множество портов, использование которых разрешено для S_1. Однако, в том случае, если система имеет несколько локальных IP-адресов, множество M будет состоять из всех подмножеств множества $P \times A$ всех упорядоченных пар (порт, локальный адрес).

1.2.1 Объединение многоуровневых политик безопасности с помощью ограниченных отношений доверия

Ограниченные отношения доверия позволяют устранить недостаток простых отношений доверия, состоящий в том, что не любые многоуровневые политики безопасности могут быть объединены с помощью простых отношений доверия.

Теорема 3. *Любое корректное объединение многоуровневых политик безопасности систем A и B может быть выражено с помощью ограниченных отношений доверия между системами A и B.*

Доказательство. Построим ограниченное отношение доверия субъектов системы B к субъектам системы A. Пусть $L_1(B) \ldots L_N(B)$ — уровни секретности системы B, а $L_1(C) \ldots L_M(C)$– уровни секретности системы C, являющейся объединением систем A и B.

Добавим в систему B субъекты $S_1 \ldots S_N$, такие, что для любого $j \in \{1 \ldots N\}$ субъект S_j лежит на уровне $L_j(B)$. Составим ограниченное отношения доверия следующим образом: каждый субъект S_j имеет решетку доверия $L = 2^P$, где P — множество доступов, состоящее в простейшем случае из двух элементов: $P = \{read, write\}$.

Таким образом уровень доверия субъекта S_j к субъекту S_A системы A является подмножеством $P' \subset P$. Тогда, если $read \in P'$, то S_A может осуществлять чтение объектов на уровне $L_j(B)$ посредством субъекта S_j. Если $write \in P'$, то S_A может осуществлять запись объектов на уровне $L_j(B)$ посредством субъекта S_j.

Пусть субъект S_A лежит в системе A на уровне $L_S(A)$, которому в системе C соответствует уровень $L_S(C)$ (выше доказано, что решетки ценностей систем A и B вложены в решетку ценностей системы C). Пусть для каждого j уровню секретности $L_j(B)$ системы B соответствует уровень секретности $L_{f(j)}(C)$ системы C. Указанное соотношение однозначно определяет функцию $f\colon \{1, \ldots, N\} \to \{1, \ldots, M\}$.

Тогда, для каждого $j \in \{1, \ldots, N\}$, определим уровень доверия S_j к S_A следующим образом:

- Если $L_S(C) = L_{f(j)}(C)$, то уровень доверия S_j к S_A совпадает с множеством P.
- Если $L_S(C) < L_{f(j)}(C)$, то уровень доверия S_j к S_A является одноэлементным множеством $P_w = \{write\}$.
- Если $L_S(C) > L_{f(j)}(C)$, то уровень доверия S_j к S_A является одноэлементным множеством $P_r = \{read\}$.
- Если $L_S(C)$ несравним с $L_{f(j)}(C)$, то уровень доверия S_j к S_A является пустым множеством.

Таким образом, объединение многоуровневых политик безопасности выражено с помощью пары ограниченных отношений доверия, что и требовалось доказать. □

1.2.2 Объединение ролевых политик безопасности с помощью ограниченных отношений доверия

Выше было показано, что любое корректное объединение ролевых политик безопасности на двух информационных системах A и B может быть выражено с помощью отношений доверия между субъектами указанных систем. Однако, при объединении с помощью простых отношений доверия политик безопасности систем, использующих большое количество атомарных привилегий, неоправданно растет число дополнительных субъектов, вводимых для их объединения. Использование ограниченных отношений доверия позволяет объединять ролевые политики безопасности, вводя лишь небольшое число дополнительных субъектов (по одному для каждой объединяемой системы).

Теорема 4. *Любое корректное объединение ролевых политик безопасности систем A и B может быть выражено с помощью ограниченных отношений доверия между системами A и B.*

Доказательство. Пусть S_B^A — субъект системы B, ролью которого является все множество $E(B)$ привилегий системы B. Пусть решетка доверия для субъекта S_B^A совпадает с множеством $R(B) = 2^{E(B)}$ ролей системы B. Тогда для каждого субъекта S_A системы A уровень доверия субъекта S_B^A к субъекту S_A определяется формулой $m = r_C(S_A) \cap E(B)$, где $r_C(S_A)$ является ролью субъекта S_A в системе C. Таким образом, построено ограниченное отношение доверия субъектов системы B к субъектам системы A.

Аналогично, пусть S_A^B — субъект системы A, ролью которого является все множество $E(A)$. Для S_A^B можно аналогичным образом определить решетку доверия и уровни доверия к субъектам системы B. □

С практической точки зрения приведенный способ объединения политик безопасности, однако, плох тем, что в системе появляется единая точка уязвимости, то есть коммуникационный субъект, имеющий все возможные привилегии. Успешная атака на подобный субъект даст злоумышленнику полный контроль над системой. С учётом этого обстоятельства можно предложить комбинированный подход, подразумевающий наличие в системе B нескольких субъектов, частично доверяющих субъектам системы A, каждый из которых должен предоставлять удаленным субъектам привилегии, относящиеся к некоторому ассоциированному с ним подмножеству.

1.2.3 Недостатки моделей безопасности, основанных на ограниченных отношениях доверия

Одним из основных недостатков моделей, основанных на нагруженных отношениях доверия, является их сложность. Для многих встречающихся на практике случаев трудно составить точное формальное описание решетки доверия подсистем.

Одним из способов построения решетки доверия для приложения (рассматриваемого как субъект некоторой сложной системы) является использования множества запросов, которые могут быть переданы указанному приложению на выполнение. Если обозначить это множество запросов через R, то можно положить $M = 2^R$ множеством всех подмножеств R. При этом стороннему субъекту S_1, имеющему уровень доверия m со стороны субъекта S_2, разрешается передавать S_2 запрос r, в том и только в том случае, если $r \in m$.

Проблема указанного подхода в том, что множество возможных запросов для многих приложений крайне велико и не имеет четкой математической структуры.

Вторым недостатком моделей, основанных на нагруженных отношениях доверия, является возможность уязвимости программного обеспечения к атакам на переполнение буфера и некоторым другим, которые позволяют злоумышленнику заставить удаленное приложение выполнить заранее подготовленный код. То есть, если субъект S_B ограниченно доверяет S_A, то с помощью атаки переполнения буфера или использовав какую-нибудь другую уязвимость программы S_B, S_A может заставить S_B выполнить операцию, которую S_B, согласно своему предназначению, не должен быть выполнять. В этом случае ограниченность доверия перестает играть какую-либо роль и только права доступа S_B определяют виды доступа, которые может осуществить S_A в системе, к которой относится S_B. Таким образом, из ограниченных, или нагруженных, отношения доверия, благодаря уязвимостям ПО, могут «превратиться» в простые. Учитывая такую возможность, представляются необходимыми дальнейшие исследования, направленные на изучение устойчивости функций безопасности распределенных систем, основанных на ограниченных отношениях доверия, к компрометации некоторых компонент.

1.3 Выводы

Выше рассмотрено два возможных средства построения политики безопасности для распределенной системы. Оба способа имеют свои достоинства и недостатки, однако заметим, что контроль выполнения простых отношений доверия со стороны ядра операционной системы представляется более простым и надежным, нежели контроль выполнения ограниченных отношений доверия. Поэтому при практической реализации защитных механизмов разумно возложить контроль простых отношений доверия на специальный модуль ядра ОС, оставив за приложениями проверку уровня доверия к удаленным субъектам.

2 Практическое построение распределенной системы, использующей отношения доверия

В данном разделе рассматривается подход к практическому построению распределенной информационной системы, имеющей политику безопасности, основанную на отношениях доверия. Для задания и контроля выполнения отношений доверия разрабатывается дополнительный модуль ядра ОС Linux. Разрабатываемая система опирается на функции модуля безопасности SELinux — разработки Агентства Национальной Безопасности, получившей в данный момент достаточно широкое распространение благодаря поддержке со стороны многих производителей дистрибутивов Linux, например, Red Hat.

2.1 Модуль безопасности SELinux

Система SELinux, разработанная Агентством Национальной Безопасности (NSA) США, представляет собой набор дополнений к ядру Linux и прикладных программ, служащих для усовершенствования механизмов разграничения доступа в ОС Linux. SELinux поддерживает модель Type Enforcement, краткое описание которой приведено ниже, а также ролевую и многоуровневую модели безопасности. При использовании SELinux продолжают работать традиционные механизмы безопасности UNIX-систем, основанные на усеченной дискреционной модели.

2.2 Модель безопасности Type Enforcement

Политика безопасности в SELinux задается в терминах модели Type Enforcement (TE). Суть этой модели в том, что каждому объекту системы приписана метка, называемая *типом*. Тип также называется *доменом*, если хотя бы один субъект системы может иметь этот тип. Кроме того, каждый объект относится к какому-либо *классу*, задающему его «сущность», например, является ли объект обычным файлом, файлом устройства, каналом FIFO, процессом и т.д. Разрешенные доступы субъектов к объектам описываются отображением $F\colon T \times T \times C \times A \to \{allow, deny\}$, где T — это множество типов системы, C — множество классов а A — множество всех возможных доступов. Таким образом, разрешение доступа определяется типом субъекта, типом объекта, классом объекта и видом доступа. Зависимость возможных доступов от класса объекта проявляется, например, тем, что сигнал можно послать процессу, но нельзя послать пассивным объектам.

Политики, основанные на модели Type Enforcement, всегда могут быть объединены с помощью отношений доверия. Чтобы доказать это утверждение, потребуются дополнительные обозначения. Если A — компонента распределенной информационной системы, то через $T(A)$ обозначим множество типов системы A, а через $T'(A) = T(A) \sqcup \{none\}$ — множество типов системы A, где $none$ — специальный тип, не имеющий прав доступа.

Определение 5. Пусть A, B и C — некоторые информационные системы, причём система C построена по следующим правилам: $O(C) = O(A) \sqcup O(B)$, $S(C) = S(A) \sqcup S(B)$, $T(C) \subset T'(A) \times T'(B)$. При этом права доступа субъекта S типа $t = (t_A, t_B)$ к объектам системы A определяются компонентом t_A, а к объектам системы B — компонентом t_B. В этом случае система C называется корректным объединением систем A и B, если каждому субъекту из $S(A)$ разрешены те и только те доступы к объектам из $O(A)$, которые разрешены ему в соответствии с политикой безопасности системы A, и, аналогично, каждому субъекту системы из $S(B)$ разрешены те и только те доступы к объектам их $O(B)$, которые разрешены ему в соответствии с политикой безопасности системы B.

Теорема 5. *Любое корректное объединение политик безопасности TE систем A и B может быть выражено с помощью отношений доверия между субъектами систем A и B.*

Доказательство. Пусть субъект S_A системы A, имеющий тип t_A доверяет субъекту S_B системы B, имеющему тип t_B, тогда и только тогда, когда в системе C субъект S_B имеет тип $t = (t_A, t_B)$. Построенное по вышеописанному правилу отношение доверия $T_{A,B}$ дает субъектам системы B необходимые права доступа к объектам системы A. Аналогичным образом можно определить отношение доверия $T_{A,B}$, дающее субъектам системы A необходимые права доступа к объектам системы B. □

2.3 Модель разрабатываемой системы

В настоящий момент активно ведется разработка добавлений к ядру Linux, позволяющих администратору распределенной информационной системы задавать отношения доверия между субъектами различных ее компонент, при наличии контроля за отношениями доверия со стороны ядра. Будем считать, что отношение доверия согласовано с разбиением объектов на типы, то есть если пара $(S_A, S_B) \in T_{A,B}$ и пары субъектов S_A и S'_A и S_B и S'_B имеют одинаковые типы в модели Type Enforcement, то $(S'_A, S'_B) \in T_{A,B}$. Иными словами, администратору необходимо задавать не доверие между субъектами (что достаточно сложно из-за того, что множество субъектов быстро меняется со временем), а между типами систем A и B, что удобнее, так как множества типов систем задаются администратором и не изменяются до задания новой политики безопасности.

При сетевом взаимодействии между субъектами различных компонент системы каждый из взаимодействующих субъектов определяет тип другого по меткам сетевых пакетов, которые передаются в поле опций заголовка IP-пакета. При этом пересылаются не символьные имена типов, а их целочисленные идентификаторы, что уменьшает взаимозависимость между политиками безопасности разных узлов. С той же целью для каждого узла допускается несколько таблиц соответствия между типами, принятыми в SELinux, и этими целочисленными идентификаторами. Таким образом, узел может входить одновременно в несколько подсетей, принадлежащим сложной распределенной информационной системе, имеющих различные политики назначения идентификаторов типов.

Опишем формальную модель предлагаемой системы.

Пусть A — компонента распределенной информационной системы, $T(A)$ — множество типов системы A, M — множество узлов, с которым взаимодействует узел A. Пусть M разбито на непересекающиеся подмножества $M = M_1 \cup \ldots \cup M_N$, тогда для каждого $k \in \{1 \ldots N\}$ имеется отображение $F_k \colon T(A) \to I_k(A)$, где $I_k(A)$ является подмножеством натуральных чисел, а $T(A)$ — множество типов системы A. При этом отображение сюръективно на $I_k(A)$ и все элементы $I_k(A)$, кроме нуля, имеют единственный прообраз в $T(A)$. Такие отображения F_k назовём таблицами соответствия типов. Нулевое значение идентификатора выделено как означающее, что уровень доверия неизвестен.

Пусть для каждого узла $B \in M_k$ задано множество $T'_{A,B} \subset I_k(A) \times I_j(B)$, где j — такое, что узел A лежит в j-й компоненте множества внешних узлов по отношению к B (также, как B лежит в k-й компоненте M_k множества внешних узлов по отношению к A). Субъекту S_B, имеющему тип t_B, разрешается взаимодействие с субъектом S_A, имеющим тип t_A, тогда и только тогда, когда $(F_k(t_A), F_j(t_B)) \in T'_{A,B}$. Указанное условие определяет отношение доверия между субъектами систем A и B.

2.4 Текущее состояние работ и планы на будущее

Для настройки данных об идентификаторах типов и о доверии между ними введено подсемейство `NETLINK_TRUST` протокола Netlink. Интерфейс протокола Netlink удобнее интерфейса системных вызовов по нескольким причинам, например, из-за того, что Netlink позволяет приложению получать уведомления об изменениях настроек в ядре, производимых другими приложениями (к сожалению, в текущей реализации системы настройки отношений доверия указанная возможность не поддерживается).

В данный момент разработка вышеописанного дополнительного механизма безопасности находится на начальном этапе. Тем не менее, уже возможно назвать некоторые достоинства разрабатываемых программных средств.

- Возможность объединения политик безопасности, развивающихся независимо друг от друга. При изменении политики безопасности одного из узлов другие узлы не нужно перенастраивать при условии, что целочисленные идентификаторы типов сохранят свой смысл.

- Возможность интеграции узла в несколько различных распределенных информационных систем, имеющих различные политики назначения идентификаторов типов. В этом случае для связи с узлами разных систем необходимо использовать разные таблицы соответствия типов.
- Необязательность использования единого сервера аутентификации для объединения отдельных узлов в распределенную систему. Это позволяет строить системы сложной структуры, не имеющие единой точки отказа.

Заключение

В виду все более возрастающей роли распределенных информационных систем во многих сферах деятельности человека нельзя недооценивать важность задачи защиты подобных систем от злонамеренных пользователей. Так как, к сожалению, невозможно гарантировать абсолютную защиту каждой из компонент распределенной системы, возникает задача защиты отдельных участков системы в тех случаях, когда некоторые из компонент полностью, либо частично контролируются злоумышленником.

Для решения упомянутого вопроса необходимо строить политики безопасности для распределенных информационных системы таким образом, чтобы взаимозависимость компонент была, с одной стороны, как можно меньше, однако, с другой стороны, оказывалась достаточной для выполнения системой своих функций.

В данной работе рассмотрен один из подходов к формализации понятия доверия между компонентами распределенных информационных систем, позволяющий моделировать объединение политик безопасности, основанных на различных распространенных моделях разграничения доступа. Кроме того, описаны некоторые технические решения, принятые авторами в ходе разработки программных средств для облегчения построения распределенных систем с применением данного подхода.

Литература

[1] А. А. Грушо, Е. Е. Тимонина. Теоретические основы защиты информации. М.: Яхтсмен, 1996.

[2] В. А. Васенин. Информационная безопасность и компьютерный терроризм. В сб. «Научные и методологические проблемы информационной безопасности», М.: МЦНМО, 2005.

[3] В. А. Васенин, А. В. Галатенко. Математические модели распределенных информационных систем. Материалы конференции «Математика и безопасность информационных технологий», 2004 г. М.: МЦНМО, 2005.

[4] О. О. Андреев. Сравнение ролевой и дискреционной моделей разграничения доступа. Материалы конференции «Математика и безопасность информационных технологий» 2004 г. М., МЦНМО, 2005.

[5] Ravi S. Sandhu, Edward J. Coyne, Hal L. Feinstein, Charles E. Youman. Role-Based Access Control Models. IEEE Computer, 29, 2, 38–47, 1996.

[6] Дж. Макнамара. Секреты компьютерного шпионажа: Тактика и контрмеры. М.: БИНОМ, 2004.

[7] Gowri Dhandapani. Netlink Sockets — Overview. The University of Kansas, 1999.

К анализу подходов классификации компьютерных атак

А. А. Климовский

Введение

Постоянно увеличивающееся число компьютерных атак [1] приводит к необходимости создания организованных (или самоорганизующихся) структур, деятельность которых направлена на обеспечение актуальной информацией о найденных киберуязвимостях, на оперативное их устранение, на создание систем обнаружения вторжений (систем активного аудита) и ряд других мер [2]. По этой причине недостатка в информации относительно последних уязвимостей и компьютерных атак не наблюдается. Однако зачастую (особенно это касается атак) такая информация очень разнородна, неструктурированна и мало пригодна для дальнейшего анализа. Как следствие, возникает необходимость в разработке модели и инструментария, позволяющих упорядочить и систематизировать накопленные знания, — создания таксономии.

Кроме содержательного и систематического описания компьютерных атак, на практике таксономия атак нужна также для их дальнейшего анализа, с целью аккумулирования знаний при оценке рисков, и создания моделей нарушителя с тем, чтобы проектировать критически важные системы, в частности, для выработки политики безопасности, а также для создания средств активного аудита [3].

1 Постановка задачи

Прежде чем перейти непосредственно к постановке задачи, определим понятие атака (для этого воспользуемся определением, данным в работе [4]):

Атака — последовательность действий предпринимаемых кем-либо для достижения несанкционированного результата, то есть действий, направленных на нарушение правил функционирования системы, установленных ее владельцем.

Субъект, совершающий эти действия, будем называть *атакующим*, а систему, на которую производится атака — *объектом атаки.*

С формальной точки зрения, задача классификации атак состоит в том, чтобы создать систему их категорирования, а именно, — выделить критерии — отличительные черты каждой из них и задать классификационную схему, как способ отнесения атаки к той или иной категории. При таком подходе возникает нечеткость терминологии. С тем, чтобы не возникало разночтения между понятиями классификация, классификационная схема и рядом других, в дальнейшем будем использовать один термин — таксономия. Слово таксономия имеет греческое происхождение: $\tau\alpha\xi\iota\nu o\mu\iota\alpha$ (taxinomia) происходит от греческого taxis — order (порядок) и nomos — law (закон) [5], [6]. Строгое современное определение, которое используется в данной работе, можно найти в [7]: *«Таксономия — классификационная схема, которая разделяет совокупность знаний и определяет взаимосвязь частей».* Примером таксономии является известная таксономия растений и животных, предложенная шведским натуралистом Карлом Линнеем (Carolus Linnaeus) [8].

Ниже в данной работе будут описаны другие, менее известные таксономии, относящиеся непосредственно к классификации компьютерных атак. Автором выбрано несколько наиболее представительных таксономий, на примере которых показано развитие идей и подходов к решению рассматриваемой задачи.

Перед тем, как перейти к их описанию, рассмотрим подходы к разработке/формулировке критериев оценки таксономии. Для того, чтобы таксономия была пригодна для решения описанных ранее основных задач, она должна удовлетворять некоторым естественным и разумным требованиям.

Списки таких требований были изложены во многих работах по классификации атак ([4], [9], [10], [11], [12], [13], [14], [15], [16]), ниже приведен объединенный список, полученный на основе анализа этих работ. Эти требования не являются абсолютно четкими, и в полной мере удовлетворить всем им практически невозможно. Как будет показано ниже, на практике таксономия в большей мере является некоторым компромиссом между ними. Тем не менее, такие требования помогают указывать на достоинства и недостатки рассматриваемых таксономий, и этот факт является основной причиной, по которой их объединенный перечень приведен в данной работе.

- Взаимное исключение ([4], [9], [10], [11], [13], [14]).

  Таксономия должна быть устроена таким образом, чтобы выбор одной категории исключал все остальные. Иными словами, категории таксономии, как атрибуты/идентификаторы множеств, состоящих из относящихся к ним атак, не пересекаются. Это требование необходимо, чтобы имело смысл понятие «класс атаки».

- Полнота ([9], [10], [11], [12], [13], [14]).

  Таксономия покрывает собой все возможные атаки и позволяет их классифицировать.

  Вполне естественное требование того, чтобы таксономия покрывала всю область компьютерных атак, а не лишь какую-то ее часть.

  Объединяя два этих требования, можно сказать, что категории классификации должны образовывать разбиение множества атак.

- Детерминированность ([9], [10], [13], [15]).

  Необходимым условием является тот факт, что сама процедура (или классификационная схема), с помощью которой можно классифицировать атаки, должна быть четко определена.

- Четкость терминов (terms well defined) ([9], [10], [16]).

  Все термины, используемые в таксономии должны быть четко определены и пояснены с тем, чтобы не возникало непонимания или разночтения в понимании того или иного термина.

- Объективность (objectivity) ([9], [13], [15]).

  В таксономии должны рассматриваться только те сведения об атаке, которые могут быть получены исходя из свойств объекта в результате беспристрастного наблюдения.

- Применимость (useful) ([4], [10], [9], [11], [12], [13], [14]).

  Таксономия должна представлять собой систему, которую можно использовать для получения информации об поле исследования. Например, исходя из класса атаки, можно получить конструктивную информацию о ней самой.

- Понятность (comprehensible) ([9], [13], [14]).

  Таксономия должна быть доступна для понимания отдельных лиц, не являющихся экспертами в области информационной безопасности.

- Недвусмысленность (unambiguous) ([4], [9], [11], [12], [13], [14]).

  Каждая категория должна быть определена настолько четко, чтобы была однозначность в отношении того, к какой из категорий данная атака должна быть отнесена.

- Согласованность (conforming) ([9], [10], [13], [14]).

  Терминология, используемая в таксономии, должна быть согласованна с общепринятой терминологией в области информационной безопасности.

- Повторяемость результатов (repeatable) ([4], [9], [10], [11], [12], [13], [15]).

  Это требование означает, что при классификации одного и того же объекта двумя разными лицами должен получаться один и тот же результат.

Однако, как можно заметить, некоторые требования из перечисленного перечня пересекаются по смыслу, некоторые являются следствием других. По этой причине представляется разумным такие требования совместить или, соответственно, удалить. Более того, требования по смыслу можно разделить на две группы: основные требования, как требования непосредственно к смыслу и структуре вводимых категорий, и второстепенные, относящиеся скорее к форме изложения таксономии. После внесения описанных изменений список требований примет следующий вид.

Основные требования: взаимное исключение; полнота; применимость; детерминированность; объективность; расширяемость.

Второстепенные требования: четкость терминов; доступность/понятность; согласованность.

Отметим, что к основным требованиям добавлено еще одно, новое требование — расширяемость (или возможность расширения). Это требование того, чтобы строение таксономии, во-первых, допускало возможность добавления новых категорий, а, во-вторых, чтобы они органично в нее встраивались, а именно — для их внесения требовались бы минимальные изменения основного каркаса таксономии. По мнению автора, оно является немаловажным требованием к таксономии в силу того, что компьютерный мир развивается очень динамично, постоянно появляются новые технологии, и, как следствие, новые способы и технические средства проведения атак. По этой причине невозможно разработать таксономию (достаточно детальную), которая бы не требовала доработок и изменений с течением времени.

2 Анализ существующих работ и возможные подходы

Существует несколько подходов к проблеме классификации кибератак. Традиционно атаки делят на категории в зависимости от эффекта, который они производят: нарушение конфиденциальности информации, нарушение целостности информации и отказ в обслуживании (нарушение доступности информации) [11], [18]. Основным недостатком такого деления, является его слабая информативность (и, как следствие, применимость), так как по информации о классе атаки мы практически ничего не можем сказать о ее особенностях. Однако, разумеется, эффект атаки является важным ее свойством и этот параметр, в том или ином виде, используется во многих таксономиях ([4], [10], [13]).

Другим подходом к классификации является классификация уязвимостей аппаратного и программного обеспечения электронно-вычислительных систем. Одной из первых работ в этом направлении является работа Атанасио, Маркштейна и Филлипса [19]. Частично деление по типу уязвимости было использовано Ховардом и Лонгстаффом в [4]. Далее этот подход получил продолжение, и в работе [20] развита уже достаточно подробная классификация уязвимостей. Однако этот подход является слишком узким и зачастую не отражает характер атаки, поэтому применяется в основном лишь для специальных классов задач (например, при тестировании программного обеспечения).

Одним из возможных вариантов является деление исходя из начального доступа, которым обладает атакующий. Самый, пожалуй, известный пример подобного подхода — это матрица Андерсона [17]. В своей работе Джеймс Андерсон (James P. Anderson) предложил положить в основу классификации возможность, либо невозможность доступа атакующего к компьютеру или к его компоненту. Таким образом, категория, к которой принадлежит атака, зависит от того, какими начальными привилегии обладал атакующий. Таким образом, можно составить следующую матрицу 2×2:

	Атакующий не имеет право запуска/использования программы/информации	Атакующий имеет право запуска/использования программы/информации
Атакующий не имеет доступ к компьютеру	*Категория A* Внешнее вторжение	—
Атакующий имеет доступ к компьютеру	*Категория B* Внутреннее вторжение	*Категория C* Злоупотребление полномочиями

Из приведенной таблицы можно заметить, что все атаки разбиваются на три категории, так как случай, когда атакующий не имеет доступа к компьютеру (такой доступ ему не разрешен доступ к компьютеру), однако при этом ему разрешено использовать хранящееся на компьютере данные и запускать программы, невозможен. Категория B подразделяется Андерсоном еще на 3 подкатегории, в зависимости от атакующего. Таким образом, полный список категорий имеет следующий вид.

Таблица 1: Категории техник вторжения

1	Внешнее
2	Аппаратное
3	Маскировка
4	Вредоносные программы
5	Обход механизмов безопасности
6	Активное злоупотребление
7	Пассивное злоупотребление
8	Инертное злоупотребление (Inactive misuse)
9	Косвенное злоупотребление

A) внешнее вторжение;

B) внутреннее вторжение;

(i) ложный пользователь (masquerader);

(ii) легальный пользователь (legitimate user);

(iii) скрытый пользователь (clandestine user);

C) злоупотребление полномочиями.

Отличие между ложным пользователем, легальным пользователем и скрытым пользователем состоит в том, что ложный пользователь маскируется под легального пользователя и, например, с точки зрения системы, не отличим от него. Тайный же пользователь действует так, чтобы остаться незамеченным механизмами обнаружения или каким-либо образом избегает их.

К примеру, если атакующий смог узнать пароль легального пользователя и воспользовался им для получения доступа, то он действовал как ложный пользователь. Если он подменил часть системных файлов для получения доступа, то он действовал как скрытый пользователь.

Прослеживая дальнейшее развитие подходов можно заметить, что некоторые авторы в своих работах попытались не отталкиваться от каких-либо свойств и параметров атак, а составить общий список типов атак. Самая известная работа представленная на этом направлении — это работа Ноймана и Паркера ([21], [22], [23], [24]). Такие же, в целом, идеи были использованы Симоном Хансманом в работе [10], подробнее о которой будет изложено ниже. Бесспорным достоинством подхода, основанного на выделении списка типовых атак, является хорошее соответствие требованию применимости, так как в большинстве случаев тип атаки дает существенно больше информации нежели знание каких-либо ее свойств. Однако область его применения весьма ограничена в силу того, что при его использовании очень трудно удовлетворить первым двум очень немаловажным требованиям — полноте и взаимному исключению. В этой связи зачастую такие списки содержат сильно пересекающиеся категории атак и вопрос об их полноте также остается открытым.

В работе Питера Ноймана и Дональда Паркера (Peter Neumann, Donald Parker) представлены 9 категорий техник вторжения (таблица 1). На их основе Нойман [22] разработал 26 типов атак, изображенных на таблице 2.

В силу изложенных выше соображений наиболее перспективным является комбинированный подход, сочетающий себе в какой-то мере все вышеописанные методы. Примером такого подхода являются работы [4], [10], [13]. Однако, следует заметить что, способы комбинирования могут быть различными.

Первый способ — это разнести отдельно все анализируемые параметры и считать их независимыми. Такой подход был реализован в работа Хансмана, где автор использует так называемую концепцию «измерений», основная идея которой состоит в том, что свойства атаки расслаиваются на несколько независимых измерений, в каждом из которых есть свой список (или дерево) категорий.

В своей работе Симоном Хансманом (Simon Hansman, [10]) предложена таксономия сетевых и компьютерных атак, в основе которой лежит способ разделения параметров атаки на несколько измерений. Автором предложено четыре основных измерения и несколько вспомогательных.

- Первое измерение — это список типов атак (например, отказ в обслуживании).

Таблица 2: Типы атак

Внешнее	
Визуальное наблюдение	Наблюдение за клавиатурой или монитором
Обман	Обман операторов и пользователей
Извлечение мусора	Извлечение информации из виртуальных корзин
Аппаратное (hardware)	
Логическое восстановление	Извлечение информации с выброшенных или украденных носителей
Прослушивание	Перехват данных
Вмешательство	
Физическая атака	Разрушение или повреждение оборудования, источников питания
Физическое удаление	Изъятие оборудования и хранилищ данных
Маскировка	
Имитирование	Использование ложных идентификаторов
Узурпирование линий связи или хостов	
Атаки с подменой параметров	
Спутывание сетей	Маскировка физического месторасположения или маршрута
Вредоносные программы	*Создание возможности дальнейших злонамеренных действий*
Троянские кони	Внедрение вредоносного кода
Логические бомбы (Logic bombs)	Разновидность троянских коней
Черви	Овладение распределенными ресурсами
Вирусы	Прикрепление к программам и размножение
Обход	Обход механизмов безопасности
Эксплуатация уязвимостей	
Взлом паролей	
Активное злоупотребление	
Основной	
Инкрементальные атаки	Постепенная эскалация привилегий, медленное продвижение к цели
Отказ в обслуживании	Совершение массивных атак
Пассивное злоупотребление	
Обзор	Случайный или выборочный поиск
Сбор и вывод данных	Использование баз данных и анализ трафика
Скрытые каналы	Использование скрытых каналов или другие способы утечки информации
Инертное злоупотребление	
Косвенное злоупотребление	

- Второе измерение — это цель (целевой объект) атаки. Если у атаки несколько объектов нападения, то в этом измерении должно присутствовать несколько записей.
- Третье измерение — это уязвимости, используемые в процессе атаки. По словам автора, это измерение обычно содержит CVE-описания уязвимостей (Common Vulnerabilities an Exposures), причем если используется несколько уязвимостей, то в этом измерении присутствует несколько записей.
- Четвертое измерение — это, фактически, результат или цель атаки.

Опишем все эти измерения более подробно.

Первое измерение		
Вирусы:	Инфицирующие файлы	
	Инфицирующие системные/загрузочные сектора	
	Макровирусы	
Черви:	Использующие массовую рассылку	
	Распознающие состояние сети	
Троянские программы:	Логические бомбы	
Переполнение буфера:	Переполнение стека	
	Переполнение кучи	
Отказ в обслуживании:	Локальный (host based):	Исчерпание ресурсов
		Вывод из строя
	Сетевой (network based):	TCP-флуд
		UDP-флуд
		ICMP-флуд
	Распределенные	
Сетевые атаки:	Подмена пакетов	
	Перехват сессии	
	Беспроводные атаки:	Взлом криптоалгоритмов беспроводных сетей
	Атаки на веб-приложения:	Использование злоумышленных web-сценариев (Cross Site Scripting)
		Подбор параметров
		Использование некорректных cookies
		Атаки на базы данных
		Использование скрытых полей
Физические атаки:	Простые	
	Энергетическое оружие:	HERF
		LERF
		EMP
	Van Eck	
Атаки на пароли	Угадывание:	Атака методом грубой силы
		Атака по словарю
	Использующие уязвимость в реализации	
Атаки — сбор информации	Прослушивание	Прослушивание пакетов
	Выявление структуры сети	
	Сканирование	

Второе измерение					
Аппаратное обеспечение:	Компьютер:	Жесткие диски			
		...			
	Сетевое оборудование:	Хаб			
		Кабель			
		...			
	Периферийные устройства:	Монитор			
		Клавиатура			
		...			
Программное обеспечение:	Операционная система:	Семейство Windows:	Windows XP		
			Windows 2003 Server		
			...		
		Семейство Unix:	Linux:	2.2	
				2.4	
				...	
			Free BSD:	4.8	
				5.1	
				...	
			...		
		Семейство Mac OS:	Marco's X:	10.1	
				10.2	
				...	
			...		
		...			
	Приложение:	Серверное приложение:	База данных		
			Почтовый сервер:		
			Веб-сервер:	IIS:	4.0
					5.0
			...		
		Пользовательское приложение	Текстовый редактор:	MS Word	2000
					XP
					...
			Почтовый клиент:	...	
			...		
		...			
Сеть:	Протокол:	Транспортный уровень:	IP		
			...		
		Сетевой уровень:	TCP		
			...		
		...			

Третье измерение

Как уже было отмечено выше, третье измерение содержит в себе стандартные описания уязвимостей, используемых в атаке. В этой связи для него необходимость описания какой-либо общей схемы как для первых двух отпадает.

Четвертое измерение

Четвертое измерение служит для классификации атак, осуществляемых не только ради своей глав-

ной цели. Например, червь помимо своего прямого назначения — заражения компьютера, может служить для удаленного управления или уничтожения каких-то файлов. Четвертое измерение состоит из пяти категорий:

- непосредственная (номинальная) цель атаки;
- нарушение целостности информации;
- нарушение конфиденциальности информации;
- захват ресурсов;
- получение контроля над частью системы для дальнейшего использования.

Другие измерения

Другие измерения, как пишет Хансман, могут быть добавлены для улучшения и развития таксономии. В качестве вариантов дальнейшей детализации предлагаются следующие категории:

- Ущерб, который описывает ущерб, нанесенный атакой.
- Стоимость восстановления, которая описывает общую стоимость восстановления системы после атаки до первоначального состояния.
- Распространение, которое описывает скорость и способ распространения атаки. Эта категория подходит больше всего для размножающихся атак наподобие вирусов и червей.
- Способы защиты.

Второй способ основан на той же идее, однако является более гибким, так как подразумевает древовидную структуру категориальных классов с самого высокого уровня. Реализацию этого подхода можно найти в [13].

В своей работе Джеффри Андеркоффер и Джон Пинкстон (Jeffrey Undercoffer, John Pinkston) придерживаются структурного подхода к классификации. В графическом представлении (рисунок 1) разработанная ими таксономия представляет собой дерево, корнем которого является вторжение. Ребра этого дерева имеют метки несущие смысловую нагрузку: например, из корня дерева выходят два ребра, означающие «осуществлено с помощью» и «имело результат». Несплошная стрелка выражает отношение «является подклассом» между вершинами, которые она соединяет и, для того чтобы излишне не нагружать рисунок, эта надпись опущена. Можно заметить, что применение таких способов хотя и дает более детальное описание атаки, однако не может отразить некоторые ее структурные особенности, сценарий атаки. И это обстоятельство является существенными недостатком, учитывая постоянное совершенствование современных систем защиты и, как следствие, тенденции к все более сложным и изощренными методам атак [25], [26], [27].

Третий способ — это комбинирование свойств с внесением структуры. Этот подход был применен в работе [4], основная его идея состоит в том, что вводится иерархия понятий: основным в данной работе считается понятие инцидент, в него включается понятие атака, а в понятие атака включается понятие действие. При этом инцидент может состоять из нескольких нескольких атак, а каждая атака из нескольких действий.

Джон Ховард и Томас Лонгстафф (John D. Howard, Thomas A. Longstaff) не только создали таксономию, но и разработали «Общий язык для инцидентов в области компьютерной безопасности». Как сказано во вступлении к [4], «Этот общий язык не является попыткой создания всеобъемлющего словаря терминов в области компьютерной безопасности. Вместо этого, мы создали минимальный набор высокоуровневых терминов, вместе со структурой, отражающей их взаимосвязь (таксономией)».

Таксономия, разработанная этими авторами, представляет собой следующую диаграмму (рис. 2). Основным понятием в данной таксономии является понятие «инцидент», так как авторы разделяют два понятия — инцидент и атака. В понятие инцидент входит атакующий, атака и цель атаки. Под атакой понимаются те сущности, которые относятся непосредственно к процессу совершения атаки: инструмент, уязвимость, действие, целевой объект и несанкционированный результат. Инструмент — это средство, которое использовал атакующий при нападении. Совокупность действия и целевого объекта называется событием.

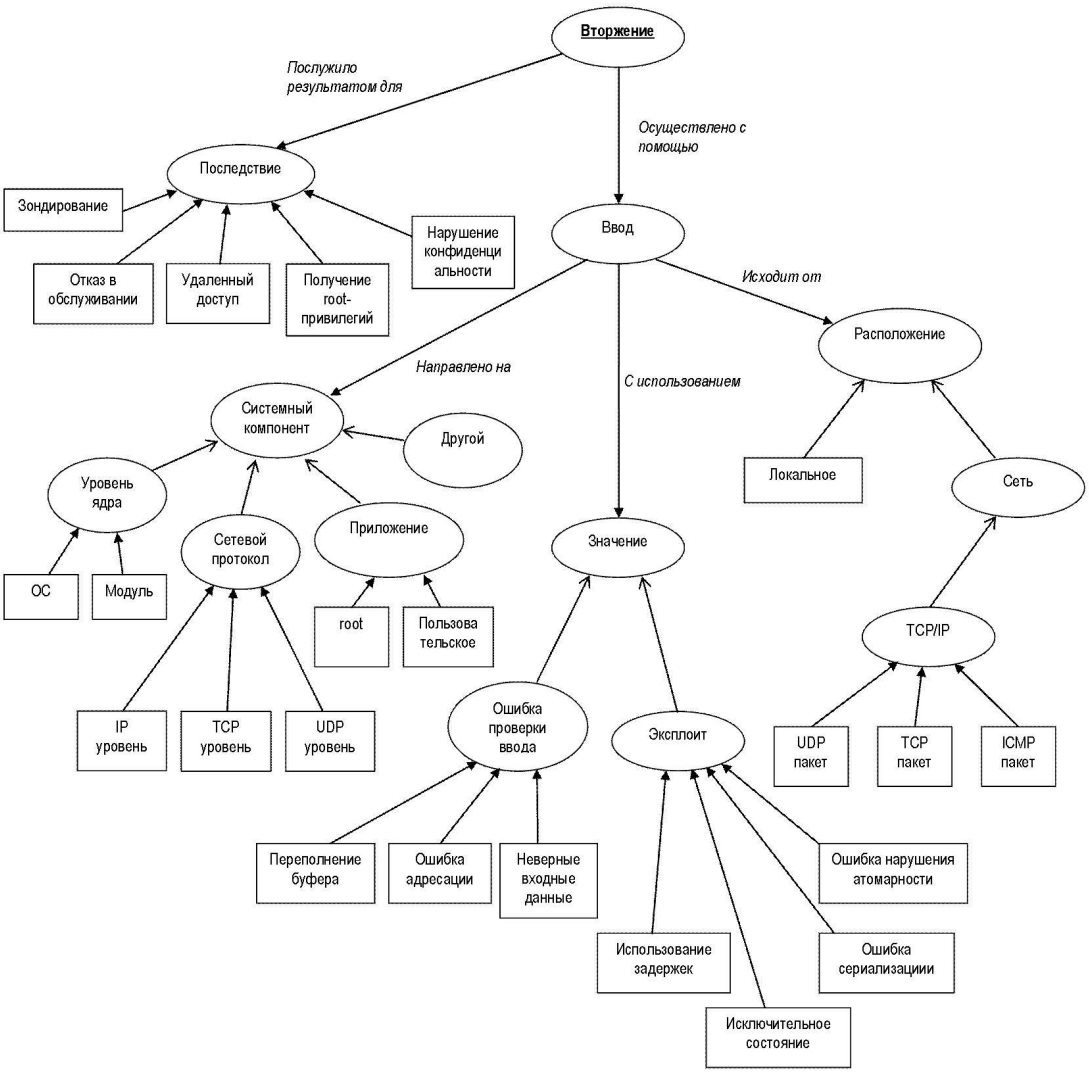


Рис. 1: Вторжение

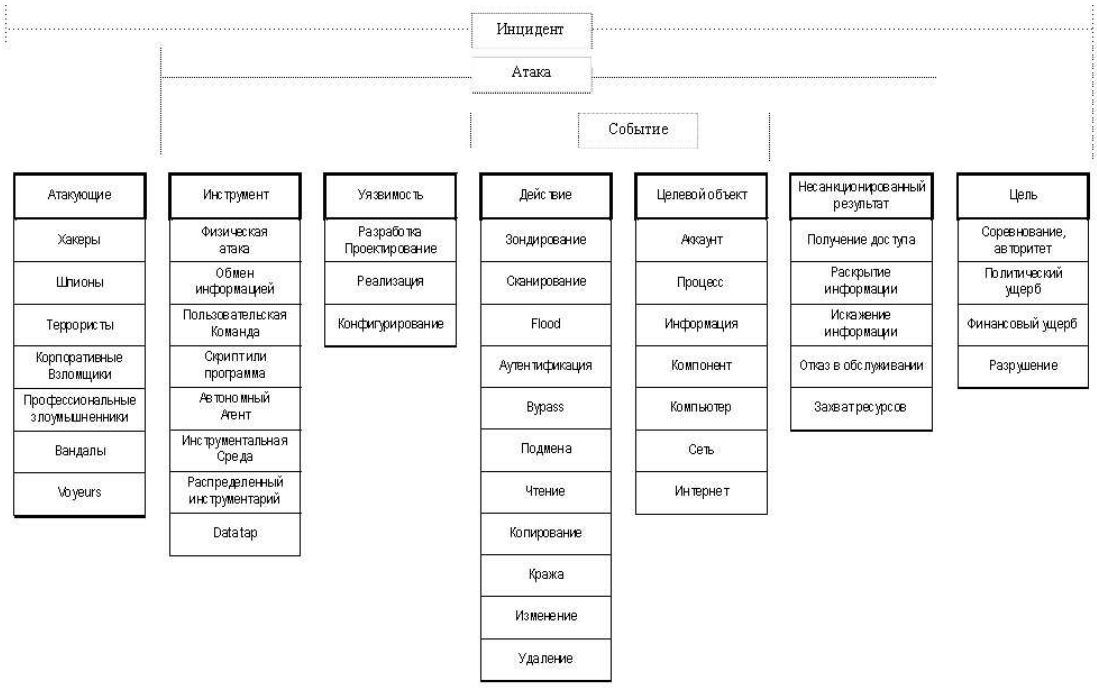


Рис. 2: Инцидент

Предложенная авторами таксономия отличается от описанных выше тем, что в ней присутствуют структурные элементы: инцидент, атака, событие и заложена возможность комбинирования этих событий. Так в одном инциденте может быть вложена последовательность атак. Данное свойство в какой-то мере позволяет описывать неатомарные (многоходовые) составные атаки и учитывать их сценарий.

3 Предлагаемая таксономия

В предлагаемой автором таксономии развивается комбинированный подход к решению задачи классификации. Однако в отличие от предыдущих работ вводится иерархическая структура отношений с древовидным раскрытием категорий. Как самостоятельный отдельный объект вводится важное понятие «этап атаки» что позволяет, в отличие от предыдущий подходов, довольно естественным образом описывать многоэтапные атаки. На рисунке 3 приведена общая схема атаки. Атака может состоять из нескольких этапов, этап, в свою очередь, из нескольких действий, действие — из нескольких событий. К примеру, взлом через pro-ftpd ([28], [29]) может являться частью одного из этапов атаки и состоит из четырех событий, которые могут идти в различном порядке.

Кроме вложенности в понятие более высокого уровня, каждое из этих четырех понятий раскрывается с помощью дерева подкатегорий, то есть имеет свой собственный набор атрибутов.

3.1 Атака

Понятием самого верхнего уровня является *атака.* Это понятие имеет такие атрибуты, как глобальная цель/результат, свойства атаки, объект атаки и атакующий. Каждый из перечисленных атрибутов тоже имеет свои атрибуты и является поддеревом дерева атрибутов атаки. Важно отметить, что цель разделяется на две компоненты: информационную составляющую и социальнозначимую составляющую. Информационная составляющая отражает информационный аспект последствий воздействия атаки на систему: нарушение конфиденциальности информации (которая подразделяется на нарушения конфиденциальности с целью разведки, либо с целью разглашения), нарушение доступности информации/ресурсов системы (которая подразделяется на нарушения с целью блокирования системы

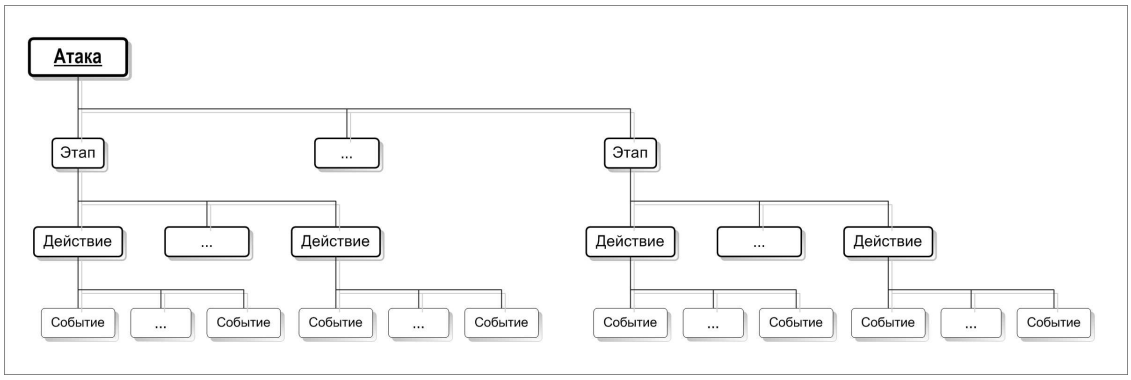


Рис. 3: Общая структура атаки

защиты либо с целью нарушения функционирования самой системы) и нарушение целостности информации (с целью внедрения и получения контроля). Социальнозначимая составляющая, в отличие от информационной, отражает внеинформационные аспекты последствия атаки. Проиллюстрируем такие последствия на примере захвата компьютеров информационно-вычислительной среды атомной электростанции и проведение теракта с целью создания техногенной катастрофы путем выведения из строя реактора. В данном случае, информационной составляющей цели является захват компьютеров, а социальнозначимой — создание чрезвычайной ситуации посредством выведения из строя реактора.

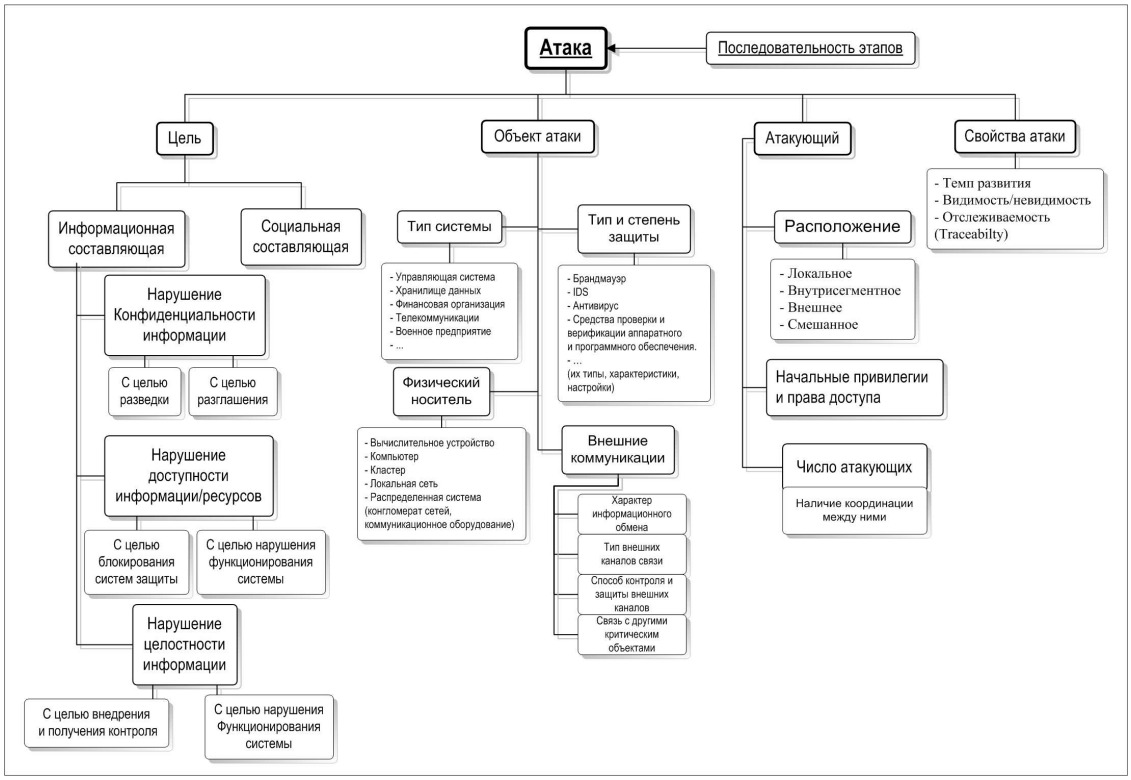


Рис. 4: Атака

Другим важным атрибутом является объект атаки, так как атаки на объекты разной функциональности и категорийности имеют, как правило, разный характер. Здесь выделяются такие свойства объекта атаки, как тип атакуемой системы, ее физический носитель (оборудование, которое формирует информационно-вычислительную среду системы), тип средств защиты, используемых в системе и степень защищенности (уровень жесткости правил безопасности) и внешние коммуникации системы.

Еще один атрибут атаки — это атакующий. Основными свойствами, которые его характеризуют

являются расположение относительно системы, начальные привилегии и права доступа. Если атакующих несколько, то в этом случае возникает враждебная многоагентная система [30], [31], [25], поэтому становится крайне важным их число, наличие и характер координации между атакующими.

Из перечисленных свойств, пожалуй, наибольшее значение имеет расположение атакующего относительно объекта атаки. Атакующий может осуществлять атаку с того же компьютера, на котором находится информация, которая является его целью. Примером локальной атаки может быть повышение привилегий с помощью переполнения буфера в одной из программ, исполняющих часть операций в привилегированном режиме, и получения доступа к данным. Пример внутрисегментной атаки — это атака, когда атакующий начинает атаку с компьютера находящегося в одном сегменте сети, что позволяет ему использовать эксплоиты для сервисов, порты которых фильтруются извне межсетевым экраном и таким образом захватить компьютер-жертву. Внешняя атака — это атака, проводимая атакующим удаленно, например, атака суперкомпьютерного центра Сан-Диего, описанная в [32]. Очень хорошо и подробно методы реализации данного типа атак описаны и разобраны в [33]. Кроме отмеченных выше существует и смешанный тип атак, которые обычно проводятся согласованно группой атакующих. Примером может служить атака, описанная в [35] , проводимая двумя пользователями с помощью образования скрытого канала. Суть состоит в том, что один пользователь находится внутри сегмента сети и каким-либо образом получает доступ к необходимой информации и передает ее другому пользователю вовне с помощью скрытого канала. Атаку может проводить несколько атакующих из разных мест, в этом случае атака называется распределенной по атакующим (о чем говорит параметр «число атакующих»), простейший пример — DDoS-атака. Более подробную информацию о возможных типах таких атак можно найти в [36].

Последним атрибутом, представленным на диаграмме 4, является атрибут «свойства атаки». Для уменьшения риска быть обнаруженной атаки иногда длятся по несколько месяцев, а других случаях, несколько секунд (чтобы, например, исключить возможность вмешательства администратора атакуемой системы), В силу этих обстоятельств темп развития атаки — немаловажное для классификации свойство атаки. Другие два свойства (упомянутые в [34]) — видимость и возможность проследить источник атаки. Видимость означает, что сценарий атаки разработан так, что предполагается, что во время проведения атака не будет обнаружена средствами обнаружения. Отслеживаемость означает, что после проведения атаки при проведении расследования существует возможность проследить источник атаки. Следует отметить, что эти два свойства сильно связаны друг с другом. Значение каждого из них зависит, в первую очередь, от поставленной злоумышленником цели, и они сильно влияют на выбор стратегии, используемой при атаке. Например, если задача злоумышленника незаметно проникнуть в систему и выкрасть конфиденциальную информацию, то при выборе стратегии он может вполне использовать сценарии, которые невидимы, однако прослеживаемы (к примеру, редактирование или стирание лог-файлов делает атаку существенно более заметной, но менее прослеживаемой).

3.2 Этап

Атака состоит из этапов, которые, в свою очередь, тоже имеют свои атрибуты. Понятие этап отражает (см. рисунок 5 отдельную часть атаки, имеющую свою локальную цель. Приведем пример: одним из этапов атаки может быть этап-разведка — сканирование подсетей какой-нибудь атакуемой организации. Цель этого этапа — по возможности незаметно, не вызывая подозрений исследовать топологию и внутреннее устройство сетевого сегмента объекта атаки для нахождения слабых мест системы защиты и последующего вторжения. Для достижения этой цели существует значительное количество различных и довольно нетривиальных способов, подробнее о которых изложено в [33].

3.3 Действие

Этап состоит из действий. Действие представляет собой, в некотором смысле, «атомарную» атаку (например, сканирование портов или использование программных уязвимостей). Действие тоже обладает атрибутами: тип действия, субъект, объект, последствия действия, результат. Фактически, оно является минимальными смысловым шагом атаки.

Рассмотрим атрибуты этого понятия (рис. 6). Атрибут «тип действия» описывает непосредственно само действие, происходящее на данном этапе атаки. Этот атрибут является наиболее важным и информативным. В некотором смысле, список возможных действий похож на перечень типовых атак ([4], [22], [24]), По этой причине ему тоже свойственно отсутствие полноты и, возможно, при

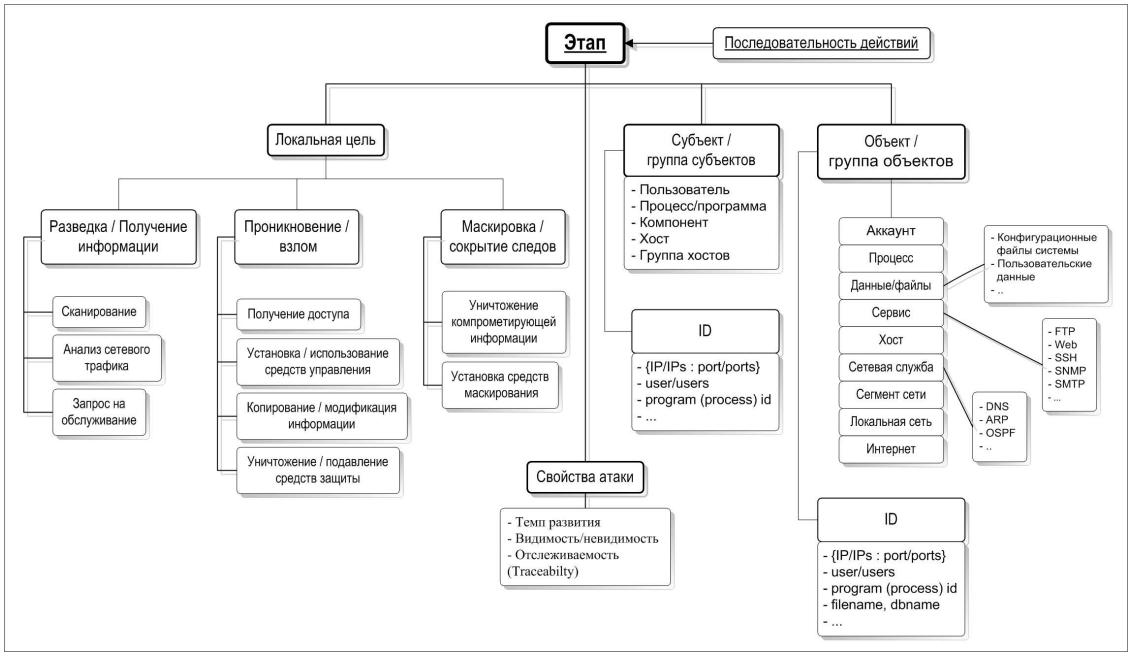


Рис. 5: Этап

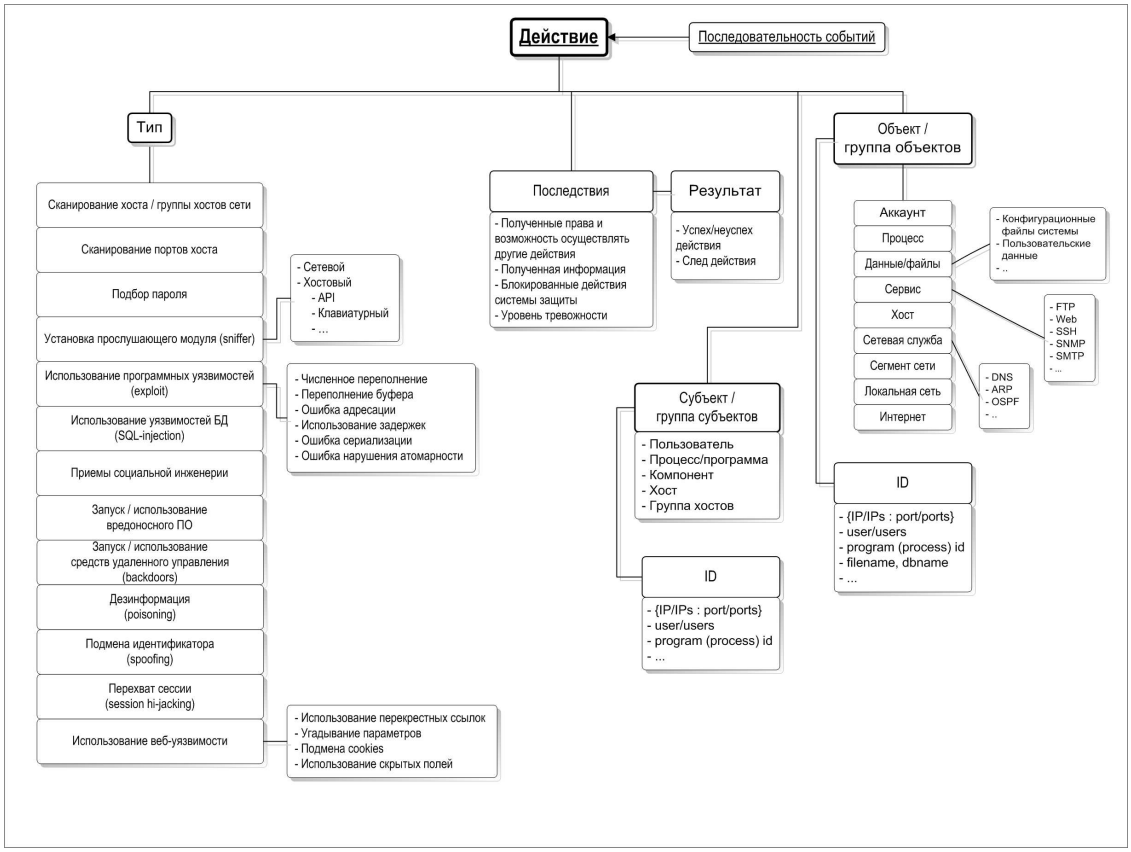


Рис. 6: Действие

применении таксономии на практике, представленный на рисунке список потребует расширения (в зависимости от специфики области применения). «Объект/группа объектов» — это то (программа, компьютер или сеть), на что данное действие направлено. «Субъект/группа субъектов» — это то, что производит данное действие. К примеру, если при взломе сети атакующему удалось захватить один из хостов сети (хост A), и дальше он производит сканирование портов другого хоста (назовем его B) от имени захваченного компьютера, то субъектом действия будет хост A, а объектом — хост B. Параметр «последствия» характеризует последствия действия, а именно, полученные атакующим права и привилегии в объекте атаки, информацию, к которой он получил доступ в результате этого действия и тому подобное. Этот параметр содержит также информацию об уровне тревожности данного действия (безусловно, уровень тревожности является весьма субъективной величиной и сильно зависит от предыстории и от окружения, в котором происходит действие, поэтому здесь имеется в виду некоторая априорная шкала оценок тревожности).

3.4 Событие

Заметим, что с точки зрения системы, действие далеко не атомарно. Сканирование портов, например, это цепочка действий, которая может сильно варьироваться (например, [38], [37]). По этой причине, если настолько детально рассматривать атаку, то необходимо ввести еще один, самый нижний уровень абстракции — уровень событий.

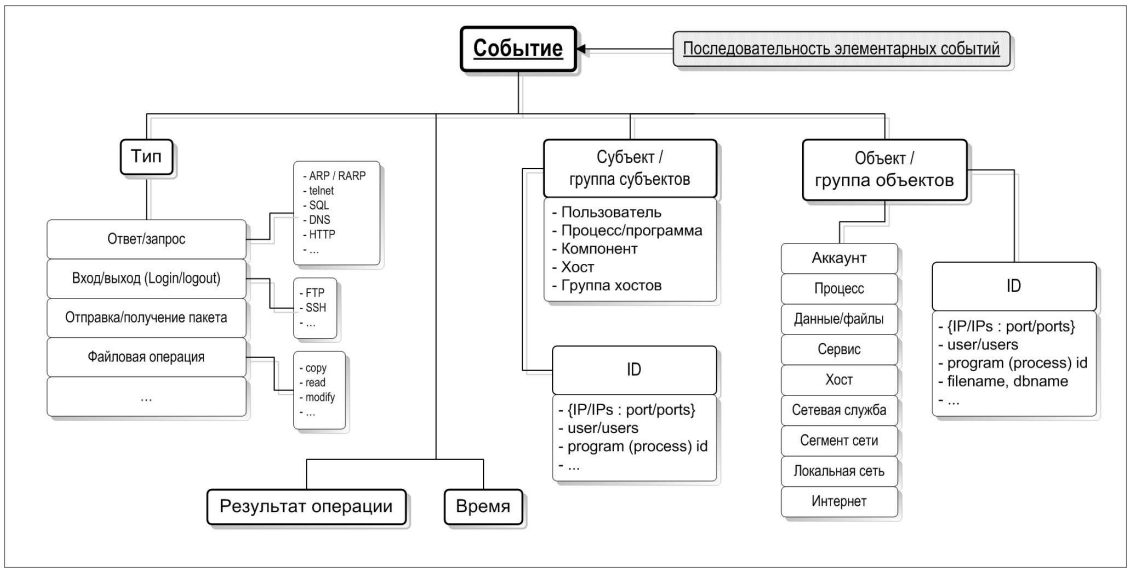


Рис. 7: Событие

Событием назовем минимальный (на заданном уровне детализации) шаг с точки зрения системы. Однако, вообще говоря, событие не входит в таксономию, так как в большинстве случаев это лишь усложняет понимание и увеличивает объем, и не несет в себе какой-либо полезной информации. По этой причине вводить этот уровень рекомендуется лишь тогда, когда необходимо детальное описание (модель атаки), скажем в случае дальнейшего формального анализа автоматизированными средствами.

Заключение

В статье рассмотрены существующие подходы к решению задачи классификации компьютерных атак, выделены направления, на которые можно их разделить. Во введении обоснована актуальность решаемой проблемы, рассмотрены возможные области применения таксономии. Также сформулирован ряд требований, которым должна удовлетворять таксономия атак для того, чтобы ее было удобно применять на практике.

Далее в разделе 2 описаны некоторые известные таксономии, проведен анализ достоинств и недостатков каждого из подходов, соответствие сформулированным требованиям. На основании полученного материала предложен подход к решению проблемы (раздел 3). В начале описана общая схема таксономии и способы устранения основных недостатков предыдущих работ. Далее детально изложены все составляющие таксономии: атака, этап, действие, событие и приведены диаграммы каждой из них с пояснением используемых терминов.

Литература

[1] CERT/CC Statistics 1988–2005.
`http://www.cert.org/stats/cert_stats.html`.

[2] *Проблемы математического, алгоритмического и программного обеспечения компьютерной безопасности в Интернет*, В. А. Васенин, Материалы конференции «Математика и безопасность информационных технологий-2003», Москва, 2003, с. 111–142.

[3] *Математическое и программное обеспечение активного аудита больших распределенных систем*, В. А. Васенин, А. В. Галатенко, В. В. Корнеев, А. А. Макаров, Материалы конференции МаБИТ-2004, Москва, 2004, с. 99–117.

[4] *A common language for computer security incidents*, John D. Howard, Thomas A. Longstaff, Sandia Report, Sandia National Laboratories, 1998.

[5] Wikipedia, free Internet encyclopedia.

[6] *Большая советская энциклопедия*, изд. Советская энциклопедия, 1969–1978 гг.

[7] *The IEEE standard dictionary of electrical and electronics terms, Sixth edition*, John Radatz, editor, Institute of Electrical and Electronics Engineers, New York, 1996

[8] *Systema Naturae per Regna Tria Naturae, Secundum Classes, Ordines, Genera, Species, cum Characteribus, Differentis, Synonymis, Locis.* n/a, editio duodecima, reformata edition, 1766. Tomus I, Regnum Animale, 1766; Tomus II, Regnum Vegetabile, 1767; Tomus III, Regnum Lapideum, 1768. Carolus Linnaeus.

[9] *A taxonomy of computer attacks with applications to wireless networks*, Daniel L. Lough, Ph. D. dissertation, 2001.

[10] *A taxonomy of network and computer attacks methodologies*, Simon Hansman, University of Canterbury, New Zealand, November 2003.

[11] *Fundamentals of Computer Security Technology*, Edward Amoroso, P T R Prentice Hall, New Jersey, 1994.

[12] *An analysis of security incidents on the internet 1989-1995*, John D. Howard, PhD thesis, Carnegie Mellon University, 1997.

[13] *Modeling computer attacks: a target-centric ontology for intrusion detection*, Jeffrey Undercoffer and John Pinkston, University of Maryland Baltimore Country.

[14] *How to systematically classify computer security intrusions*, Ulf Lindqvist, Erland Jonsson, Chalmers University of Technology, Sweden, 1997.

[15] *Software vulnerability analysis*, Ivan Victor Krsul, PhD thesis, Purdue University, 1998.

[16] *A critical analysis of vulnerability taxonomies*, Matt Bishop, David Bailey, University of California, Davis, September 1996.

[17] *Computer security threat monitoring and surveillance*, James P. Anderson, Technical Report Contract 79F296400, Washington, April 1980.

[18] *Математические модели распределенных компьютерных систем*, В. А. Васенин, А. В. Галатенко, Материалы конференции МаБИТ-2004, Москва, 2004, с. 91–98.

[19] *Penetrating an operating system: a study of VM/370 integrity*, C. R. Attanasio, P. W. Markstein, and R. J. Phillips. IBM System Journal, 15(1):102–116, 1976.

[20] *Bug Taxonomies*, Giri Vijayaraghavan, Cem Kaner, STAR EAST 2003, Orlando, FL, May, 2003.

[21] *A summary of computer misuse techniques*, Peter Neumann, Donald Parker, In 12th National Computer Security Conference, 1989.

[22] *Computer-Related Risks*, Peter G. Neumann, ACM Press/Addison Wesley, 1995.

[23] *COMPUTER CRIME Criminal Justice Resource Manual*, Donald B. Parker, U.S. Department of Justice National Institute of Justice Office of Justice Programs, August 1989.

[24] *Computer Security Reference Book*, Donald B. Parker, chapter 34, Computer Crime, pages 437-476. CRC Press, K. M. Jackson and J. Hruskh, U.S. Associate Editor Donn B. Parker, Boca Raton, Florida, 1992.

[25] *Роль скрытых каналов при построении защиты в распределенных компьютерных системах*, А. А. Грушо, Е. Е. Тимонина, Материалы конференции «Математика и безопасность информационных технологий-2003», Москва, 2003, с. 276–283.

[26] *Реализация системы управления доступом к информации в виде встраиваемых модулей аутентификации*, А. В. Галатенко, А. А. Наумов, А. Ф. Слепухин, Материалы конференции «Математика и безопасность информационных технологий-2003», Москва, 2003, с. 237–240.

[27] *Обеспечение информационной безопасности систем на программной платформе ос2000*, В. Б. Бетелин, В. А. Галатенко, А. Н. Годунов, А. И. Грюнталь, Материалы конференции «Математика и безопасность информационных технологий-2003», Москва, 2003, с. 254–267.

[28] *Language-based generation and evaluation of NIDS signatures*, Shai Rubin, Somesh Jha, Barton P. Miller, University of Wisconsin.

[29] Beyond Security Inc. ProFTPD ASCII file remote root exploit. `http://www.securiteam.com/exploits/`.

[30] *Враждебные многоагентные системы*, А. А. Грушо, Е. Е. Тимонина, Материалы конференции МаБИТ-2004, Москва, 2004, с. 249–256.

[31] *Multi-agent technologies for computer network security: Attack simulation, intrusion detection and intrusion detection learning*, Владимир Городецкий, Игорь Котенко, Олег Карсаев, 2003.

[32] *Атака через Internet*, И. Д. Медведовский, П. В. Семьянов , В. В. Платонов.

[33] *Breaking into computer networks from the Internet*, Roelof Temmingh, 2001.

[34] *Building Computer Network Attacks*, Ariel Futoransky, Luciano Notarfrancesco, Gerardo Richarte, Carlos Sarraute, CoreLabs, Core Security Technologies, 2003

[35] *Representation and analysis of coordinated attacks*, Sviatoslav Bryanov, Murtuza Jadiwala.

[36] *A taxonomy of DDoS Attack and DDoS defense mechanisms*, Jelena Mirkovic, Peter Reiher, 2002.

[37] *A Comparison of Various Port Scanning Techniques*, Arpit Aggarwal, Ranveer Kunal, Indian Institute of Information Technology — Allahabad, India.

[38] *Examining port scan methods — Analysing Audible Techniques*, Synnergy Networks, 2001.

Обеспечение информационной безопасности в системе удостоверяющих центров

В. Д. Аносов, А. С. Логачёв, И. Г. Савастеев

1 Удостоверяющие центры

Одним из центральных элементов обеспечения достоверности информации, циркулирующей в системе ЭДО, является удостоверяющий центр (УЦ) цифровых сертификатов ключей. Важнейшими условиями обеспечения информационной безопасности УЦ являются научно-техническая оценка уровня информационной безопасности УЦ как объекта информатизации в процессе его создания и ввода в эксплуатацию и научно-технический контроль за состоянием информационной безопасности в процессе эксплуатации УЦ.

Основные функции удостоверяющего перечислены в п. 1 ст. 9 Закона об ЭЦП.

УЦ представляет собой сложный комплекс организационно-технических мероприятий и программно-аппаратных средств, при реализации которого на практике должен применяться весь комплекс организационных, программно-технических и нормативно правовых методов обеспечения информационной безопасности.

2 Программно-технические методы обеспечения информационной безопасности

Построение инфраструктуры УЦ можно разделить на несколько этапов:

- анализ возможного риска;
- реализация политики безопасности;
- поддержание построенной политики безопасности.

При этом, в частности, должны быть обеспечены:

- конфиденциальность хранимой, обрабатываемой и передаваемой по каналам связи информации (ключевая информация средств криптографической защиты информации, личная информация, охраняемая в соответствии с действующим законодательством, закрытые ключи удостоверяющего центра, парольная информация и информация аудита и др.);
- целостность хранимой и передаваемой информации (информация о владельцах, входящая в состав сертификатов, информация об отозванных сертификатах и др.);
- доступность информации;
- предотвращение утечки информации по побочным техническим каналам, а также за счет специально внедренных в помещения и технические средства электронных закладочных устройств.

Отметим, что при реализации работы защищенного удостоверяющего центра в совокупности выполняемых им функций должны присутствовать все эти составляющие.

3 Обеспечение конфиденциальности информации

Решение данной задачи базируется на:

- применении криптографических методов защиты информации;
- предотвращении утечки информации по побочным техническим каналам, а также за счет специально внедренных в помещения и технические средства электронных закладочных устройств.

Несмотря на то, что удостоверяющий центр может быть предназначен для использования в сети связи с обработкой только открытой информацией, даже в такой сети возникает необходимость обеспечения конфиденциальности информации, в частности, при хранении закрытого ключа подписи удостоверяющего центра, а также возможно при хранении по их просьбе закрытых ключей пользователей.

4 Обеспечение целостности информации

4.1 Рекомендации по использованию протоколов работы и криптографических примитивов удостоверяющих центров

В работе удостоверяющих центров используется значительное количество протоколов. Заметим, что помимо протоколов прикладного уровня в работе удостоверяющих центров широко используются и протоколы более низких уровней, такие как Transport Layer Security (TLS), Lightweight Directory Access Protocol (LDAP), Secure HyperText Transfer Protocol (HTTP-S), File Transfer Protocol (FTP) и ряд других. Анализ описания протоколов SSH и TLS показывает, что для его реализации требуются следующие криптографические примитивы:

- алгоритм генерации и проверки ЭЦП и, опционально, алгоритм открытого шифрования;
- алгоритм открытого распределения ключей (ОРК);
- хэш-функция, используемая для итеративной генерации ключевой информации; - Алгоритм симметричного шифрования;
- алгоритм аутентификации сообщений (MAC);
- алгоритм генерации псевдослучайных чисел.

Прикладные протоколы, используемые в удостоверяющих центрах, можно разбить на следующие два класса:

- основные протоколы (необходимы для работы удостоверяющего центра в целом, обеспечивают взаимодействие компонент удостоверяющего центра, взаимодействие удостоверяющего центра с пользователями, взаимодействие удостоверяющих центров в рамках инфраструктуры УЦ);
- дополнительные протоколы (обеспечивают работу дополнительных служб и сервисов удостоверяющего центра (таких как служба меток времени, служба онлайнового предоставления информации о статусе сертификата и т.п.).

Основными протоколами являются протоколы управления сертификатами CMP (Certificate Management Protocols), описанные в RFC 2510.

Анализ спецификаций протоколов CMP, рекомендованных в RFC 2510, показывает, что в них содержатся следующие криптографические компоненты:

- электронная цифровая подпись (ЭЦП);
- код аутентификации сообщения MAC (Message Authentication Code).

4.2 Определение текущего статуса сертификата

Особое значение в системах управления цифровыми сертификатами имеет проблема определения текущего статуса сертификата. Вариант проверки информации о статусе сертификата с использованием протокола OCSP заключается в том, что в системе управления открытыми ключами существует служба, которая на основе запросов пользователя в режиме реального времени предоставляет информацию о сертификате. Порядок данных взаимодействий рассматривается протоколом OCSP.

OCSP, предоставляя информацию о статусе сертификата, выступает как замена проверки на основе использования списков отозванных сертификатов (СОС). OCSP предоставляет возможность преодолеть ряд ограничений, присущих использованию СОС, таких как необходимость распространения обновленного СОС OCSP определяет формат сообщений-запросов и сообщений-ответов между приложением клиента, которому необходимо получить информацию о статусе отзыва сертификата и приложением сервера, который имеет базу данных статусов отзыва.

При рассмотрении вопросов обеспечения безопасности протокола OCSP необходимо учитывать основные применяемые технологии использования систем предоставления информации о статусе цифровых сертификатов открытых ключей, построенных на основе использования протокола OCSP. Существуют две основные технологии использования систем, реализующих работу по протоколу OCSP: Trusted OCSP - доверенный OCSP, - Distributed OCSP - распределенный OCSP.

4.3 Протокол меток времени

Рекомендуется использовать в обязательном порядке протокол «Меток времени» (Time-Stamp Protocol, (TSP), см. RFC 3161), позволяющий сопоставлять основные операции с цифровыми сертификатами и электронной цифровой подписью фиксированным моментам времени, что может использоваться при уточнении статуса сертификата, разрешении спорных ситуаций и т.п. Наличие службы меток времени в качестве одного из сервисов удостоверяющего центра позволяет уменьшить последствия возможной компрометации закрытого ключа.

4.4 Участие ЦР в протоколах и защите сообщений

Рекомендуется обмены сообщениями между центром сертификации (ЦС) и конечным пользователем (КП) осуществляются только через ЦР. Такой подход является более предпочтительным, поскольку обеспечивает более высокий уровень информационной безопасности ЦС. Криптографические примитивы. Описанные ранее протоколы генерации и проверки сертификатов нуждаются в обеспечении следующими криптографическими примитивами:

- симметричное шифрование и расшифрование;
- асимметричное шифрование и расшифрование;
- вычисление функции хеширования;
- выработки и проверки электронной цифровой подписи;
- открытое распределение ключей;
- генерации и проверки параметров схемы ЭЦП;
- аутентификации сообщений (MAC) на секретном ключе;
- генерации ключевой пары ЭЦП.

Имеются возможности реализации соответствующих алгоритмов при помощи отечественных криптосредств.

1. Алгоритм шифрования ГОСТ 28147-89 может быть использован для шифрования данных как один из параметров протокола записи, после установления связи и генерации ключевой информации.

2. Хэш-функции используются в TLS как составные элементы алгоритма ЭЦП, а также как составные блоки функций контроля целостности (HMAC) и генерации ключевой информации (PRF). В случае ЭЦП, при использовании алгоритма ГОСТ Р 34.10-2001 необходимо использовать хэш-функцию ГОСТ Р 34.11-94, как это определяется требованиями стандарта. Это может быть обеспечено введением единого идентификатора алгоритма, описывающего ЭЦП ГОСТ Р 34.10-2001. В случае псевдослучайной функции ситуация является несколько более сложной, так как описание TLS явно требует использования в ее составе хэш-функций SHA-1 и MD5.

3. Для вычисления кода аутентификации сообщений можно использовать алгоритм ГОСТ 28147-89 в специальном режиме

4. Стандарт ЭЦП ГОСТ Р 34.10-2001 может быть реализован в рамках описания TLS.

5. Для ОРК возможно использование аналогов алгоритма Диффи-Хеллмана в группе точек ЭК, удовлетворяющей ГОСТ Р 34.10-2001.

С целью унификации и обеспечения функциональной совместимости отечественных УЦ, необходимо выработать единый набор идентификаторов отечественных алгоритмов электронной цифровой подписи, используемых при обработке сертификатов в отечественных УЦ.

5 Обеспечение доступности информации

Удостоверяющий центр должен обладать устойчивостью к сетевым атакам, для достижения которой целесообразно использовать следующие методы.

1. Использовать доверенную программную платформу обработки конфиденциальной информации.

2. Использовать межсетевые экраны. При этом в реализации защищенного удостоверяющего центра необходимо использовать как внешний, так и внутренние межсетевые экраны.

3. В целях обеспечения обнаружения и предупреждения компьютерных атак на инфраструктуру удостоверяющих центров необходимо осуществление мониторинга безопасности информационных ресурсов удостоверяющих центров.

4. С задачей мониторинга компьютерных атак тесно связано противодействие вирусному заражению.

6 Типовое решение «Стандарт УЦ»

Одним из перспективных направлений по обеспечению информационной безопасности инфраструктуры удостоверяющих центров (ИУЦ) является разработка типовых компонент удостоверяющих центров, обладающих заданными функциональными свойствами и свойствами по безопасности. Наличие в распределенных информационных системах нарушителя, потенциально обладающего значительными возможностями по деструктивному воз-действию на ИУЦ, требует разработки типовых решений по построению УЦ, имеющих класс защиты достаточный для использования в сетях общего пользования.

Для решения данной проблемы создан и сертифицирован типовой доверенный программный комплекс «Стандарт УЦ». Он обеспечивает выполнение целевых функций УЦ со-гласно Федеральному закону от «Об электронной цифровой подписи» и соответствует классу защиты КВ2 на уровне достаточном для использования в сетях общего пользования.

К разработке средств имитационного моделирования для решения задач обеспечения безопасности информационных технологий

И. С. Батов

1 Введение

В последние годы все большее внимание уделяется вопросам применения методов имитационного моделирования для решения задач обеспечения безопасности сложных, практически значимых объектов информационно-телекоммуникационной инфраструктуры [1], [2], [3]. Данный факт объясняется необходимостью исследования процессов, происходящих в больших, сложных информационно-телекоммуникационных системах, непосредственное использование которых для изучения и тестирования рассматриваемых технологий, не представляется возможным из-за больших затрат и риска деградации или отказа промышленно используемых систем при проведении подобных экспериментов. Кроме того, проводить эксперименты с помощью специализированного программного комплекса для имитационного моделирования гораздо удобнее с технологической точки зрения, чем аналогичные опыты на реальных системах. К таким преимуществам следует отнести возможность оперативного изменения начальных условий, параметров модели и плана проведения эксперимента, получения статистики и ряд других. Основным недостатком имитационного моделирования является сложность оценки адекватности результатов проведенных опытов тем, которые могут быть получены в ходе аналогичных экспериментов на реальной системе. Более подробно решение этих вопросов обсуждается в разделе 3 данной работы. Тем не менее, имитационное моделирование является удобным инструментом для предсказания поведения разрабатываемых систем (в том числе систем информационной безопасности), оценки влияния внедряемой технологии на существующую инфраструктуру, оценки рисков деградации функциональных свойств, которые могут быть обусловлены применением новых служб и сервисов. Таким образом, имитационное моделирование представляет собой инструментарий, дающий исследователю возможность учитывать и исправлять ошибки на ранних стадиях разработки и построения комплекса средств, служащих для той или иной цели. Более подробное обсуждение перечня задач, решение которых эффективно с помощью методов имитационного моделирования представлено в разделе 2 данной работы.

Моделирование отдельных компонент инфраструктуры компьютерных сетей ведется уже много лет, и к настоящему моменту создано большое количество программных комплексов, включающих различные модели протоколов, служб, приложений и устройств, использующихся в современных сетях. Среди таких комплексов можно выделить имитаторы ns2 [4], OmNet++ [5], OPNET Modeler [6], SSFNet [7], а также ряд других. Учитывая необходимость в моделировании большего числа объектов сетевой инфраструктуры, представляется целесообразным использовать существующие инструментальные средства в данной области для создания на их основе имитатора, ориентированного на решение проблем информационной безопасности. Раздел 3 данной работы посвящен требованиям, предъявляемым к базовому комплексу для имитационного моделирования процессов противоборства на сетевой среде.

Очевидно, что для построения многофункциональной, масштабируемой и удобной в работе системы имитационного моделирования, необходимо определить классы задач, которые предполагается с ее помощью решать и общие подходы к их решению. В частности, уже на начальной стадии создания системы необходимо определить уровень детализации, на котором предполагается производить моделирование процессов киберпротиводействия. Раздел 4 посвящен методу выбора имитационной модели компьютерной атаки и уровня ее детализации. Раздел 5 иллюстрирует описанный метод примерами сценариев для моделирования процессов киберпротивоборства с использованием программного ком-

плекса на базе ns2 [4].

2 Назначение и область применения

Необходимо заметить, что наибольший интерес представляет моделирование многоэтапных, распределенных объектов, реализующих сложные составные атаки на систему в целом. Имитационное моделирование простых атак на получение несанкционированного доступа с использованием уязвимости программного обеспечения, например, типа «buffer overflow», практически не имеет смысла. Система либо будет поражена, при наличии таких уязвимостей, либо нет, при их отсутствии. С другой стороны, очевидный интерес представляет детальное описание и предсказание поведения информационной системы, например системы обнаружения вторжений (Intrusion Detection System — IDS), при сложных, многоэтапных атаках. Целесообразно, например, использовать имитационное моделирование при исследовании влияния на компьютерную систему атаки типа «червь», которая является достаточно репрезентативной в классе распределенных атак. С учетом сделанного замечания область применения имитационного моделирования включает следующие задачи.

1. *Задачи исследования воздействия атаки на сеть*[1]. Такие задачи возникают при разработке систем безопасности для определения наилучших (с точки зрения скорости и эффективности определения и устранения атаки, ее последствий) методов противодействия угрозам. Примером такой задачи может быть исследование влияния атаки типа «червь» на инфраструктуру сети и выбор адекватных для перманентного контроля характеристик сети с целью обеспечения своевременной реакции на атаку [8].

2. *Задачи тестирования систем безопасности.* Примером такой задачи является исследование времени реакции системы безопасности в условиях, когда кроме «рабочей» нагрузки на сетевой сегмент появляется дополнительная, создаваемая трафиком атаки, целью которой является отказ в обслуживании сетевой IDS.

3. *Задачи обучения обслуживающего персонала.* Моделирование различных атак на сеть может быть использовано для информирования администраторов о возможном её поведении и выходных сообщениях системы IDS, установленной на данном сетевом сегменте, при реализации тех или иных атак. Результаты моделирования могут быть использованы для получения большого количества выборок для обучения нейросети с целью автоматического определения атаки по сообщениям системы обнаружения вторжений.

4. *Проектирование топологии и расположения сервисов сетей.* Используя симулятор можно исследовать различные варианты инфраструктуры будущей сети, включая топологию, состав и размещение служб и сервисов, организацию политик безопасности с тем, чтобы определить оптимальный из вариантов [10, 9].

5. *Отладка инфраструктуры реальной сети.* Сравнивая результаты имитационного эксперимента на модели сети и аналогичного испытания на соответствующей реальной сети, можно выявлять ошибки в реализации инфраструктуры последней. Упоминание о таком применении можно найти в работе [11].

6. *Исследование как существующих, так и исследовательских протоколов, алгоритмов, технологий и служб.* Результаты имитационного моделирования могут быть использованы для оценки целесообразности, с точки зрения информационной безопасности, применения той или иной технологии в рассматриваемых сетях.

[1] Здесь и далее под понятием сеть, если это не отмечено отдельно, будем иметь в виду сетевую структуру или сегмент, поддерживающий информационно-вычислительный комплекс, подлежащий моделированию с целью решения одной из перечисленных далее задач.

3 Выбор системы имитационного моделирования

Рассмотрим требования, предъявляемые к системе имитационного моделирования для решения задач обеспечения безопасности информационных технологий. Такая система должна удовлетворять следующим требованиям.

- обеспечивать расширение функциональных возможностей имитируемого процесса путем добавления новых моделей;
- обладать способностью выполнять различные сценарии кибератак со степенью детализации, необходимой для получения результатов с точностью, которая планируется постановкой задачи;
- обладать способностью моделировать большие распределенные системы, состоящие из тысяч узлов;
- предоставлять возможности для исследования результатов моделирования на основе аналитических, графических и статистических методов и средств.

В планируемом к разработке программном комплексе должна присутствовать возможность проводить следующие виды верификации и валидации используемых моделей и методов [12].

- *Верификация реализации моделей.* Такое требование подразумевает проверку правильности реализации симулятора в целом, включая отдельные модели протоколов, служб, сервисов и систем. Несмотря на то, что такая проверка относится к проблеме верификации любого программного обеспечения, для комплекса имитационного моделирования она особенно важна, так как даже незначительные ошибки в реализации отдельных модулей симулятора могут привести к большим отклонениям в результатах сложных экспериментов.
- *Валидация моделей протоколов и сервисов.* При проведении моделирования необходимо учитывать точность, с которой модель данного протокола или сервиса, представляет соответствующий объект реальной сети. Валидация проводится сравнением модели протокола с его спецификацией. Этого, однако, бывает недостаточно, так как часто спецификации не определяют протокол полностью, оставляя реализацию деталей на усмотрение разработчиков. Так, например, спецификация TCP допускает большое число различных реализаций, с поведением, сильно отличающимся при изменении деталей алгоритма обработки подтверждений [11]. В этом случае, при необходимости, модель сверяется с конкретной реализацией протокола.
- *Валидация моделей систем.* Симулятор, обычно, использует модели не только программного обеспечения, но также и физических процессов передачи сигнала, модели оборудования. Например, модель процессора для имитации задержек в обработке, модели распространения сигнала (для беспроводных сетей), модель сетевой карты, канала передачи данных.
- *Валидация сценария моделирования.* На этом направлении следует рассматривать, на каких сценариях необходимо проводить моделирование, решать задачу выбора репрезентативных случаев, исследуя, насколько результаты моделирования зависят от изменения параметров сценария (инфраструктуры сети, модели загрузки и других).
- *Валидация методологии моделирования.* Для того, чтобы результаты имитации были статистически корректны, необходимо учитывать свойства используемых генераторов случайных чисел (длину цикла, соответствие распределению), проводить достаточное число имитаций для получения приемлемых оценок характеристик систем и так далее [16, 17].

В качестве основы для построения системы имитационного моделирования для решения задач информационной безопасности была выбрана система имитационного моделирования ns2 [4]. Этот симулятор является широко распространенной системой моделирования компьютерных сетей и включает большое количество модулей реализующих различные модели сетевых протоколов, служб и компонентов компьютерной сети.

Способность моделирования больших распределенных систем в имитаторе ns2 осуществляется за счет возможности параллельных вычислений [18] и выбора подходящего уровня детализации системы [19], [20].

Система имеет встроенные средства протоколирования состояния модели сети для последующей аналитической обработки и статистического исследования [22]. Кроме того, существует пакет для динамической визуализации и построения графиков по результатам моделирования [21].

Открытость исходного кода как ядра симулятора, так и его модулей позволяет проводить необходимые виды валидации и верификации. Кроме того, открытость кода позволяет добавлять реализации собственных модулей и, при необходимости, изменять реализации существующих для решения задач обеспечения безопасности информационных технологий.

Одним из преимуществ ns2 является постоянная валидация моделей различными исследовательскими группами, которые могут сообщать о найденных неточностях в список рассылки — `http://mailman.isi.edu/mailman/listinfo/ns-users`. Имеется набор тестов для проверки корректности работы модулей так, что при внесении изменений имеется возможность проверить, повлияли изменения в исходном коде на работу других модулей ns2.

Таким образом, система ns2 удовлетворяет всем перечисленным выше требованиям и позволяет на ее основе построить систему имитационного моделирования для решения задач обеспечения безопасности информационных технологий.

4 Выбор уровня детализации

Одной из важнейших задач при планировании имитационных экспериментов для исследования систем информационной безопасности является определение состава (структуры) потенциальных атак и уровня их детализации. Модель изучаемого явления (объекта) обязана с должной степенью подробности отображать исследуемые свойства реально протекающих процессов с учетом специфики поставленной задачи и имеющихся средств моделирования. Для наиболее полного исследования возможных воздействий атак на сеть и тестирования систем безопасности необходимо использовать таксономию потенциальных угроз. Метод выделения атак, при этом, может строиться следующим образом.

- На первом этапе определяются объекты рассматриваемой сети, которые могут быть конечной целью многоэтапной атаки с кратким их описанием (идентификацией). Для исследования наиболее важных классов атак, такой список и описание должны содержать информацию о степени критичности объектов. Примерами таких объектов могут быть отдельные комплексы, хранящие значимые базы данных, узлы, обеспечивающие мониторинг критически важных (с точки зрения заранее принятых критериев) объектов, сервера с разделяемой файловой системой и так далее, вплоть до сегментов сети в целом.

- Далее атаки на каждый конкретный объект классифицируются по их целевой установке — атаки на целостность, конфиденциальность и доступность.

- Все атаки с конкретной целью разделяются *по методам проведения*. Под методом понимается основная идея (целевая установка, схема, способ и т.п.) организации атак, которая, обычно, определяется расположением и квалификацией атакующего. Например, для нарушения доступности сегмента сети, атакующий может выбрать один из следующих способов:

  - атаки на ресурсы маршрутизатора;
  - атаки типа «червь»;
  - атаки на используемый протокол маршрутизации;
  - атаки на DNS.

- Атаки каждого из методов разделяются на классы по *этапам проведения*. Этапы выделяются в зависимости от характеристик, которые предполагается оценивать по результатам моделирования. Например, приведенная ниже модель атаки типа «червь» (см. раздел 5) состоит из следующих этапов:

  a) зараженный хост A пытается послать информацию другому зараженному хосту или пытается найти новый объект атаки — хост B, или установить соединение, которое вызовет на хосте B использование уязвимости;

  b) на объекте атаки B используется уязвимость;

c) на объект атаки пересылается тело червя;

d) производится инсталляция, запуск тела червя, установка дополнительных черных ходов, сокрытие появления червя.

При проведении атаки возможны следующие последовательности этих этапов:

– a, b, c, d;
– a, c, b, d;
– a, c, d, b.

Последовательность этапов важна с точки зрения обнаружения атаки системой обнаружения (IDS), так как сообщения о тревогах с отдельных хостов будут отсылаться в последовательности, соответствующей последовательности проведения этапов. Таким образом, в приведенном примере, метод «червь» включает три класса в соответствии с очередностью этапов проведения атаки.

- Продолжая детализацию, следует отметить, что каждый этап может иметь свой *метод проведения*. Так, например, этап поиска нового объекта атаки в приведенном выше примере может реализовываться следующими способами [14].

  – *Пассивное сканирование.* Наблюдая за файлами и действиями узла, программа «червя» делает выводы о том, какие существуют узлы в сети и какие из них уязвимы.
  – *Hitlist scanning.* Атака с использованием списка уязвимых узлов, созданного заранее.
  – *Скрытое сканирование.* Медленное вертикальное сканирование по всем возможным узлам сети
  – *Обычное сканирование.* Быстрое сканирование по всем возможным узлам сети.
  – *Profile matching.* Наблюдая за файлами и действиями узла, программа «червь» делает выводы о стандартном (не вызывающим подозрений) поведении хоста. Обучившись, червь производит скрытое сканирование подходящим образом.

  Аналогично этап использования уязвимости зависит от состава программного обеспечения на атакуемом хосте.

Необходимо заметить, что приведенный выше метод выделения атак использует классификацию по объектам воздействия на каждом из этапов атаки. Объектом этапа может быть любая технология, использующаяся в сети, на любом из уровней ISO/OSI. Примерами таких объектов могут быть протоколы TCP, DNS, OSPF, SMTP, технологии Web, VPN, IDS, программное обеспечение отдельных узлов сетевого сегмента, вплоть до физического воздействия на поддерживающее его оборудование. После выделения множества потенциально уязвимых объектов рассматриваемого сетевого сегмента, для каждого такого объекта выделяются возможные атаки, которые могут быть проведены на поддерживаемую им технологию, с описанием условий успешного проведения атаки и её последствий. Описание атак, условий их успешного завершения и последствий формализуется для последующей автоматической генерации многоэтапных атак, составленных из описанных частных атак на объекты. Процесс автоматической генерации становиться возможным благодаря формализации последствий и условий атак. Пример такой формализации можно найти в [13].

Реакцию модулей системы безопасности, находящихся на отдельных узлах сети, можно моделировать на низком уровне детализации — вероятностями той или иной реакции на соответствующие входные данные. Можно использовать детальную модель узла и модуля безопасности, такую как SIMS [15]. Несмотря на то, что второй способ точнее имитирует процессы отдельного узла сетевого сегмента, низкий уровень детализации может дать более адекватные исследуемой задаче результаты, так как оперирует не с конкретной реализацией этапа атаки, а с вероятностной оценкой ее последствий, охватывая, таким образом, широкий класс возможных реализаций этапа атаки. Этот факт особенно важен при тестировании системы безопасности на предмет выявления атак, использующих неизвестные уязвимости. Кроме того, вероятностный подход позволяет проводить эксперименты, прогнозирующие развитие событий, при пониженных значениях вероятностей определения той или иной атаки.

Значения вероятностей предлагается определять на основе аналитических моделей, используемых в системе безопасности и результатов ее тестирования на реальных системах.

5 Пример атаки для моделирования

Продемонстрируем описанный выше подход на примере модели атаки типа «червь», реализованной с помощью программного комплекса на базе ns2. Модель данной атаки разрабатывалась для тестирования IDS на различных видах возможных «червей» и оптимальной настройки системы обнаружения и защиты от данного вида атак. С этой целью модель включает большое количество настраиваемых параметров для имитации максимально большого числа разновидностей «червя».

5.1 Общие утверждения о возможных тревогах при внедрении червя

Перечислим тревоги, которые могут выдаваться модулями безопасности на хостах сети при распространении «червя» в рамках описываемой модели.

1. Зараженный хост A пытается послать информацию другому зараженному хосту или пытается найти новый объект атаки — хост B, или установить соединение, которое вызовет на хосте B использование уязвимости.

a) Тревога с сетевого IDS узла A «подозрительные исходящие пакеты». Тревога может содержать расширенную информацию — куда идут пакеты, какое отклонение вызывает подозрительность или сразу выдавать название атаки, например «A сканирует порты».

b) Тревога с сетевого IDS узла A «подозрительные входящие пакеты» — может быть вызвано ответами, ранее не фиксировавшимися.

c) Тревога с сетевого IDS узла B «подозрительные входящие пакеты».

d) Тревога с сетевого IDS узла B «подозрительные исходящие пакеты».

2. На объекте атаки B используется уязвимость. Тревога с хостового IDS — обнаружено использование уязвимости.

3. На объект атаки пересылается тело червя.

a) Тревога с сетевого IDS узла A — «подозрительные исходящие пакеты». Атрибуты тревоги — куда идут пакеты, какое отклонение вызывает подозрительность.

b) Тревога с сетевого IDS узла A «подозрительные входящие пакеты» — может быть вызвано ответами, ранее не фиксировавшимися.

c) Тревога с сетевого IDS узла B «подозрительные входящие пакеты».

d) Тревога с сетевого IDS узла B «подозрительные исходящие пакеты».

4. Инсталляция, запуск тела червя, установка дополнительных черных ходов, сокрытие появления червя. Тревога с хостового IDS.

5.2 Сценарий внедрения червя

Червь начинает распространяться одним из следующих трех методов:

- с одного внутреннего узла корпоративной сети;
- с нескольких внутренних узлов корпоративной сети в одно время;
- с нескольких внутренних узлов корпоративной сети в разное время;

1. Выбирается с некоторой вероятностью один из методов поиска объекта атаки — a, ab, ac, ae, b, bc, bd, be, c, ce, d, abc, abd, abe, ace, bce, abce, где a, b, c, d, e — следующие методы поиска.

a) *Пассивное сканирование.* Наблюдая за файлами и действиями узла, программа «червя» делает выводы о том, какие существуют узлы в сети, какие из них уязвимы. Вероятность обнаружения этой атаки системой безопасности задана.

b) *Hitlist scanning.* Атака с использованием списка уязвимых узлов, созданного заранее. Вероятность обнаружения этой атаки системой безопасности задана.

c) *Скрытое сканирование.* Медленное вертикальное сканирование по всем возможным узлам сети — сначала сканирование для поиска существующих узлов, потом поиск уязвимостей на найденном узле. Вероятность обнаружения этой атаки системой безопасности задана.

d) *Обычное сканирование.* Быстрое вертикальное сканирование по всем возможным узлам сети. Вероятность обнаружения этой атаки системой безопасности задана.

e) *Соответствие профилю (profile matching).* Наблюдая за файлами и действиями узла, червь делает выводы о профиле поведения хоста. Начинает атаковать, когда считает, что обучился достаточно. Распределение вероятности того, что червь начнет атаковать через время t, и вероятность обнаружения данной атаки системой безопасности заданы.

2. На узле используется уязвимость. Предполагаем, что червь может использовать несколько типов уязвимостей с разными вероятностями обнаружения. Пересылается исполняемый модуль (или код) червя, который, затем инсталлируется (запускается). Каждому типу уязвимости соответствует своя пересылка с определенной вероятностью обнаружения и свой процесс инсталляции, также с определенной вероятностью обнаружения. Все перечисленные вероятности заданы.

3. После успешной инсталляции червя на зараженном узле начинается процесс поиска нового объекта атаки, описанный выше.

Необходимо заметить, что метод доставки модуля с определенного узла не рассматривается из-за большой вероятности обнаружения и излечения этого узла, что приведет к уничтожению всего червя (метод poison updates [14]).

6 Результаты моделирования

Проиллюстрируем функциональность программного комплекса для имитационного моделирования в применении к одной из описанных выше задач (см. раздел 2), а именно, — к задаче 1. С помощью комплекса исследовалась динамика процесса заражения хостов на сетевом сегменте с использованием одного из видов атаки типа «червь». Данный вид «червя» определяется следующими параметрами.

- В качестве метода поиска объекта атаки используется обычное сканирование (см. подраздел 5.2). В данном случае зараженный хост (субъект атаки) производит поиск уязвимых[2] объектов (или объектов атаки), отсылая всем узлам модели сетевого сегмента пробный пакет на определенный, заранее заданный порт. В случае если на узле установлено уязвимое приложение, субъекту атаки высылается ответ (уведомление), подтверждающий наличие уязвимости на данном порте.

[2]Под уязвимым объектом понимается хост, который может быть заражен рассматриваемым «червем».

Если уязвимости нет, уведомление не посылается. Размеры сканирующих пакетов[3] составляют 1000 байт, задержка между их отсылкой — 0.08 секунды.

- При получении ответа от объекта атаки, зараженный узел отправляет пакеты, содержащие код для использования уязвимости и пакеты с телом «червя» на адрес соответствующего хоста.
- При получении пакетов с телом «червя» на хосте начинается процесс поиска объектов атаки по истечении 0.1 секунды. Суммарный размер пакетов, содержащих код (или исполняемый модуль) для использования уязвимости составляет 5000 байт, пакетов с телом «червя» — 100000 байт.

Данная атака достаточно репрезентативна в классе распределенных кибервоздействий. Кроме того, исследование такой атаки требует проведения экспериментов на больших сегментах сетей, создание и настройка которых связаны с большими затратами. Использование же существующих работающих сегментов сети для экспериментов с приведенной атакой типа «червь» с необходимостью вызовет сбои в работе сети, убытки от простоя в работе и, возможно, значительные затраты на восстановление. С другой стороны, имитационное моделирование предоставляет в данном случае гибкий и эффективный инструмент проведения экспериментов на моделях сетевых сегментов любого масштаба. В качестве модели сетевого сегмента использовались две подсети, имитирующие локальные сети организации, соединенные одна с другой каналом связи с пропускной способностью 100 Мбит. На рисунке 1 изображена топология данной модели. В правом верхнем углу представлена первая локальная подсеть (137 узлов), в нижнем левом — вторая подсеть (123 узла). Топология каждой из подсетей получена с помощью генератора топологий tiers2.1 [23]. Пропускная способность всех каналов локальной сети 100 Мбит. Задержки в пределах локальных подсетей не превышают 0.001 мс. Задержка при передаче по каналу, соединяющему локальные подсети, составляет 5 мс. Для обработки очередей на портах коммутаторов локальных сетей используется алгоритм DropTail. Реализации приведенных моделей сетевого сегмента и атаки на программном комплексе для имитационного моделирования позволяют быстро и эффективно получать практически любые данные, которые можно было бы извлечь в результате аналогичного опыта на реальной системе. Целью данной публикации, однако, не является подробное и всесторонне описание экспериментов по исследованию влияния атаки типа «червь» на модель сети и их результатов, поэтому мы ограничимся представлением только одной характеристики воздействия данной атаки, а именно, представим характеристику, отображающую зависимость роста количества зараженных узлов от времени. Данная характеристика позволяет оценивать опасность исследуемого вида «червя», а также оценивать преимущества атакующего при использовании сразу нескольких одновременных точек старта. Для оценки последнего параметра, данные о скорости заражения снимались для двух конфигураций сценариев моделирования. В первой конфигурации «червь» начинает распространяться с двух узлов — по узлу для каждой локальной сети. Во втором случае «червь» имеет единственную точку старта. Необходимо отметить, что смена сценария осуществляется заменой нескольких строк в файле, описывающем конфигурацию эксперимента, и происходит за считанные минуты. На рисунках 2 и 3 представлены графики роста количества зараженных узлов от времени для первой конфигурации (рисунок 2) и, соответственно, для второй (рисунок 3).

Используя приведенные данные можно сказать, что время распространения незначительно отличается в двух рассмотренных конфигурациях точек старта червя.

Описанный эксперимент позволяет сделать следующие выводы:

- на разрабатываемом комплексе для имитационного моделирования возможна реализация сложных, распределенных атак;
- результаты моделирования позволяют эффективно наблюдать за интересующими исследователя характеристиками;
- с помощью разрабатываемом комплекса возможно моделирование экспериментов, проведение которых на аналогичных реальных сегментах сети представляет большие сложности, а иногда и не возможно.

[3]Здесь и далее под пакетом понимается сообщения протокола сетевого уровня (IP). Максимальный размер пакета составляет 1024 байт.

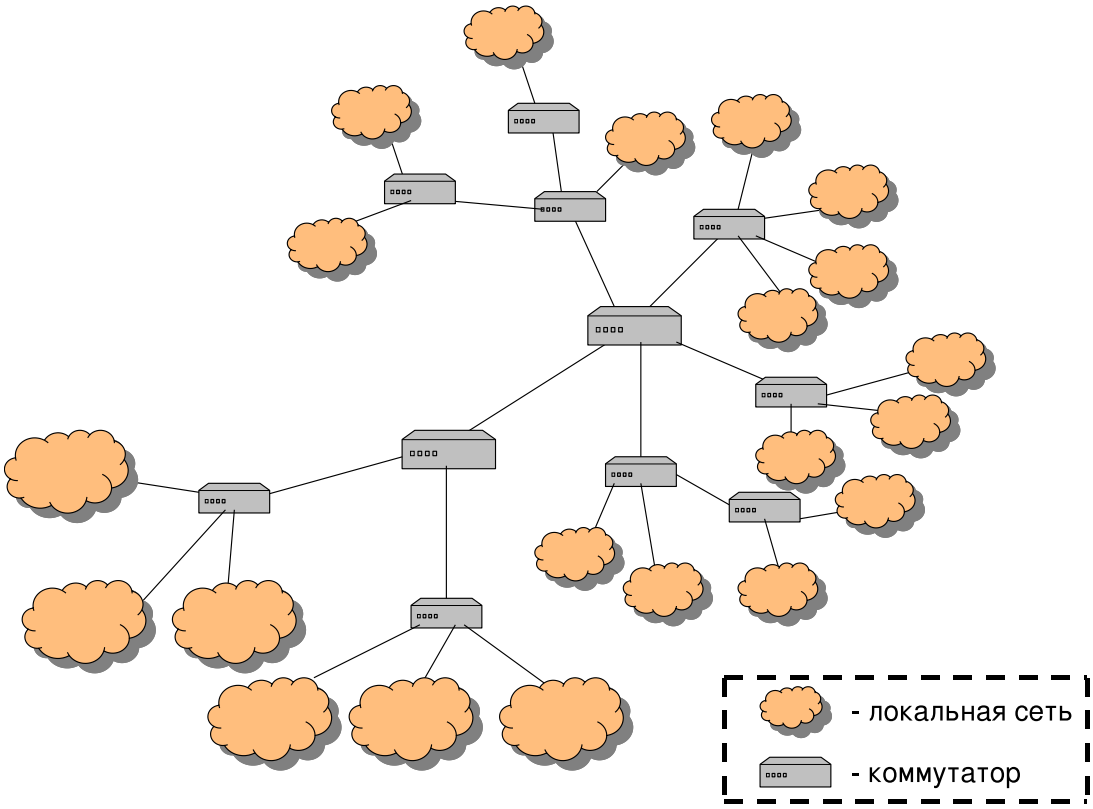


Рис. 1: Иллюстрация модели сегмента сети для исследования атаки типа «червь»

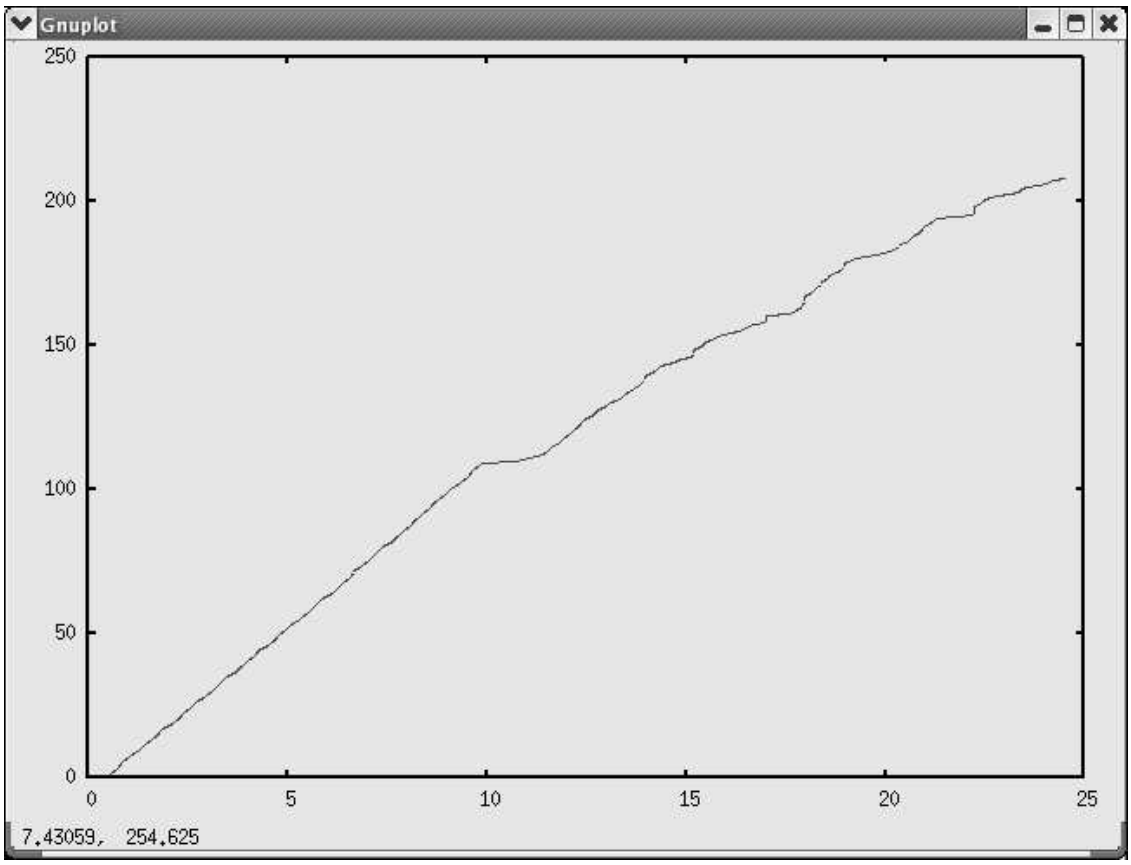


Рис. 2: График скорости распространения червя (количество зараженных узлов/сек) для двух точек старта

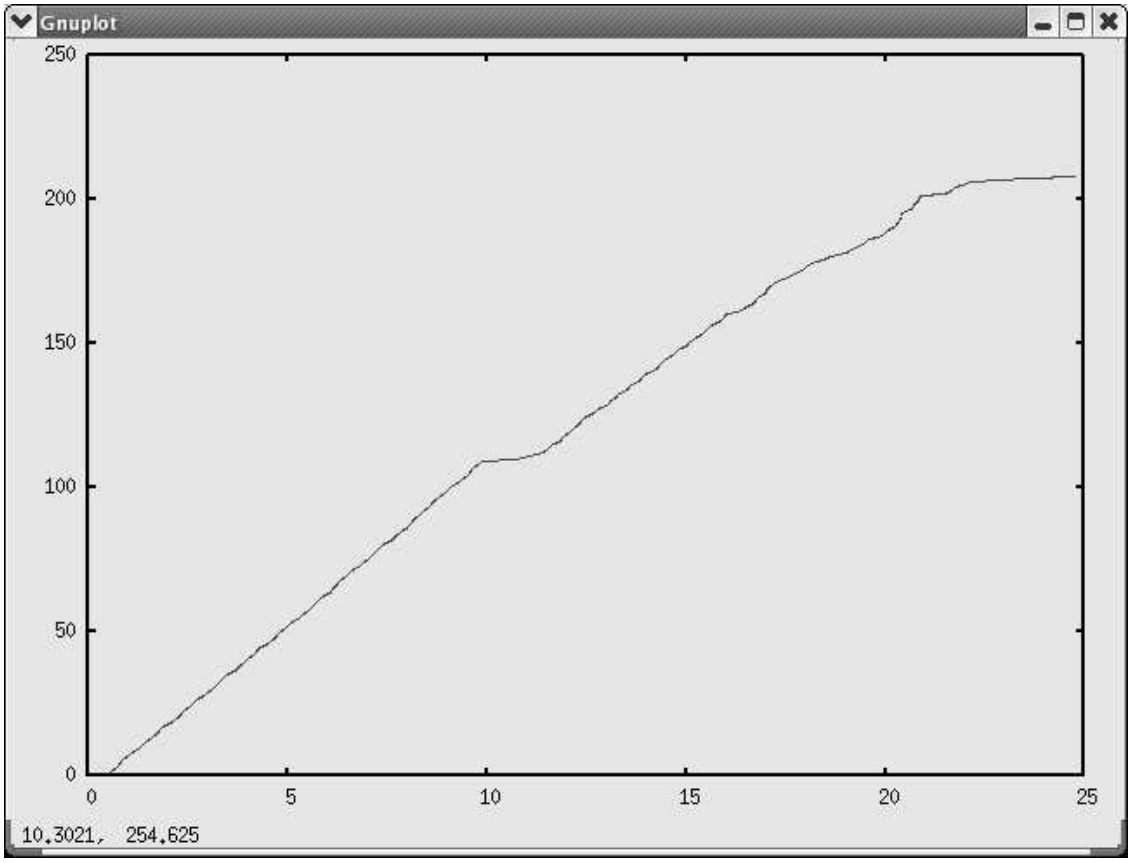


Рис. 3: График скорости распространения червя (количество зараженных узлов/сек) для одной точки старта

7 Заключение

В данной работе описан подход к разработке средств имитационного моделирования для решения задач обеспечения безопасности компьютерных сетей. На основе анализа существующих на этом направлении результатов представлен перечень задач, которые предполагается решать с помощью соответствующего программного комплекса.

Предложен способ выбора атак и уровня их детализации для наиболее полного описания возможных сценариев нападения. В качестве примера предложенной детализации приведена модель атаки типа «червь». Данная модель разрабатывалась с целью тестирования алгоритмов обнаружения на как можно большем числе возможных вариантов проведения атаки, с целью выявления наиболее опасных сценариев.

По итогам работы на первом этапе выделены постановки задач и методы их решения с целью создания на этой основе общего каркаса системы имитационного моделирования для решения задач сетевой безопасности. В качестве первого, представительного в классе рассматриваемых объектов примера, реализована описанная выше атака типа «червь».

Литература

[1] Васенин В. А. Проблемы математического, алгоритмического и программного обеспечения компьютерной безопасности в Интернет. Математика и безопасность информационных технологий. Материалы конференции в МГУ 23–24 октября 2003 г. — М.: МЦНМО, 2004, с. 111–143.

[2] Nicol D. Modeling and Simulation in Security Evaluation. *IEEE Security and Privacy*, vol. 03, no. 5, p. 71–74, September/October, 2005.

[3] Second Workshop on Ultra Large Networks: New Research Directions in Modeling and Simulation-based Security, 2003.

[4] SAMAN, CONSER, ACIRI, Network Simulator 2. `http://www.isi.edu/nsnam/ns`.

[5] Varga A. OmNet++ (`http://www.omnetpp.org`).

[6] OPNET Modeler (`http://www.opnet.com/products/modeler/home.html`).

[7] Renesys Corporation, SSFNet (`http://www.ssfnet.org`).

[8] Aranya A. In Search of a More Insidious Worm. Computer Science Department, Stony Brook University / Aranya A., Callanan S.

[9] Carrier B. Impact of network design on worm propagation / B. Carrier, S. Jeyaraman, S. Sellke; Center for Education and Research in Information Assurance and Security, Purdue University, West Lafayette, IN 47907-2086.

[10] Surdu J. R. Military Academy Attack/Defense Network Simulation / J. R. Surdu, J. M. D. Hill, R. Dodge, S. Lathrop, and C. A. Carver, Jr. Department of electrical engineering and computer science. United States Military Academy.

[11] Heidemann, J. Expanding Confidence in Network Simulation / J. Heidemann, K. Mills, S. Kumar USC/Information Sciences Institute Research Report 00-522, April 2000, submitted for publication to IEEE Computer.

[12] Bagrodia R. Position Paper on Validation of Network simulation models. Bagrodia Rajive, Takai Mineo Computer Science Department UCLA. (`pcl.cs.ucla.edu/papers/`).

[13] Ning P. Building Attack Scenarios through Integration of Complementary Alert Correlation Methods, Cyber Defense Laboratory Department of Computer Science North Carolina State University.

[14] Nazario J. Defense and Detection Strategies against Internet Worms. 2004 ARTECH HOUSE, INC. 685 Canton Street Norwood, MA 02062.

[15] Garg A. SIMS: A Modeling and Simulation Platform for Intrusion Monitoring/Detection Systems / Garg A., Upadhyaya S., Chinchani R., Kwiat K.; 2003 Summer Computer Simulation Conference, SCSC 2003, July 20–24, 2003, Montreal, Canada

[16] Pawlikowski K. Do Not Trust All Simulation Studies Of Telecommunication Networks. Department of Computer Science, University of Canterbury Christchurch, New Zealand.

[17] Jeruchim M. Simulation of Communication Systems : Modeling, Methodology and Techniques (Information Technology: Transmission, Processing and Storage), second edition / Michel C. Jeruchim, Philip Balaban, K. Sam Shanmugan; Plenum US; 2 edition, 2000.

[18] Georgia Institute of Technology PDNS — Parallel/Distributed NS. `http://www.cc.gatech.edu/computing/compass/pdns/`.

[19] Huang P. Enabling Large-scale Simulations: Selective Abstraction Approach to the Study of Multicast Protocols / Polly Huang, Deborah Estrin, John Heidemann; USC/Information Science Institute University of Southern California.

[20] Dutta D. Faster Network Design with Scenario Pre-filtering / Debojyoti Dutta, Ashish Goel, and John Heidemann; in proceedings of the International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, pp. 237-246. Fort Worth, Texas, USA, USC/Information Sciences Institute, IEEE. October, 2002. `http://www.isi.edu/~johnh/PAPERS/Dutta02d.html`.

[21] Estrim D. Network Visualization with the Nam, VINT Network Animator / Deborah Estrin, Mark Handley, John Heidemann, Steven McCanne, Ya Xu, Haobo Yu; *IEEE Computer*, 33 (11), p. 63–68, November, 2000. `http://www.isi.edu/~johnh/PAPERS/Estrin00b.html`.

[22] The *ns* Manual. `http://www.isi.edu/nsnam/ns/index.html`.

[23] Calvert K. Modeling Internet Topology / K. Calvert, M. B. Doar, E. W. Zegura; IEEE Communications Magazine, June 1997. `http://www.geocities.com/ResearchTriangle/3867/sourcecode.html`.

К вопросу о создании комплекса имитационного моделирования составных компьютерных атак

М. В. Большаков

1 Постановка задачи

Имитационное моделирование информационно-вычислительных комплексов и сетевых структур, как правило, проводится с целью получения оценок тех или иных характеристик, которые невозможно, или практически сложно получить с помощью аналитических моделей. Полная проверка (валидация) таких характеристик исследуемой системы часто является затруднительной, в первую очередь — в виду большой размерности пространства состояний. Использование же натурного моделирования для этих целей требует, как правило огромных временных и вычислительных затрат и не является достаточно эффективным подходом.

Поведение компьютерной системы в значительной степени определяется ее программным обеспечением. Имитационное моделирование удобно лишь для некоторых классов атак. Моделировать атаки на получение несанкционированного доступа с использованием уязвимости программного обеспечения, например, типа buffer overflow, практически не имеет смысла. Система будет либо поражена, при наличии таких уязвимостей, либо нет, при их отсутствии. Подобные уязвимости, очевидно приводящие к захвату системы, неплохо выявляются и исправляются известными методами (например, с помощью специализированных утилит MBSA и nessus, см. [1] и [2]). Анализ атаки на более детальном уровне представляет интерес в случае, когда во время ее проведения на систему оказывается несколько последовательных воздействий, в результате которых нарушитель получает все больший контроль. Такой контроль, например, он может получить в результате использования нескольких «не опасных» (при использовании каждой в отдельности) уязвимости и ошибок в конфигурировании системы. Такие атаки будем называть составными.

Оценка уязвимости объекта может быть проведена на основе перебора всех возможных сценариев развития атаки и оценки адекватности реакции объекта на внешние воздействия. Сценарии составных атак могут быть созданы на основе абстрактной модели нарушителя, позволяющей описать возможные методы воздействия на объект и методы выбора направления развития атаки. Оценку реакции моделируемой системы на развитие атаки можно провести на основе детальной модели системы. Описание сценария атаки в рамках модели нарушителя будем называть *моделью атаки на высоком уровне абстракции*. Описание сценария атаки в рамках более детальной модели системы будем именовать *моделью атаки на низком уровне абстракции*.

Таким образом, имитационное моделирование атаки на информационно-вычислительную распределенную систему целесообразно проводить по схеме, представленной на рис. 1.

Составные атаки будем строить с помощью систем логического вывода на основе высокоуровневой модели системы и возможного поведения нарушителя в рамках этой модели. На основе такого представления проводится логический вывод возможных сценариев составных атак в виде последовательностей атомарных атак, каждая из которых приводит к увеличению количества знаний нарушителя о системе. Полученные последовательности передаются на вход генератора низкоуровневых атак, результатом работы которого является входная последовательность для комплекса детального моделирования поведения системы и оценки адекватности ее реакции.

Предложение. *Такие эксперименты будут независимы и могут быть использованы для проведения статистического анализа случайных величин, определенных далее.*

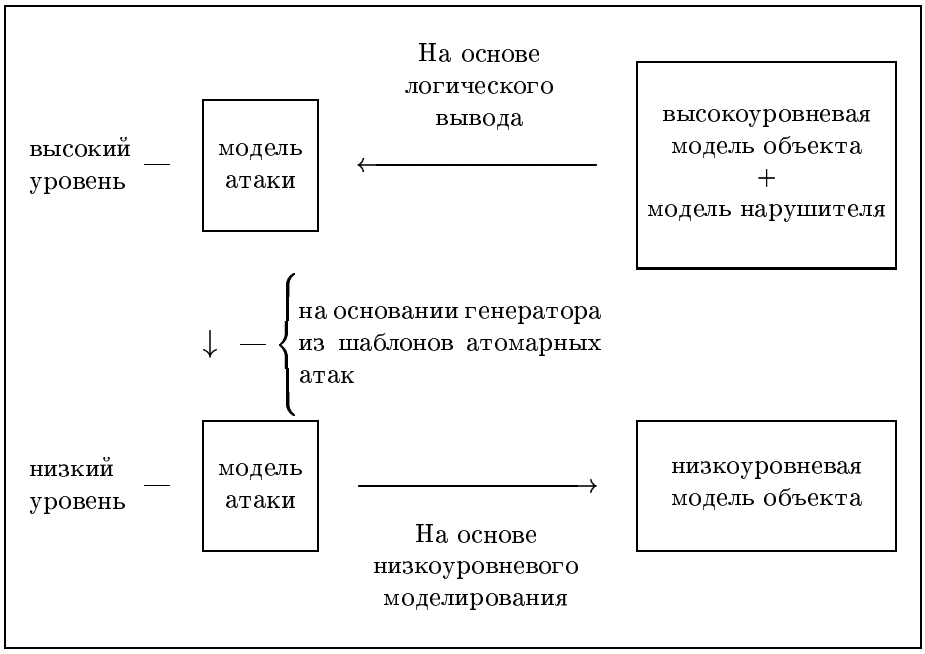


Рис. 1: Схема проведения имитационного моделирования

2 Модель атаки высокого уровня абстракции

Далее будем считать, что модель высокого уровня адекватно описывает поведение среднего нарушителя, использующего в основном не им самим разработанные инструменты. Защита от более «продвинутых» нарушителей, умеющих выявлять уязвимости в программном обеспечении, для обеспечения заданного уровня гарантии неуязвимости системы должна происходить на этапе разработки.

2.1 Модель нарушителя

Модель нарушителя в рамках системы логического вывода представлена в виде следующих компонент:

- библиотека уязвимостей в виде преобразований состояний высокоуровневой модели системы;
- библиотека атомарных атак на уязвимости в виде преобразований данных, известных нарушителю;
- набор возможных для использования конкретным нарушителем уязвимостей;
- начальные данные об объекте, известные нарушителю.

2.2 Модель системы

Моделируемая система в рамках программного обеспечения логического вывода представлена в виде следующих компонент:

- топология системы;
- описание процессов (в частности доступов к ним) в подсистемах в терминах библиотеки уязвимостей (см. модель нарушителя).

Пусть $\mathcal{O}$ — множество объектов, описывающих систему моделирования, со следующими свойствами. Во-первых, объекты представляют собой достаточно крупные подсистемы с тем, чтобы не порождать большое пространство перебора. Во-вторых описание моделируемой системы в виде совокупности подобъектов достаточно детально для описания уязвимостей, которые могут быть использованы для захвата системы. Символом $\mathcal{T}_{\mathcal{O}}$ будем обозначать конечные последовательности объектов.

Пусть $\mathcal{P} = \{P_i^{(n)}\}$, где $P_i^{(n)}$ — n-мерный предикат, описывающий некоторое свойство моделируемой системы, а именно, — функциональные взаимосвязи между подобъектами его составляющими.

2.3 Метод описания атомарных атак

Атомарные атаки возможны либо в случае наличия уязвимостей в подобъектах, либо в результате неправильной конфигурации и настройки их взаимодействия друг с другом.

Абстрактно, атака на уязвимость описывается в виде правил вывода: $P_{b_1}, \ldots, P_{b_l} \vdash P_{d_1}, \ldots, P_{d_k}$ для некоторых целых l, k и предикатов $P_{b_i}, P_{d_j} \in \mathcal{P}$. Левая часть правила описывает условия использования уязвимости. Правая часть описывает данные, которые стали доступны нарушителю в результате использования уязвимости. Существуют базы данных с описаниями известных уязвимостей (см., например [4]) в виде предусловий и возможных результатов, которые можно использовать для создания вышеописанных правил.

Наличие уязвимости в системе задается в виде специального предиката истинного для каждого из объектов системы, в которых содержится уязвимость и ложного для всех остальных. Такие предикаты также входят в высокоуровневую модель системы. Их порождение для реальных систем возможно на основе утилит сканирования системы на наличие уязвимости (например, см. [1, 2]) и описаний уязвимостей (например, см. [3]). Таким образом эта часть модели может быть получена автоматически.

Неправильная конфигурация системы выявляется на основе анализа информации о возможных доступах к объектам. Поиск таких уязвимостей может быть описан в виде поиска последовательности правил вывода, позволяющего получить доступ к объекту.

Описание системы в терминах конфигурации может быть очень сложным, оценка адекватности ограничения количества данных, заложенных в модели, может быть сделана только опытными администраторами. С учетом этого обстоятельства, создание автоматических утилит для генерации всех доступов в заданной конфигурации является задачей второй очереди построения комплекса имитационного моделирования.

Определение. Моделью системы верхнего уровня будем называть набор предикатов, описывающих доступы и свойства объектов (в частности наличие уязвимостей) объектов системы, то есть $M_{HI} = \left\langle \{O_i\}, \left\{P_i^{k_i}\right\} \right\rangle$.

Пространство всех систем будем обозначать $\mathcal{M}_{\mathcal{HI}}$.

2.4 Метод генерации составных атак

Пусть $O_k \in \mathcal{O}$ объект, описывающий нарушителя. В своих действиях нарушитель преследует определенную цель, например, одну из следующих.

- Завладеть определенными привилегиями на определенном объекте системы $O_i \in \mathcal{O}$, то есть достичь предиката $root(O_i, O_k)$, где $root \in \mathcal{P}$.
- Получить доступ на чтение в какой-либо объект системы, то есть достичь предиката $read(O_i, O_k), read \in \mathcal{P}$.

Множество всех целей будем обозначать $\mathcal{T}$.

В модели нарушителя будем учитывать начальные знания, которыми он обладает, для моделирования более осведомленных нарушителей, имеющих частичные знания о системе, а также внутренних нарушителей. Таким образом, модель нарушителя описывается в виде $N = \left\langle \left\{P_i^{k_i}\right\}, \{O_i\} \right\rangle$, где первое множество это предикаты начальных знаний, а второе — возможные к использованию уязвимости. Множество всех нарушителей будем называть $\mathcal{N}$.

Генерация составных атак в терминах высокоуровневой модели производится на основе поиска путей логического вывода для достижения поставленной цели на основе правил из библиотеки уязвимостей. Такой вывод будет представлять из себя последовательное применение правил, описывающих в системе вывода атомарные атаки, к данным, которые известны нарушителю. Последовательность вывода в дальнейшем используется для порождения модели составных атак в низкоуровневом представлении объекта. Процесс преобразования модели атаки можно описать в виде функции $generate$: $\mathcal{M}_{\mathcal{HI}} \times \mathcal{N} \times \mathcal{T} \times \mathbb{N} \to \mathcal{T}_{\mathcal{O}}$, где первый аргумент — описание системы, второй — описание

нарушителя, третий — цель атаки, четвертый — номер пути атаки при заданной нумерации и, соответственно результат — последовательность атак. В целом, атаки предполагается порождать аналогично тому, как это описано в [5], с учетом ограниченности дерева переборов, например, ограничивая время проведения атаки путем задания времени на проведение атомарной атаки.

3 Модель низкого уровня

Модель атаки низкого уровня представлена в терминах детальной модели системы и описывает возможные реакции моделируемой системы на атаку. В терминах модели низкого уровня возможно проводить оценку работы систем обнаружения атак, оценивать последствия атак, исследовать поведение какого-либо объекта во время атаки. Далее опишем способ формального описания модели системы, метод представления атак в терминах этой модели и предложим функции для исследования поведения системы во время атаки.

3.1 Определения

Определим следующие множества:

- $\mathcal{S}$ — множество состояний объекта;
- $\mathcal{T}_{\mathcal{S}}$ — множество конечных последовательностей состояний объекта;
- $\mathcal{I}$ — множество входных состояний объекта, например сетевых пакетов (не ограничивая общности можно считать $\mathcal{I} \subseteq \mathcal{S}$);
- $\mathcal{T}_{\mathcal{I}}$ — множество конечных последовательностей входных состояний.

Модель объекта, в общем случае представляемая в виде алгоритма, описывающего изменения состояния системы в соответствии с входными данными и начальным состоянием, то есть: $M_{LOW} \colon \mathcal{T}_{\mathcal{I}} \times \mathcal{S} \to \mathcal{T}_{\mathcal{S}}$. Количество статистических испытаний, необходимых для получения оценки заданной точности может быть получено, например, из оценок для схемы испытаний Бернулли. Для вышеприведенного абстрактного случая пространство следов поведения объекты велико и рост количества следов оценивается величиной $2^{l \cdot |\mathcal{S}|}$, где l — время функционирования объекта. Однако в реальной жизни, функции, описывающие функционирование объекта обладают рядом свойств — префиксностью особого вида, или различными видами «sequential consistency». По этой причине оценка количества необходимых экспериментов может быть существенно улучшена. Пространство всех моделей низкого уровня будем обозначать $\mathcal{M}_{\mathcal{LOW}}$. Разработаны средства для достоверного описания компьютерных систем с помощью таких функций M_{LOW} — различного рода симуляторы и эмуляторы, в качестве основы для разработки моделирующего комплекса используется система NS2 (см. [6]).

3.2 Фиксируемые характеристики

На основе модели атаки, транслированной из модели высокого уровня, проводится моделирование (далее называемое *экспериментом*) поведения сетевого оборудования, узлов и, возможно программного обеспечения на них в соответствии с последовательностью входных состояний. В результате проведения эксперимента получаем последовательность состояний оборудования, описывающих функционирование объекта во время проведения атаки. На основе этих последовательностей можно вычислить усредненные характеристики результатов экспериментов. Предположительно, такие характеристики могут быть представлены в виде одной из следующих функций.

- Произошло ли поражение объекта, то есть задан предикат

$$P \colon \mathcal{T}_{\mathcal{S}} \to \{true, false\}, \tag{1}$$

  оценивающий поведение оборудования в процессе эксперимента.

- Количественный ущерб объекту, представленный в виде функции

$$U : \mathcal{T}_{\mathcal{S}} \to \mathbb{R}. \tag{2}$$

  Например, U может отображать количество зараженных компьютеров, или уровень пропускной способности backbone сети.

Для оценки значений характеристик этих типов мы получаем следующие статистические задачи.

- Оценка вероятности следующего события (для случая (1)):

$$\{P(M_{LOW}(pt, s_0)) = false\}, \quad pt \in PT \subseteq \mathcal{T}_{\mathcal{I}}, \quad s_0 \in S_0 \subseteq \mathcal{S}. \tag{3}$$

- Оценка распределения случайной величины (для случая (2)):

$$U(M_{LOW}(pt, s_0)), \quad pt \in PT \subseteq \mathcal{T}_{\mathcal{I}}, \quad s_0 \in S_0 \subseteq \mathcal{S}. \tag{4}$$

- Оценка корреляции поведения для двух заданных функций из случая (2) — U_1, U_2.

Далее приведем несколько примеров исследуемых величин.

Примеры использования подхода (1)

- Предикат P задается следующим образом — $P(st) = \forall s \in st,\ P_s(s)$, где $P_s : \mathcal{S} \to \{true, false\}$ — предикат, оценивающий состояния оборудования как безопасное и небезопасное в каждый момент времени. Предикат P_s может описывать следующие события:
  - для моделирования результата атаки

$$P_s(state) = \begin{cases} true, & \text{если поражен критический узел сети;} \\ false & \text{в противном случае;} \end{cases}$$

  - для моделирования срабатывания простейших IDS (Intrusion Detection System)

$$P_s(state) = \begin{cases} true, & \text{если IDS сработало;} \\ false & \text{в противном случае.} \end{cases}$$

- Предикат P имеет более сложную структуру и зависит от пути, которым объект пришел к конечному состоянию. Такие предикаты уже сильно зависят от модели системы и описывают события, которые произошли в строго последовательном порядке.
  - В случае оценки работы более сложных IDS, если за $s_{ids} \in \mathcal{S}$ обозначим событие отражающее срабатывание IDS, тогда

$$P(st) = \begin{cases} true, & \text{если } s_{ids} \text{ появилось в } st \text{ после событий,} \\ & \text{отражающих проведение атаки или ее части;} \\ false, & \text{иначе.} \end{cases}$$

Примеры использования подхода (2)

- Функция U отражает количество захваченных узлов в результате атаки.
- Функция U отражает минимум свободной полосы пропускания (максимум потерь пакетов) на выделенном участке сети.
- Функция U отражает урон в денежном эквиваленте по раскрытию добытой в рамках модели M_{LOW} информации нарушителем.

4 Методы преобразования сценариев атак во входные последовательности модели нижнего уровня

Для каждой атомарной атаки строится шаблон в виде программы, позволяющий генерировать последовательность входных событий (например, для сетевого уровня — пакетов), соответствующий этапам проведения атомарной атаки. Таким образом, на основе знания характеристики системы и использования дополнительных параметров шаблон позволяет генерировать «атаку» на систему.

В дальнейшем, полученный трафик, моделирующий атаку перемешивается с неопасным фоновым трафиком и подается на вход моделирующего комплекса.

Представим шаблон в виде функции $T_O \colon \mathcal{M}_{\mathcal{HI}} \times \mathbb{N} \to \mathcal{T}_{\mathcal{I}}$, где $O \in \mathcal{O}$, первый аргумент — система, второй аргумент — номер возможной модификации этой атаки на нижнем уровне, результатом является последовательность входных событий для модели. Для составной атаки получаем следующую входную последовательность:

$$PT_A(M_{HI}, T_O) = concat(T_{O_1}(M_{HI}, i_1), \ldots, T_{O_n}(M_{HI}, i_n))$$

где M_{HI} — модель системы, а $T_O = concat(O_1, \ldots, O_n)$.

Замечание. Очевидно, что шаблоны описывают лишь подмножество возможных реальных атак для класса абстрактных атомарных атак, поэтому возможность их статистического использования должна быть обоснована.

Также задана некая функция перемешивания трафиков: $mix \colon \mathcal{T}_{\mathcal{I}} \times \mathcal{T}_{\mathcal{I}} \times \mathbb{N} \to \mathcal{T}_{\mathcal{I}}$, где первый и второй аргументы отвечают за перемешивание трафика, а третий описывает номер возможной подстановки.

Итак, если заданы

- $M_{HI} \in \mathcal{M}_{\mathcal{HI}}$ — высокоуровневая модель системы,
- $N \in \mathcal{N}$ — модель нарушителя,
- $T \in \mathcal{T}$ — цель нарушителя,
- $M_{LOW} \in \mathcal{M}_{\mathcal{LOW}}$ — высокоуровневая модель системы,
- $s_0 \in \mathcal{S}$ — начальное состояние системы,
- предикат P для случая (1),
- функция U для случая (2),

то получаем следующие случайные величины, зависящие от a, b, c и T_{back} ($a, b, c \in \mathbb{N}$ — целые числа, отвечающие за выбор варианта проведения атаки и $T_{back} \in \mathcal{T}_{\mathcal{I}}$ — фоновый трафик):

- $P\left(M_{LOW}\left(mix\left(PT_A\left(generate(M_{HI}, N, T, a), b\right), T_{back}, c\right), s_0\right)\right)$ для случая (1);
- $U\left(M_{LOW}\left(mix\left(PT_A\left(generate(M_{HI}, N, T, a), b\right), T_{back}, c\right), s_0\right)\right)$ для случая (2).

5 Заключение

В настоящей публикации представлен подход к построению комплекса имитационного моделирования, а также методов анализа оценки защищенности компьютерных систем применительно к сложным, составным атакам. Проект комплекса разрабатывался для достижения следующих целей.

- Попытаться получить *обоснованные* статистические результаты исследований уязвимости сетевых структур к составным атакам.
- Создать среду тестирования работы IDS на предмет обнаружения составных атак.
- Использовать создаваемый комплекс, как компоненту активного мониторинга состояния системы для своевременного предупреждения о возможности проведения составных атак.
- Обеспечить упрощенное, полуавтоматическое порождение модели реальной системы.

На настоящее время идет реализация такого комплекса и уже разработана часть его компонентов. Перспективы развития комплекса связаны с исследованиями на следующих направлениях.

- Модель сложна и имеет огромное пространство состояний, поэтому единственный способ оценки ее параметров — способ независимых статистических испытаний. Для этого необходимо разработать адекватный математический аппарат поддержки.
- Разработка полуавтоматического построения модели (в частности, моделей верхнего и нижнего уровня) на основе информации от различных сканеров безопасности и библиотек уязвимостей. Полностью автоматическое описание системы в рамках такого подхода невозможно в силу того, что автоматический выбор значимых характеристик для описания модели верхнего уровня практически нереализуем, а полное описание системы значительно увеличивает пространство вывода и уменьшает точность оценок параметров.

Литература

[1] *Microsoft Baseline Security Analyzer (MBSA).* `http://www.microsoft.com/technet/security/tools/mbsahome.mspx`.

[2] *NESSUS Open Source Vulnerability Scanner Project.* `http://www.nessus.org`.

[3] Matthew W., Tiffany B., Todd W., Robert R. *Introduction to OVAL: A new language to determine the presence of software vulnerabilities.* `http://oval.mitre.org/documents/docs-03/intro/intro.html`, November 2003. Web page fetched on October 28, 2004.

[4] *National Vulnerability Database*, `http://nvd.nist.gov/`, Comprehensive CVE vulnerability database that integrates all U.S. Government publicly available vulnerability resources.

[5] Xinming O., Sudhakar G., Andrew W. A. *MulVAL: A logic-based network security analyzer.* 14th USENIX Security Symposium, Baltimore, Maryland, August 2005.

[6] *The Network Simulator — ns-2.* `http://www.isi.edu/nsnam/ns/`.

Модель динамического мониторинга безопасности состояний системы

С. С. Корт

Основным подходом, используемым для описания модели системы при исследовании ее безопасности в процессе проектирования, является анализ состояний системы. В данной статье предлагается описание подхода к анализу безопасности состояний с целью выявления нарушений безопасности вычислительной системы в процессе ее функционирования. Подсистема динамического мониторинга безопасности состояний системы должна обнаруживать переходы системы в небезопасные состояния, вызванные выполнением некорректных операций пользователя в системе или вторжениями в систему.

Таким образом, задача динамического мониторинга безопасности состояний системы состоит в:

1. Обнаружении состояний, противоречащих определенной в системе политике безопасности, — небезопасных состояний.
2. Выявлении причин, приведших систему в небезопасное состояние.
3. Оценке безопасности системы, в которую осуществляется вторжение.

Предлагаемый подход должен учитывать нарушения безопасности системы, вызванные как невыполнением политики безопасности, так и атаками на систему, что ведет к необходимости описания двух моделей. При этом обе модели должны быть описаны на базе единого математического формализма, позволяющего в дальнейшем провести объединение моделей. В качестве такого формализма в работе выбран автомат конечных состояний, описывающий поведение субъекта системы.

1 Модель системы, учитывающая нарушения политики безопасности

Введем определения понятий, используемых в дальнейших рассуждениях: $\{S\}$ — множество субъектов; $\{O\}$ — множество объектов; $\{Op\}$ — множество операций; $\{Prg\}$ — множество сервисов, используемых субъектом (программ или программных интерфейсов). Введение данного множества в описание модели системы обосновано тем фактом, что операции субъекта системы над объектами выполняются с использованием сервисов. Тогда матрица доступа субъектов системы к объектам может быть определена следующим образом: $M'(s, prg, o)$ — матрица доступа программ, используемых от имени субъектов для выполнения операций с объектами системы.

Тогда автомат $A = \{\sigma, t, Out, \sigma_0, \delta, \lambda\}$, представляющий безопасность функционирования пользователя по отношению к политике безопасности, определенной в системе, может быть описан следующим образом:

$\sigma = \{op_1(prog_1, O_1), \ldots, op_i(prog_i, O_i)\}$ — состояния автомата, описывающие операции, выполненные субъектом системы над объектами; множество состояний является частично-упорядоченным.

$t \in Op(prg_i, o_j)$ — управляющие символы автомата, соответствующие операциям, выполненным субъектом над объектами системы с использованием программ.

Под безопасным состоянием понимается состояние, описывающее операции, выполненные субъектом и не противоречащие политике безопасности. Таким образом, оценка безопасности состояния описывает выход автомата как $Out = \{Sec, UnSec\}$. Автомат заканчивает свою работу, если он переходит в небезопасное состояние.

Тогда функция перехода δ может быть описана следующим образом:

$$\forall t = op_i \; \exists op_i(prog_i, O_i) \in \sigma_i = t \to \sigma_{i+1} = \sigma_i;$$
$$\forall t = op_i \; !\exists op_i(prog_i, O_i) \in \sigma_i = t \to \sigma_{i+1} = \sigma_i \cup t.$$

Функция выхода автомата λ может быть записана следующим образом:

$$\forall t = op_i(op_i(prog_i, O_i) \in \sigma_i) \vee (op_i(prog_i, O_i) \in M'(s, prg, o)) \to Out = Sec;$$
$$\forall t = op_i(op_i(prog_i, O_i) \in \sigma_i) \vee (op_i(prog_i, O_i) \notin M'(s, prg, o))$$
$$\to Out = UnSec.$$

2 Модель обнаружения вторжений

После того, как описана модель системы, соответствующая проверке безопасности функционирования системы в соответствии с политикой безопасности, рассмотрим модель, описывающую возможные атаки на систему.

Атаки на систему идентифицируются с использованием сигнатур атак $\{Sgt_m\}_{m=1}^{M}$. Множество сигнатур, описывающих атаки, может быть разбито на подмножества в соответствии со свойством PROPERTY. Свойство PROPERTY отображает множество сигнатур атак во множество prp_n, $n \in 1 : N$, описывающие цели атаки. Каждый элемент множества prp_n отражает цели атаки, связанной с использованием сигнатуры. Множество $\{prp_n\}_{n=1}^{N}$ является частично упорядоченным.

Важным является тот факт, что нарушитель, выполняющий вторжение продвигается в своих действиях, реализуя (успешно или нет) различные атаки на систему. Тогда множество сигнатур может быть упорядоченным в соответствии с этапом вторжения следующим образом: $\{Sgt_{i_1}\}_{i_1=1}^{M_1} \ldots \{Sgt_{i_k}\}_{i_k=1}^{M_k}$, причем $\sum_{j=1}^{K} m_j = M$.

При этом сценарии нарушения безопасности (вторжения) могут быть описаны как $Scen = (Sgt_0, Sgt_1, \ldots, Sgt_k)$, $k \leqslant M$ при условии, что:

1. $\forall i, j \in 1 : k$, $i \neq j \to Sgt_i \neq Sgt_j$.
2. $\forall i, j \in 1 : k$, $i \leqslant j \to prp(Sgt_i) \leqslant prp(Sgt_j)$.

Автомат, описывающий нарушения безопасности системы, может быть представлен с использованием следующих понятий:

$\sigma = \{Sgt_m\}$ — состояния автомата, описываемые сигнатурой, соответствующей наибольшему достигнутому нарушителем этапу вторжения;

$t \in Sgt_i$ — управляющие символы автомата;

$prp(\sigma_i)$ для текущего состояния автомата — выход автомата;

$\sigma_0 \in \sigma$ — начальное состояние, в котором субъект начинает работу с системой.

Функция перехода δ автомата может быть описана так:

$$\forall t = Sgt_j \; \exists Sgt_i \in \sigma_i : prp(Sgt_j) \leqslant prp(Sgt_i) \to \sigma_{i+1} = \sigma_i;$$
$$\forall t = Sgt_j \; !\exists Sgt_i \in \sigma_i : prp(Sgt_j) \leqslant prp(Sgt_i) \to \sigma_{i+1} = \sigma_i \cup Sgt_i.$$

Функция выхода λ описывается следующим образом:

$$\forall t = Sgt_i \; Out = prp(\delta(\sigma_i, t)).$$

3 Объединенная модель

Компоненты, описывающие модель систему могут быть описаны с использованием единой структуры. В этой структуре состояния автомата будут описываться следующим образом: $\sigma = \{\{op_1(prog_1, O_1), \ldots, op_i(prog_i, O_i)\}, Sgt_m(Scen_l)\}$.

Входом объединенного автомата является операция, выполненная пользователем с использованием сервиса, или атака, выполненная пользователем на систему с использованием сервиса. Таким образом, $t \in (Op(prg_i, o_j) \vee Sgt_m)$ — управляющие символы автомата.

Функция перехода δ объединенного автомата описывается так:

$$\forall t = Sgt_j \; \exists Sgt_i \in \sigma_i : prp(Sgt_j) \leqslant prp(Sgt_i) \to \sigma_{i+1} = \sigma_i;$$
$$\forall t = Sgt_j \; !\exists Sgt_i \in \sigma_i : prp(Sgt_j) \leqslant prp(Sgt_i) \to \sigma_{i+1} = \sigma_i \cup Sgt_i;$$
$$\forall t = op_i \; \exists op_i(prog_i, O_i) \in \sigma_i = t \to \sigma_{i+1} = \sigma_i;$$
$$\forall t = op_i \; !\exists op_i(prog_i, O_i) \in \sigma_i = t \to \sigma_{i+1} = \sigma_i \cup t.$$

Выходом автомата является характеристика состояния объединенного автомата. В соответствии с определениями, приведенными выше, состояния могут быть: безопасными, небезопасными и состояниями с признаками атаки.

Функция выхода объединенного автомата λ описывается так:

$$\forall t = Sgt_i \; Out = prp(\delta(\sigma_i, t));$$
$$\forall t = op_i(op_i(prog_i, O_i) \in \sigma_i) \vee (op_i(prog_i, O_i) \in M'(s, prg, o)) \to Out = Sec;$$
$$\forall t = op_i(op_i(prog_i, O_i) \in \sigma_i) \vee (op_i(prog_i, O_i) \notin M'(s, prg, o))$$
$$\to Out = UnSec.$$

Таким образом, в настоящем докладе предлагается структура, позволяющая осуществить динамический мониторинг безопасности состояний системы.

Предлагаемый подход был использован в работах, выполненных в СПбГПУ:

- система антивирусного мониторинга «Основа»;
- система динамического анализа безопасности программного обеспечения «Мелкоскоп»;
- анализатор сетевых вторжений «Авгур».

В таблице 1 приведены аспекты модели, предложенной выше, применительно к разработанным системам.

Как показано в таблице, все разработанные системы, укладываясь в единую модель динамического мониторинга безопасности состояний системы, являются ее частными случаями вследствие особенностей назначения.

Таблица 1: Механизм в терминах предложенной модели

	Основа	Мелкоскоп	Авгур
Назначение	Система антивирусного мониторинга	Система динамического анализа безопасности программного обеспечения	Анализатор сетевых вторжений
Год разработки	2001	2004	2004
Управляющие символы	$\{Sgt_m\}_{m=1}^{M}$	$\{Op\}$	$Op(prg_i, o_j) \vee Sgt_m$
Состояние	Sgt_m	$\{op_1, (prog_1, O_1), \ldots, op_i, (prog_i, O_i)\}$	$\{\{op_1, (prog_1, O_1), \ldots, op_i, (prog_i, O_i)\}\}$
Выходные данные	Попытка перехода в атакующее состояние	Попытка перехода в запрещенное состояние	Попытка перехода в запрещенное или атакующее состояние
Источник входных данных (Prog)	Сменные носители, сетевые интерфейсы	Программы, выполняемые локально, системные вызовы	Сетевые интерфейсы
Обнаружение неизвестных атак	Нет	Да	Да
Обнаружение известных атак	Да	Да (без их идентификации по имени)	Да
Обнаружение некорректного использования системы	Нет	Да	Да

Внедрение средств контроля доступа в системы с архитектурой «тонкого клиента»

А. В. Коротич

Большинство современных информационных систем строится в соответствии с архитектурой «тонкого клиента». В таких системах за хранение данных отвечают сервера БД, за обработку данных — сервера приложений, в то время как клиентские приложения отвечают только за представление данных в виде, понятном пользователю. Средства защиты, включая средства контроля доступа, входят в состав программного обеспечения сервера приложений. С точки зрения безопасности такая архитектура имеет ряд недостатков:

1. Средства защиты отданы на откуп разработчикам прикладного приложения. Зачастую разработчики прикладных приложений жертвуют безопасностью системы, используя наиболее простые методы реализации бизнес-логики приложения.

2. Средства защиты сильно связаны с конкретным функционалом системы. Добавление новой функциональности и поддержка существующей приводят к существенным изменениям в реализации средств защиты и политике безопасности системы.

3. Политика безопасности таких систем с трудом поддается формализации и не может быть доказана на примере моделей безопасности. Сложность формализации политики безопасности объясняется невозможностью выделить средства контроля доступа в отдельный модуль и сформировать набор параметров, от которых зависит функция контроля доступа.

Для решения перечисленных проблем предлагается подход ко внедрению средств контроля доступа в системы с архитектурой «тонкого клиента». Смысл подхода состоит в вынесении средств контроля доступа на отдельный шлюз контроля доступа и связывании шлюза с компонентами системы таким образом, чтобы его присутствие было прозрачным для клиентов системы.

В такой системе средства контроля доступа становятся независимыми от прикладного приложения, и это позволяет существенно повысить безопасность информационных систем в целом.

Для того чтобы шлюз контроля доступа имел возможность осуществить проверку легитимности запроса к БД в соответствии с формализуемой политикой безопасности, запрос должен передаваться в таком виде, чтобы:

1. Можно было определить все параметры запроса, от которых зависит функция контроля доступа.

2. Формат запроса был стандартизован и широко используем в современных информационных системах.

3. Для разбора параметров запроса существовали бы свободно доступные парсеры.

В качестве протокола передачи данных, удовлетворяющего таким требованиям, был выбран SOAP. В последнее время протокол SOAP получил свое развитие в новой технологии построения информационных систем Web Services. Смысл SOAP сводится к передаче вызовов методов объекта с инкапсулированием параметров вызова в XML структуре.

Гибкость XML позволяет реализовать на базе SOAP широкий набор процедур обработки информации. Простота XML позволяет легко определять все параметры, необходимые для вынесения решения о легитимности запрошенного доступа. Открытость SOAP обеспечивает широкую поддержку данного протокола клиентскими платформами.

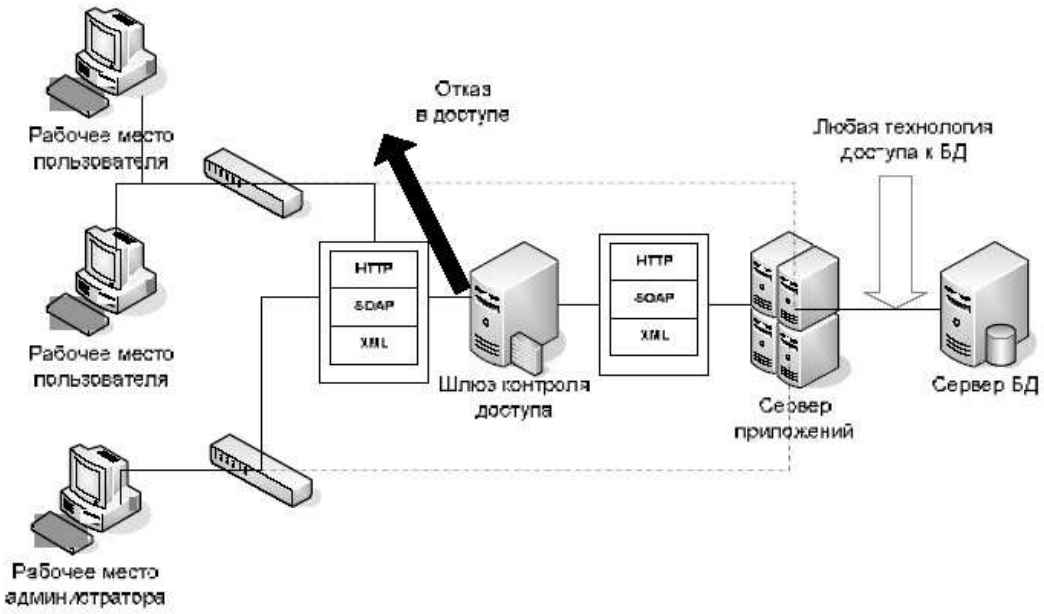

Рабочее место пользователя
Рабочее место пользователя
Рабочее место администратора
Отказ в доступе
HTTP
SOAP
XML
Шлюз контроля доступа
HTTP
SOAP
XML
Любая технология доступа к БД
Сервер приложений
Сервер БД

Рис. 1:

Часть V
Семинар-круглый стол «Информационная война и борьба с терроризмом: основные проблемы и международное сотрудничество»

Замысел проведения семинара-круглого стола «Информационная война и борьба с терроризмом: основные проблемы и международное сотрудничество»

Зачем нужен этот семинар?

Меры по обеспечению информационной безопасности являются важным составным элементом национальной безопасности. В прошлом десятилетии ведущие промышленные страны сделали значительные капиталовложения именно в эту сферу с целью обеспечить безопасность критически важных инфраструктур и поднять общий уровень понимания необходимости защиты национальных информационных систем. Что касается вооруженных сил, то операции, связанные с информационно-вычислительными сетями (оборона и нападение), активно развиваются, а информационные операции (ИО) стали уже обычными в военных доктринах современных вооруженных сил. Но все равно, несмотря на значительные инвестиции в сферу информационной безопасности, остаются открытыми вопросы:

- являются ли принятые меры достаточными?
- можно ли, используя эти меры, действительно защитить национальную безопасность?
- направлены ли они на защиту от «правильных угроз»?

Например, являются ли современные стратегии информационной безопасности эффективными или могут ли они в достаточной мере противостоять угрозам со стороны асимметричных акторов (таких как «террористическая» группа)? Ведь асимметричные акторы отличаются тем, что не следуют установленным правилам или договоренностям о ведении боевых действий (а таким образом возникают трудности с выбором ответных мер или целей). Часто успех их операций и выживание зависят от использования неожиданных и непредвиденных возможностей, поэтому порой для достижения своих целей они используют «неконвенционные» подходы (порой для проведения психологической атаки, а не нанесения материального ущерба). Много внимания уделяется так называемому сценарию «электронный Перл-Харбор». В соответствии с этим сценарием произойдет заранее спланированный массовый захват критически важных информационных систем, энергетических систем или системы управления полетами. Пока этот сценарий есть только в теории. В то же время асимметричные акторы использовали методы информационной войны и другие «неконвенционные» стратегии для достижения своих целей. Информационно-телекоммуникационные технологии используются в целях получения финансирования, обеспечения командования и контроля, для проведения пропагандистских кампаний, а также для поддержки диаспорных и других социальных групп сочувствующих и сторонников. На самом деле, традиционные подходы обеспечения информационной безопасности могут оказаться совсем не эффективными, если эти акторы используют «неконвенционные» методы, в том числе и информационно-телекоммуникационные технологии. Но самое плохое — это то, что многие подходы не полностью понимают суть проблемы и поэтому будут создавать в киберпространстве электронный эквивалент «линии Мажино».

В то же время неконвенционные угрозы безопасности не связаны исключительно с вызовами террористических или криминальных группировок, или с конкуренцией между государствами. Доктрины информационной безопасности должны учитывать глобализацию информационно-телекоммуникационных систем, которые выходят из под контроля на национальном уровне. Разработка и создание систем информационных технологий и программного обеспечения постепенно выходит на транснациональный уровень и в их основе порой лежат не патриотические чувства, а скорее желание получить

выгоду. Часто получается так, что программное обеспечение, которое используется на правительственном уровне, было создано с другой целью, а иногда из-за проблем с правами на интеллектуальную собственность это программное обеспечение легко найти или даже посмотреть. Порой в частном секторе просто нет необходимой профессиональной компетенции для обеспечения этого соответствия, а для многих стран непосильным является разрешение проблем, связанных с правом собственности на программное обеспечение или комплектующие. Более того, стратегическое значение этих технологий для технического сбора информации разведывательными службами огромно, но эти проблемы могут стать помехой для информационной безопасности страны и могут определить «неофициальный» рубеж в области международного сотрудничества.

Эти проблемы очень сложны, но, понимая их центральность и важность для безопасности, необходимо над ними поразмышлять и проанализировать. Несмотря на то, что довольно легко адаптировать парадигму безопасности в новой области, гораздо труднее выйти за рамки соглашения и понять необходимость выдвижения новых парадигм. В эпоху, когда технологическая зависимость и взаимозависимость становятся важными аспектами национальной безопасности, мышление, выходящее за рамки традиционных границ, является важным и необходимым осуществлением «должной осмотрительности». В то же время по причине уязвимости, присущей медленно адаптирующимся акторам (таким как государства), и возможностей асимметричных акторов, не требующих затрат, международные семинары, на которых обсуждается «неконвенционная» сторона информационной войны, проводятся редко, впрочем это касается и многодисциплинарных дискуссий, на которых собираются ученые и эксперты.

Цели семинара

Цели этого семинара — дать возможность профессионалам и ученым из стран НАТО и СНГ сфокусироваться на трех основных темах:

- определить угрозу кибертерроризма в контексте сегодняшней войны с терроризмом;
- оценить значимость других форм «неконвенционных» угроз национальной информационной безопасности;
- сравнить опыт, меры борьбы и «извлеченные уроки» в сфере международного сотрудничества в области борьбы с угрозами информационной безопасности.

Это специальный семинар по этой теме, на котором соберутся профессионалы и ученые из стран НАТО и СНГ.

Terrorism and democracy

J. Ryder

I've been asked to say a few words about the relation of terrorism and democracy — a rather large topic, so I will limit my remarks to one aspect of the problem, which are the potential threats to democracy of efforts to defend against and combat terrorism. Couple points on background first:

1. These remarks will consider the question, consider the problem from the American perspective. I will make some judgments about the relation of terrorism or struggle against terrorism in relation to democracy that bear on American context. It would be interesting to have others of you to comment how this might bear on in Russian context.

2. The details of this kind of analysis, certainly if we will develop this kind of analysis more fully, invariably will depend on what we mean by democracy. Do we mean something simply formal or procedural, do we mean something more substantial. In some of my work I made a distinguish between "thin" democracy and "thick" democracy. But basically one can understand democracy as primarily a formal structure whereby there are elections are more or less free and open and that is it. And there are many people including many in the leadership in the United States who regard democracy this way. But that is a rather thing that may of us could have and many of us do have a much a deeper, thick notion of what democratic society, democratic situation consists of and depending on which of those and others within that range or range between them which use of the democracy it takes when we draw different conclusions.

But while those background points are being made let me divide these remarks into two parts. One has to do with democracy and terrorism in internal/national circumstance and then external that is with prospect to foreign affairs/foreign policy.

With respect to internal situation (and remember the context is here American) Bush' Administration policies after September, 11 have run rough shard over a number of democratic rights and liberties. In generally had thrown into question for a lot of Americans and others around the world what just democracy means we name democracy we undermine a good deal of it. And just to give a quick example for those of you who are familiar with this kind of ethnic profile on development policies to try to identify the potential terrorist before they are able to organize, before they are able to strike, we will regenerate the opinion based on physical appearance of means or of national origin for example. The suspension of rights of the accused which is a controversial matter in my own country, question of access of people who have been arrested for a suspicion in terrorist activities or support of terrorist activities, the access to defense attorneys, by extension defense attorney's access to information that prosecution has, that government has in various cases. Some of these issues as you probably know have gone into our own federal courts and even the Supreme Court. Bush administration has been reinvented in some cases and compelled by the courts to readjust its behavior because much of what it has tried to do has undermined too much of what we regard to as democratic rights. There were issues of surveillance. One of the characteristics of the first version of the Patriot Act that was passed right after 9/11 as many people has judged to was that the government assumed to itself the right to have access to people's reading habits, for example, in libraries and so on. And this made a lot of people nervous as you might imagine since we have generally operated on the assumption people were entitled to read what they please and was none of the government's business terrorism or no terrorism. There were many other matters, like the list kept of many organizations as terrorist organizations and what damage that does to the question of the preemptive association. The tension here accords is between liberty and security. I don't pretend that it is a tension easily resolved. If it were there wouldn't be much of a tension. This is a serious problem. And it is complicated by the fact that in a less or more democratic society this is quite possible for the majority for people in the name of defending their own liberties and democratic rights to sacrifice these liberties in the name of security. The problem here is a version of what Tokwill and John Stuart Mill and many others in 19th–20th centuries have called integrity majority. It is not he majority of citizens who are suspected of terrorist activity, it is a very small minority but one can make the claim and

many have that the democratic character of the society is marked not so much by the rights and liberties that he majority gives itself because that is rather easy. The democratic strength of the society is characterized by the democratic rights and liberties of the majority accords to everybody and in a case where people begin to run scared when they are nervous about security; it is quite easy for the majority to sacrifice their rights and liberties of a minority. And that is a very serious threat to a democratic character of a society. So the relation between these two that is terrorism and democracy, in internal context is a rather serious problem to which I have no answer.

If we turn to external question, again we do see that the American Administration Foreign Policy in the name of fighting terrorism turned out to the fundamentally credit in number of important aspects. The general question here is whether democracy can be opposed/imposed. And again we need to considerate the issue of what we mean by democracy. Let me now say a word before going in it about this term we use the thick conception of democracy. I think it is worth taking seriously and it does have a variant on what we end up thinking on the relation between democracy and terrorism.

A thick democracy is a democracy that is far more than simply system of political parties, competing in open and periodic elections, that is more than just a formal procedure of political democracy. What I mean by a thick democracy is:

1) the society which individuals, groups or communities consciously communicate in pursue common interests with others;

2) a society which is driven not by ideology but a willingness to experiment with new ideas and solutions to social problems. It is a society not driven by ideologically but experimentally in its approach to problems;

3) a society that communicates with, collaborates with in pursue common interests with those beyond its borders. For foreign policy situation, for external situation that may be the most crucial consideration. A democratic society is the one that makes a sustained an effort to communicate, collaborate and find points of commonality with those beyond its boarders.

In pursuing its policies in Afghanistan and Iraq and its interactions with Iran, Korea, China, Russia and many other nations the US currently behaves more like empire power rather like democracy. And again I am not the first to make this observation, there were couple very interesting books written on this topic only last year. A couple were arguing the virtue of American democratic imperialism. But I will argue that imperialism and democracy, certainly democracy in the thicker sense, are not compatible. So to pursue one is undermine the other. I would also argue that it is possible in fact to behave internationally in such a way as to advance democracy even in its thicker sense and it would be a better part of wisdom of our international community to begin to think in terms of maintaining democratic values as perhaps in a long run the best way we have to undermine the foundations of terrorism and to defend ourselves against it.

О доктрине создания сети региональных информационно-психологических поясов безопасности

В. И. Таирян, Е. И. Таирян

Современный этап развития общества характеризуется возрастающей ролью информационной сферы, представляющей собой совокупность информации, информационной инфраструктуры, субъектов, осуществляющих сбор, формирование, распространение и использование информации, а также системы регулирования возникающих при этом общественных отношений. Информационная сфера, являясь системообразующим фактором жизни общества, активно влияет на состояние политической, экономической, оборонной и других составляющих безопасности государств и их национальная безопасность существенным образом зависит от обеспечения информационной безопасности, и в ходе технического прогресса эта зависимость будет возрастать. Под информационной безопасностью государства понимается состояние защищенности его национальных интересов в информационной сфере, определяющихся совокупностью сбалансированных интересов личности, общества и государства [1, 2].

Безопасность любого отдельно взятого государства не может не зависеть от региональной стабильности, то есть необходимо рассматривать проблему безопасности государства в рамках региональной безопасности. С другой стороны, события последних лет привлекают внимание к такой серьезной угрозе для региональной безопасности как международный терроризм со всеми его проявлениями [3, 4, 5, 6, 7, 8].

Последние резолюции ООН по проблеме борьбы с международным терроризмом призывают мировое сообщество очень серьезно отнестись к этой проблеме и мы считаем, что наиважнейшей из проблем борьбы с международным терроризмом является проблема выявления причин и очагов международного терроризма и информационно-психологическая их нейтрализация [9].

Мы полагаем также, что для решения этой проблемы международные структуры коллективной безопасности должны располагать доктриной создания сети региональных информационно-психологических поясов безопасности, доктриной, представляющей собой совокупность научных подходов к обоснованию цели, задач, принципов и основных направлений информационно-психологического содействия обеспечению региональной безопасности.

Сеть региональных информационно-психологических поясов безопасности должна содействовать:

- обеспечению региональной безопасности и стабильности;
- выявлению, локализации и нейтрализации причин и очагов международного терроризма.

Региональный информационно-психологический пояс безопасности, как нам представляется, может быть рассмотрен как международная многоуровневая (духовно-культурный, интеллектуальный, технологический и физический уровни) программа, политические, социальные, правовые и научно-технические мероприятия которой направлены против международного терроризма и служат содействию региональной стабильности.

Рассмотрим на примере Южного Кавказа некоторые политические аспекты необходимости создания регионального информационно-психологического пояса безопасности.

> *«Балканы и Малая Азия занимают самую важную стратегическую позицию в мире. Они представляют собой ядро и Центр Старого Света, разделяют и одновременно связывают три материка: Европу, Азию и Африку... Они расположены в месте, откуда можно угрожать и вести нападение против трех континентов.»*
>
> Дж. Бакер.

Десятилетия понадобились, чтобы очевидная истина, высказанная английским политологом в XX веке, стала столь же очевидной и жизненно важной сегодня.

Поэтому не может являться неожиданным то, что Южный Кавказ и Центральная Азия объявлены приоритетной зоной НАТО, а страны Южного Кавказа включены в программу Евросоюза «Расширенная Европа: Новые соседи». Только вот возникает закономерный вопрос: насколько эти шаги адекватны обеспечению безопасности Европы уже в ближайшие годы.

Вполне понятно, что регион Южного Кавказа есть ни что иное, как оборотная сторона Балкан: ряд неразрешенных межгосударственных конфликтов, разрушенная экономика и, как следствие, социальная напряженность внутри стран, разноконфессиональные и этнонациональные противоречия.

Усилиями самих этих государств противоречия, в конечном счете, будут разрешены, пусть даже на это уйдут многие годы, но соответствует ли такая ситуация интересам России и Европы — при тщательном рассмотрении геополитических процессов ответ отрицателен и вот почему.

Конечно же, Россия прекрасно понимает стратегическую значимость региона и как свое «подбрюшье», и как «критическую» опорную точку для всего Евразийского континента, но она «уходит» из региона.

«Присутствие» России в регионе создает иллюзию присутствия, как для прикрытия второстепенных интересов, так и для «ублажения» Европы, не решающейся взять на себя ответственность за урегулирование конфликтов в регионе и создание в регионе такой единой системы безопасности, которая не только не позволила бы в будущем использовать регион в качестве плацдарма против Европы, но и превратить его в важнейший бастион Европы.

Формирование двух геополитических и геостратегических макрорегионов — Большой Европы и Большого Ближнего Востока имеет два нежелательных для Европы сценария.

1. «План Гобла», или ТУРАН.

Такой план поначалу мог бы быть реализован в рамках осуществляемой США глобализации, но: во-первых, увязнув в Ираке США более не удастся завершить свой «Мегапроект» блицкригом в Иране и тем самым создать оптимальные условия для выхода Турции через Южный Кавказ на Центральную Азию, чтобы получить из рук США «эстафетную палочку». А во-вторых, события в Ираке позволили США понять, наконец, очевидную истину в отношениях в Турцией — двойная макрорегиональная принадлежность Турции совсем не означает всего две модели ее политического поведения. Турция слишком сильный и агрессивный «зверь», чтобы всегда «слушаться» дрессировщика, если ей покажется, что дрессировщик жирнее и вкуснее предлагаемого ей кролика, он не будет медлить с выбором.

Аналогичные «просчеты» США в Израиле и в Пакистане позволяют придти к выводу, что если даже США не откажутся от реализации тюркизации региона, то осмысление и переоценка своих реальных возможностей и увязка их с реальностями, поиски новых партнеров потребуют у США слишком много времени, которого у них уже нет.

2. «Приход» Китая в регион Южного Кавказа.

Хотя США, как в хорошо разрекламированном шоу, и удалось, проведя ряд военных операций в Афганистане, по сути решить некоторые из своих стратегических задач по взятию под контроль Центральной Азии и в перспективе временно купировать процесс продвижения Китая на Восток, на этом все и закончилось.

Что это — антракт или финиш?

Если и антракт, то он тоже затянется на годы, а набирающий силы Китай этого времени не даст никому. Тем более, что перспектива восстановления «Поднебесной» уже не призрак, и неясно пока только, где будет возводиться новая «Великая Китайская стена».

Гибнущий коммуно-космополитизм в России дал Китаю великий импульс к организации. Легко переварив масонскую идеологию космополитизма отрыжкой в 50 миллионов человеческих жизней в культурной революции, Китай заимствовал из всех систем лучшее: систему идеологии, систему организации, систему производства, систему проведения массовых операций с глубиной воздействия на сознание масс и опыт создания психокомплексов поведения с воздействием из подсознания.

И как результат Китай уже сегодня самодостаточен во всех сферах и обладает более чем достаточными ресурсами для достижения практически любых целей.

И здесь надо отметить, что не следует ожидать от Китая громких шоу, аналогичных тем, что демонстрирует США. Китайского «вторжения» не будет, оно уже давно началось - заимствовав у тюрок метод «ползучей агрессии», китайцы под видом беженцев, мигрантов, мелких торговцев расползаются по всей планете, но сосредотачиваются, в основном, там, где это функционально соответствует главным интересам Китая (США, Европа), и в нужное время проявляются не как граждане, например,

США, а как патриоты Китая; примером этому могут быть события, когда на финансовую атаку фонда Сороса против юаня, китайцы ответили не только грамотной защитой национальной валюты, но и, использовав свою диаспору в США, провели ответную атаку и практически разорили фонд Сороса.

Вот почему Китай не спешит: время его главный союзник.

Теперь, предположив, что нам удалось привести убедительные доводы для России и Европы в пользу принятия ими более радикальных мер и усилий в отношении Южного Кавказа, попробуем обратить внимание на то, а с чего же начать?

Информация формирует главнейшую составляющую типа мышления человека, его психику и сознание, состояние души и формулу нравственности, отношение к реальности и к абстракции, материальному и духовному, земному и божественному. Следовательно, при определении начального звена в цепи акций по созданию зоны безопасности южнокавказского региона, необходимо учесть, что духовность воздействует на сознание, сознание формирует мысль, мысль выражается словами и поступками, что в итоге и реализуется интегральными действиями, определяющими потенциал субъекта, страны и региона.

Таким образом, этим начальным (по сути фундаментальным) звеном может и должно быть создание информационно-психологическогого пояса безопасности общего и единого для всего Южного Кавказа, а в перспективе увязка его с системой безопасности Ирана и вот почему!

Иран. Заложив историческое мышление у своего народа, и воздействуя устойчивым моделированием на сознание, иранская элита неуклонно возрождает в своем народе забытые ононимы, что и является конечным элементом в укреплении связи времен в сознании нации. И, как результат, в нужное время миллионы иранцев по призыву муллы в одночасье превратятся в «шахидов».

Вот где находится главный «аргумент» Ирана, когда он в диалоге с США подчеркивает, что Иран — это не Ирак.

И хорошо, что «мудрая Европа» предпочитает расширять торговые и другие отношения с Ираном, взамен приглашения США совместно искать там ядерные арсеналы.

Такая дальновидная позиция Европы может быть поддержана Россией созданием системы, включающей в себя информационно-психологические пояса безопасности Южного Кавказа и Ирана, согласованные между собой и соответствующие общим интересам.

Особо отметим, что исключительно важным для создания такой системы является построение адекватной ей математической модели и проведение сетевых компьютерных экспериментов, что позволит дать оценку и прогноз соответствующим международным процессам.

Литература

[1] В. П. Шерстюк. Проблемы информационной безопасности в современном мире. 18.04.2003

[2] Обеспечение информационной безопасности России. Теоретические и методологические основы. А. А. Стрельцов. М., МЦНМО, 2002 (избранные материалы книги). 12.02.2003

[3] А. А. Сальников, В. В. Ященко. Методологические проблемы противодействия кибертерроризму. 26.03.2004

[4] Е. Н. Мощелков, В. А. Носов. Деятельность Рабочей группы Консорциума «Воздействие информационных технологий на национальную безопасность» в 2002–2003 г. (на англ. яз.). 25.06.2003

[5] А. В. Манойло. Информационно-психологическая война как средство достижения политических целей. 2.01.2004

[6] А. В. Крутских, И. Л. Сафронова. Международное сотрудничество в области информационной безопасности. 18.04.2003

[7] А. В. Манойло, А. И. Петренко. Информационно-психологическая безопасность системы социально-политических отношений современного общества. 5.03.2004

[8] А. В. Крутских. Война или мир: международные аспекты информационной безопасности. 26.03.2004

[9] Tairyan Vasiliy. Humanitarian problems of the information security. In Proc. of the NATO ASI — Armenia, Nork, Armenia, 2005 (to be published).

Образовательные аспекты обеспечения информационной безопасности в условиях усиления терроризма

А. Н. Курбацкий

Задачи построения информационного общества, возрастание роли информационных ресурсов, роли информации в развитии общества и государства заставляют рассматривать проблемы информационной безопасности как важнейшие проблемы переднего плана. Переход информации в разряд важнейших ресурсов общества приводит к активизации борьбы за обладание этим ресурсом. Следствием этого стало резкое усиление значимости информационных войн и информационного оружия. Они получили достаточно неожиданное развитие в начале 21 века, после резкого усиления иррациональности терроризма. Стала очевидной своего рода демократизация деструктивных информационных технологий.

Терроризм заставляет общество балансировать между свободой и безопасностью. Еще в XVIII веке Бенджамин Франклин писал: «Тот, кто не отказывается от основ свободы ради временной безопасности, не заслуживает ни свободы, ни безопасности». Быстро растущая опасность со стороны международного терроризма подрывает авторитет государства. В частности, видя неэффективно обороняющееся от международного терроризма государство, граждане перестают видеть в государстве главную и незаменимую форму общественной организации. Особенно опасно, когда это смыкается с усиливающейся сейчас тенденцией этнического самоутверждения всех возможных видов. Эти процессы, как правило, обосновываются «историческими исследованиями прошлого, языка, границ». Текущие мировые события и тенденции подтверждают правоту У. Черчилля: «Если мы будем сражаться с прошлым, мы потеряем будущее». Часто основной формой этнического самоутверждения является терроризм (национальный терроризм). Уже стали классическими примерами деятельность Ирландской республиканской армии (ИРА) в Великобритании, Фронта национального освобождения Корсики во Франции, «Эускади та Аскатасуна» (ЭТА) в Испании, албанский терроризм на Балканах и т. д. Практически становится невозможным однозначно отличать террористов от борцов за свободу. Возможно, не так уж не прав испанский философ Хосе Ортега-и-Гассет, размышляя о национальных государствах: «Итак, ничего другого не остается, как покончить с давним и привычным передергиванием в вопросах национального государства и привыкнуть смотреть на трех китов [язык, кровь, родная земля], на которых якобы держится нация, как на изначальные помехи ее возникновению... Надо отважиться видеть разгадку национального государства в том, что присуще ему именно как государству, в самой его политике, а не посторонних началах биологического или географического свойства».

На усиливающуюся, особенно после распада СССР, тенденцию этнического самоутверждения в Европе наложился процесс культурного и религиозного самоутверждения стремительно растущей исламской диаспоры, поскольку культурной ассимиляции мусульман западной цивилизацией даже непосредственно на своей территории так и не произошло.

Европейская (да и вся западная в целом) цивилизация оказалась практически бессильной перед схемой, по которой инициатива, подготовка и непосредственное исполнение терактов осуществляется по сути изнутри. Теракты летом 2005 года в Лондоне в явном виде эту схему проявили и продемонстрировали, что существует реальная внутренняя инфраструктура и спрос на теракты.

Американские исследователи из вашингтонского исследовательского института Nixon Center выяснили, что самая крупная исламская террористическая группировка в Европе и Северной Америке зародилась не на Ближнем Востоке или развивающихся странах, а на самом Западе. В основу исследования была положена база данных, в которую было занесено около 400 террористов, которые за период с 1993 по 2004 год были привлечены к суду, осуждены или убиты в Европе и Северной Америке. На основе базы данных сделаны следующие выводы. Менее половины зарегистрированных террористов родились на Ближнем Востоке. 41% имеют гражданство стран ЕС или США, а 36% — стран Магриба.

Только 17% террористов были родом из Ближнего Востока, 3% родом из азиатских стран.

Мы имеем ситуацию, когда реальные и потенциальные террористы — не отчаявшиеся беднейшие мусульманские слои, а достаточно образованные молодые люди. И опаснее всего, что массовая подпитка международного терроризма в скором времени возможна не только из мусульманской молодежной среды, а из пока еще традиционной западной в широком смысле молодежной среды. Поясним, почему такая тенденция на наш взгляд возможна.

Как известно, терроризм никогда не существует в вакууме. Даже средневековые организации, которые можно считать в определенной мере предтечами современных террористов, такие как исмаилиты, старались привлекать внимание тогдашнего общества к своим действиям — например, они объявляли на базаре об очередном убийстве, совершенном ими. Уже в прошлом веке газеты были абсолютно необходимы террористам, так как терроризм всегда «взаимодействует» со структурами общества и государства.

Международные террористы с большой эффективностью для себя сумели задействовать *информационное поле.*

Сегодня теракты в «традиционном» смысле используются лишь как *элементы информационных технологий*, и собственно теракты уже не являются целью и задачами терроризма. При этом наблюдается рост «качества» терактов, именно, как *информационных продуктов*, благодаря чему теракты утрачивают свою изначальную роль инструмента локального действия, и переходят в разряд инструментов *глобального управления.* Произошли принципиальные изменения в составе перечня угроз национальной безопасности, а именно резкое снижение порога устойчивости общества к информационным воздействиям — как внешним, так и внутренним.

Мы не можем жить без газет и телевидения, Интернета, но, глядя на многократно повторяющиеся «новостные телевизионные живые картинки» после терактов (вспомним освещение событий 11 сентября 2001 в Нью-Йорке, взрывов в метро и «Норд-Ост» в Москве, трагедии в Беслане, взрывов в Мадриде и т. д.), возникает ряд резонных вопросов:

- Само информационное общество и СМИ, как его главный инструмент, не «провоцируют» ли стихию терроризма, живущего только тогда, когда его «освещают»?
- Не являются ли порой СМИ пособниками террористов в создании неадекватного страха?
- Всегда ли СМИ выступают против терроризма, если действия террористов являются красивым информационным поводом для удовлетворения стремления некоторых СМИ к производству зрелища? Теракт является зрелищем, а зрелище — оторванное от содержания — является приоритетной стихией СМИ.

Террористические организации пытаются навязать обществу новую модель регулирования экономической и политической жизни, в которой информационное и физическое насилие над личностью становится основным *методом управления* обществом и его институтами. И все это сопровождается «живыми картинками в СМИ». Даже Збигнев Бжезинский в своей статье в «Los Angeles Times» от 11 октября 2005 года пишет «Террористами не рождаются, а становятся — под воздействием конкретных событий, личного опыта, представлений, фобий, национальных мифов, исторической памяти, религиозного фанатизма и сознательного "промывания мозгов". Ими становятся под влиянием "телевизионной картинки"...».

А теперь посмотрим, чем массово занимаются дети школьного возраста. В Интернете сейчас достаточно активно дискутируется вопрос: Кем могут вырасти наши дети, растущие в условиях технологий 21 века?

Молодежь все больше времени проводит за компьютером, предпочитая виртуальный мир реальному и в существенной степени худший виртуальный мир, создаваемый агрессивными компьютерными играми. «Живые картинки» с экранов телевизоров и компьютерных мониторов практически совпадают. Убийство и насилие становятся нормой жизни.

Существование зависимости от компьютерных игр, аналогичной наркотической, пока еще вызывает сомнения у многих ученых и специалистов, но несомненно то, что количество молодежи, увлекающейся нахождением в виртуальной реальности растет катастрофически быстро. Человек, находясь длительное время в такой среде, переносит ее законы на реальный мир: начинает чувствовать себя более уязвимым. Многие психологи подтверждают, что игры, где присутствует насилие, формируют

агрессивность в сознании человека. При этом такая агрессивность может проявиться не сразу, а постепенно накапливаясь, через некоторый временной промежуток, иногда и достаточно длительный. Взрослый человек сегодня пока еще способен более-менее четко разделять виртуальные и реальные миры. Но дети, которые еще не в достаточной степени получили представление об окружающем их реальном мире, такую способность сейчас приобретают все меньше. Начинается подмена нравственности. И действия в виртуальном мире переносятся в реальность. И тогда — массовые расстрелы прохожих маньяками, необъяснимые самоубийства и так далее. Практически даже стадии формирования компьютерной (виртуальной) зависимости схожи со стадиями «привязки» к наркотику.

Современный терроризм — это распределенная система *сетевого типа*. Этим в существенной степени обусловлена и трудность борьбы с терроризмом, поскольку сетевые системы невозможно уничтожить «точечными ударами»: они способны к самовосстановлению. Здесь напрашивается аналогия с сетевым принципом многих популярных агрессивных компьютерных игр.

Последствия сегодняшних упущений в образовании и воспитании могут проявиться гораздо раньше, чем мы думаем.

К большому сожалению, зачастую информационные технологии в образовании внедряются лишь для того, чтобы шагать в ногу со временем, а не для решения тщательно спланированных образовательных задач. Во многих случаях те, кто занимается компьютеризацией школьного обучения, тратят слишком много сил и средств на само компьютерное оборудование и подключение к Интернету, уделяя слишком мало внимания профессиональной подготовке и поддержке тех же учителей. И скучные занятия в школах заменяются внешкольными игровыми, массово низкого качества.

Практически мы пока не имеем массового формирования молодежной культуры на основе применения информационных и коммуникационных технологий, которая способствовала бы применению в реальной жизни знаний и навыков, полученных молодежью в виртуальной среде.

Что такое безопасность? Что такое угроза? Что такое терроризм?

В. И. Мунтиян

Информация является самым ценным глобальным ресурсом развития человечества и государства. Наша жизнь, мышление, сознание, интеллект — это информация. Вселенная, жизнь, разум построены на информационной основе.

Информация — это основной вид стратегического оружия государства и основной ее ресурс.

Существуют три глобальные угрозы катаклизма для человечества в целом:

- термоядерная катастрофа;
- экологическая, техногенная катастрофа;
- информационный коллапс цивилизации.

Без преувеличения, к глобальной угрозе можно отнести терроризм. Сегодня не существует ни одной страны, нации, народа, где была бы обеспечена абсолютная безопасность от терроризма.

Эта угроза сегодня вышла на планетарный уровень. И для ее нейтрализации необходима концентрация усилий всего мирового сообщества. Если суммировать все угрозы, начиная от экологических, заканчивая угрозами техногенного характера, можно сделать вывод, что человечество находится на «пороховой бочке». Но спичкой, которая может привести к взрыву этой «пороховой бочки», может стать терроризм. Вот такая простая по форме, но страшная по существу, угроза существованию человечеству от терроризма. Неправильной будет политика по решению данной проблемы, если винить только слабо развитые страны в порождении источников терроризма. Свою негативную лепту к разрастанию этой угрозы внесла эгоцентрическая политика развитых стран Запада по отношению к остальным странам мира. Она породила неравенство между богатыми и бедными, дисбаланс между научно-техническим прогрессом и духовным развитием человечества. Нарушение 8 буферных зон, которые обеспечивают безопасность планеты, спровоцировало необратимые процессы биосферной устойчивости. И вместо того, чтобы решать проблемы мирного существования, экологической, энергетической, демографической безопасности, преодоление бедности и распространения болезней, вместо этого человечество каждый год тратит 1 трлн. долларов на военные расходы. Существует пропорция: чтобы поразить 1 квадратный километр обычными средствами поражения, необходимо 2 тысячи долларов США расходов, ядерным оружием — 800 долларов, химическим — 600, биологическим — 1 доллар. А посредством кибертерроризма, для этого необходимо всего лишь от 5 до 10 центов.

Поэтому, сегодня информационный ресурс необходимо направлять не на информационные войны, а на разработку и внедрение моделей мирного существования человечества между собой и в гармонии с природой.

Информация осложняется по качеству. Растет количество ее источников, с различной целью влияя на человека. Рассмотрим две стороны медали информационной революции. Первая — это положительные факторы влияния информации на развития мирового сообщества, а вторая — отрицательные факторы, когда самые современные информационные технологии и ресурсы, попадая в руки террористов, представляют огромную опасность, которая несет в себе горе и человеческое страдание. По нашему мнению, решение этой сложнейшей для современного человечества проблемы лежит в области знаний информационной сферы.

Именно в информационной сфере можно установить причинно-следственные связи угрозы терроризма, и в ней же находятся методы и механизмы ликвидации и нейтрализации этих угроз еще на самых ранних стадиях зарождения. Проблема обеспечения безопасности от угроз терроризма многогранная. Но именно информация является одновременно первопричиной, оружием и защитой. Поэтому, она в первую очередь должна стать объектом научных исследований, что и подтверждает конференция, участниками которой мы с вами являемся.

Наша цивилизация, которая руководствуется ежеминутными потребительскими приоритетами и развивается стихийно, в целом неумолимо приближается к границе бифуркации, где человечество должно определить: или стремительная экологическая катастрофа (гибель), или новые приоритеты и принципы гармонизации с природой.

Возрастает роль интеллекта (научной мысли) в развитии цивилизации. В точках бифуркации необходима определенная величина «критической массы» интеллекта для предотвращения информационного и экологического коллапса.

Все большее значение в сфере политических интересов получают информационные факторы. Экономический потенциал также в наибольшей степени определяется объемом информационных ресурсов и уровнем развития информационной инфраструктуры. Одновременно возрастает уязвимость экономических структур от недостоверности, несвоевременности и незаконного использования коммерческой информации, промышленного шпионажа и хакерства. Возрастание роли информационно-психологической безопасности обусловлено также усилением угрозы использования информационного оружия в международном информационном обмене. Это предопределяет необходимость решения проблем, связанных с возможностями информационной войны, отрицательного информационного влияния на индивидуальное и общественное сознание, психику людей, на компьютерные сети и другие информационные системы.

Информационная борьба прежде имела место практически во всех военных действиях, проявляясь в таких основных формах, как ведение разведки и противодействие ей, распространение дезинформации, слухов и борьба с ними.

Информационную борьбу можно определить, как борьбу сторон за завоевание преимущества по количеству, качеству и скорости получения информации, ее своевременного анализа и использования.

Под информационной войной понимают действия, направленные на достижение информационного преимущества в национальной военной стратегии путем влияния на информацию и информационные системы противника при одновременном обеспечении безопасности и защите собственной информации и информационных систем. К особенностям информационной войны следует отнести следующие:

- перехват всех видов информации и информационных систем с отсечением информации от среды использования;
- объекты могут быть и оружием, и объектом защиты;
- распространение территории и пространства ведения войн как при объявлении войны, так и в кризисных ситуациях в разнообразных сферах жизнедеятельности;
- ведение войны как специальными военными частями, так и гражданскими структурами.

Концепция информационного противостояния, по оценке специалистов, предусматривает:

- подавление (во время военных действий) элементов инфраструктуры государственного и военного управления (уничтожение центров командования и управления);
- электромагнитное влияние на элементы информационных и телекоммуникационных систем (радиоэлектронная борьба (РЭБ));
- получение разведывательной информации путем перехвата и дешифровки информационных потоков, передаваемых каналами связи, а также побочными излучениями с помощью специально установленных в помещениях электронных приборов; технические средства перехвата информации (радиоэлектронная разведка);
- осуществление несанкционированного доступа к информационным ресурсам (путем использования программно-аппаратных средств прорыва систем защиты информационных и телекоммуникационных систем противника) с последующим их искривлением, уничтожением, хищением или нарушением нормального функционирования этих систем (хакерская война);
- формирование и массовое распространение по информационным каналам противника или глобальным сетям информационного взаимодействия дезинформации или тенденциозной информации для влияния на оценку, намерения и ориентацию населения и лиц, принимающих решения (психологическая война);

- получение необходимой информации посредством перехвата и обработки открытой информации, передающейся по незащищенным каналам связи и циркулирующей в информационных системах, а также публикующейся в средствах массовой информации.

Информационное оружие, как и информационное противостояние, в процессе развития общества и информационных технологий подвержено изменениям. В современной практике под информационным оружием (ИО) понимают средства уничтожения, искажения или похищения информационных объемов, получения из них необходимой информации после преодоления систем защиты, ограничения или запрещения доступа к ним, дезорганизацию работы технических средств, выведение из строя телекоммуникационных сетей, компьютерных систем, всего высокотехнологического обеспечения жизни общества и вмешательство в функционирование государства.

К видам информационного оружия можно условно отнести пять соответствующих совокупностей или группы средств, которые могут быть применены для деструктивных (дезориентирующих, дезинформирующих, дезорганизирующих, дестабилизирующих, разрушающих, подавляющих и др.) информационных влияний на содержательные компоненты стратегических систем управления:

- средства массовой информации (СМИ — радио, пресса, телевидение) и агитационно-пропагандистские средства (видеокассеты, электронные учебники и энциклопедии и др.);
- психотронные средства (специальные генераторы, специальная видеографическая и телевизионная информация, видео;
- средства наноинформационных технологий (типа «Виртуальная реальность» и др.);
- электронные средства (оптико- и радиоэлектронные средства) — специальные передаточные устройства и излучатели электромагнитных волн и импульсов;
- электронно-вычислительные средства — «компьютерные вирусы», разрушающие программные закладки и др.;
- лингвистические средства (языковые единицы, «специальная» терминология, речевые обороты, которые имеют семантическую неоднозначность при переводе на другие языки и др.);
- психотропные средства (специально-структурированные лекарства, психо-фармакологические средства, транквилизаторы, галлюциногены, наркотики, алкоголь и др.).

От обычных средств поражения информационное оружие отличают следующие признаки:

- секретность, то есть возможность достигать цели без имеющейся подготовки и объявления войны;
- масштабность — возможность наносить непоправимый вред, не нарушая национальных границ и суверенитетов, без привычного ограничения пространства во всех сферах жизнедеятельности человека;
- универсальность — возможность многовариантного использования как военными, так и гражданскими структурами страны нападения как против военных, так и против гражданских объектов страны поражения.

Сфера применения ИО включает как военную сферу, так и экономическую, банковскую, социальную и другие отрасли потенциального использования с целью:

- дезорганизации деятельности управленческих структур, транспортных потоков и средств коммуникации;
- блокировки деятельности отдельных предприятий и банков, а также стратегических отраслей промышленности путем нарушения многозвеньевых технологических связей и системы взаиморасчетов, проведения валютно-финансовых операций;
- инициирования больших техногенных катастроф на территории противника в результате нарушения процесса управления технологическими процессами и объектами, производство которых связано с большим количеством опасных веществ и высокой концентрацией энергии;

- массового распространения и внедрения в сознание людей определенных представлений, привычек и стереотипов поведения;
- провокации недовольства или паники среди населения, а также деструктивных действий разнообразных социальных групп.

При этом основными объектами применения ИО как в мирное, так и в военное время являются следующие:

- компьютерные системы и связь, используемые государственными организациями при выполнении своих управленческих функций;
- военная информационная инфраструктура, решающая задачи управления войсками и боевыми средствами, сбор и обработка информации в интересах вооруженных сил;
- информационные и управленческие структуры банков, транспортных и промышленных предприятий;
- средства массовой информации, и в первую очередь электронные (радио, телевидение и др.);
- телекоммуникационные узлы, центры спутниковой связи и каналы международного информационного обмена;

Чрезвычайно опасным ИО является сегодня для информационных компьютерных систем органов государственной власти, управления войсками и оружием, химическими и биологическими объектами, а также атомной энергетики, финансами и банками, экономикой страны, а также для людей при информационно-психологическом влиянии на них с целью изменения и управления их индивидуальным и коллективным поведением.

При этом по своей результативности информационное оружие сравнивается с оружием массового поражения. Большую опасность представляет использования информационного оружия террористами, этого допустить нельзя ни в коем случае, иначе процессы могут быть не контролированными.

К ИО, применение которого эффективно как в военное, так и в мирное время, относятся средства поражения информационных компьютерных систем и средства поражения людей (их психики).

Средства влияния (поражения) на людей и их психику различают в зависимости от цели их применения в психологической войне, к которым принадлежат следующие:

- искажение информации, получаемой политическим руководством, командованием и личным составом вооруженных сил противника, и навязывание им неправдивой или бессодержательной информации, лишающей возможности правильно оценивать события или текущую ситуацию и принимать взвешенные решения;
- психологическая обработка войск и населения;
- идеологические диверсии и дезинформация;
- поддержка благоприятного общественного мнения;
- организация массовых демонстраций под фальшивыми лозунгами;
- пропаганда и распространение ложных слухов;
- изменение и управление индивидуальным и коллективным поведением.

Наряду с использованием традиционных средств (печатные и электронные средства массовой информации) активно разрабатываются и испытываются специальные средства влияния на человека как через системы массовой информации (СМИ), так и через компьютерные сети: средства информационно-психологического (психофизического) влияния.

Применение этих и других видов информационного оружия в условиях открытости и роста международного информационного обмена определяет особенности защиты информационных систем от его влияния.

Психическая деградация общества в ближайшем будущем станет вполне реальной в случае, если государственные руководители не проанализируют мировые тенденции в экологии сознания и не

сделают соответствующих конструктивных выводов. Если государство не будет наблюдать пассивно за ростом деструктивных сил и мощностей разнообразных средств, применяемых ими, а примет необходимые меры по защите человечества от любой возможности насилия средствами ПСО, защиты интеллекта личности каждого гражданина (самого ценного генофонда нации и государства), сделав эту защиту открытой, доступной для широкого международного участия в работе и контроле, мы сможем предотвратить возможности психотронной войны.

Все чаще выражается мнение, что в третьем тысячелетии лидерство в мире будет определяться не столько экономическим потенциалом государства, сколько его способностью контролировать информационные процессы. Тофлер в своей книге «Война и антивойна» утверждает, что информационные технологии превращают общества второй волны (индустриальные) в общества третьей волны (информационные). В настоящее время мировая индустрия информационных и коммуникационных компьютерных технологий, по оценкам Мирового банка, составляет более чем 1 трлн. дол. США. Другими словами, происходит переход от экономической к информационной эре развития цивилизации.

Перестройка обществ на новых информационных базах обусловила потребность определения новых подходов к проблемам обеспечения национальной безопасности в информационную эпоху.

Информация, как совокупность знаний о фактических данных и зависимостях между ними, стала наиболее высоколиквидным товаром — стоимость информации и ее своевременной доставки в нужное место возрастает лавинообразно.

Информация, проникающая во все сферы деятельности государства, приобретает конкретное политическое, материальное и стоимостное выражение. Проблема безопасности информации, с точки зрения государственных интересов, в настоящее время приобрела особую актуальность и рассматривается как одна из приоритетных государственных задач, как важный элемент национальной безопасности.

Новые информационные технологии интегрируют мир в глобальных сетях инструментализма. Опосредствованная компьютерами коммуникация породила множество виртуальных сообществ.

Не секрет, что в настоящее время одним из основных средств обеспечения государством или любой организацией своих интересов на международной арене становится завоевание информационного пространства путем развития информационных технологий и создания на их основе информационных систем (ИС), определяющих доступ к достижениям в разнообразных отраслях науки, техники, экономики и т. п. При этом системы более высокого уровня, как правило, получают возможность управлять системами с относительно низким уровнем информатизации, направляя и постоянно контролируя эту деятельность согласно своим интересам.

Вопрос безопасности — важнейшая часть концепции внедрения новых информационных технологий во все сферы жизни общества, внутренняя система распознавания влияния и формирования реакции субъекта на действия внешней среды. Основное назначение иммунитета — защита субъекта от нарушения его целостности. Следовательно, информационный иммунитет — это система защиты человека от какой-нибудь информационной агрессии среды.

Наиболее конструктивной борьба с терроризмом будет тогда, когда на государственном уровне будет обеспечиваться не только информационная безопасность, но и экономическая, которая должна лишить ресурсного обеспечения террористические организации. Аналогичные уровни информационной и экономической безопасности должны быть обеспечены и на международном уровне. Это будет дополнительной защитой в том случае, если какая-то страна не сможет справиться с поставленными задачами для обеспечения безопасности международного терроризма.

Информационная безопасность — это такое состояние защищенности жизненно важных интересов личности, общества и государства, при котором сводится к минимуму причинение убытков из-за неполноты, несвоевременности и недостоверности информации, из-за отрицательного информационного влияния, отрицательных последствий функционирования информационных технологий, а также из-за несанкционированного распространения информации.

Рассматривая проблему национальной безопасности с системных позиций можно констатировать, что информационная безопасность занимает в ней особое место по следующим причинам:

- во-первых, информационные отношения и процессы пронизывают все отношения, имеющие место в обществе;

- во-вторых, в современных условиях, при широком использовании разнообразных информационных технологий, вопросы информационной безопасности получают самостоятельное значение;

- в-третьих, система внешних и внутренних угроз информационной безопасности имеет комплексный, всеобъемлющий для всех сфер деятельности человека, общества и государства характер.

Террор — особая форма политического насилия, характеризующаяся жестокостью, целеустремленностью и призрачной эффективностью. Терроризм был распространенным инструментом борьбы революции и контрреволюции в период глубоких социальных потрясений общества. В современных условиях наблюдается эскалация террористической деятельности экстремистских организаций, усложняется ее характер, возрастает обостренность и антигуманность террористических актов (захваты заложников, самолетов, взрывы, акты геноцида в религиозных конфликтах, прямые угрозы в ходе политической борьбы, похищение политических деятелей и их убийство, другие опасные действия по отношению к человечеству).

Терроризм — организация и осуществление взрывов или других действий, создающие опасность гибели людей, которые приводят к значительным имущественным убыткам или другим общественно опасным последствиям, если эти действия были осуществлены с целью нарушения общественной безопасности, запугивания населения или оказания влияния на принятие решений органами власти, а также угроза осуществления подобных действий в этих же целях.

Терроризм является преступлением против общественной безопасности.

Признаки терроризма:

1. Терроризм является одной из форм организованного насилия.
2. Терроризм влияет на более широкий круг общества, не ограничиваясь непосредственными жертвами насилия.
3. Формирование целей в большинстве случаев не связано с конкретными проявлениями насилия, то есть между жертвами и целью, на которую направляют свои действия террористы, нет прямой связи.
4. Тактическая цель терроризма заключается в том, чтобы обратить внимание на проблему, стратегическая — достичь определенных социальных изменений (свобода, независимость, избавление из исправительно-трудовых учреждений определенного контингента лиц, революция и т. п.).
5. Акты терроризма сами по себе составляют традиционные формы общекриминальных действий.
6. Терроризм парализует противодействие со стороны общественности.
7. Орудием влияния является психологический шок, который порождается осознанием того, что кто-нибудь может стать жертвой, независимо от того, к какой прослойке общества он принадлежит.
8. Любые правила или законы террористами не признаются — жертвами террористических актов могут быть как взрослые, так и женщины и дети.
9. Расчет делается на эффект внезапности, неожиданности.
10. Публичность является основным признаком терроризма.
11. Демонстративность актов терроризма — желание произвести впечатления на широкие массы.
12. Терроризм предполагает «политическое требование», поэтому не связан со стихийными восстаниями и выступлениями населения.
13. Терроризм требует немедленного удовлетворения выдвинутых требований, в противоположном случае прибегает к реализации угроз и эскалации насилия.
14. Может быть использован организациями любой политической окраски.
15. Практически всегда берет на себя ответственность за содеянные акты насилия, так как они являются средством достижения цели, а не самоцелью.

16. Представляет собой антитезу политического убийства. Ему присуща индифферентность по отношению к жертвам, в отличие от селективности при политическом убийстве.

17. Разрыв между непосредственной жертвой насилия и группой, которая представляет объект влияния, а также целью насилия.

Угрозами, которые могут быть причинены терроризмом в социальной сфере, являются: — низкий уровень жизни и социальной защищенности населения, наличие значительного количества граждан трудоспособного века, не занятых в общественно полезной деятельности; общественно-политическое противостояние отдельных социальных слоев населения и регионов странах.;

Кроме того, такая разновидность терроризма, как ядерный и биологический, является одним из опаснейших.

Фактически не существует причин, вследствие которых наша планета не смогла бы обеспечивать жизнь намного большего количества людей, чем ее нынешнее население. В действительности, тем не менее, распределение плодородных почв и благоприятных условий для их обработки не соответствует распределению населения. Эта проблема усугубляется возрастающей деградацией земельных ресурсов. Почти 2 млрд. гектаров земли испытывает деградацию вследствие деятельности человека, что ставит под угрозу, почти у 1 млрд. человек, наличие средств к существованию.

Основные причины этого — засоление почв в результате орошения, эрозия, вызванная чрезмерным выпасом и обезлесением, сокращение биологического многообразия.

Глобализация вместе с положительными факторами несет человечеству целый каскад угроз.

1. Прирост населения на один миллиард человек окажется серьезной проблемой, поскольку экономический застой и высокий процент безработного населения препятствуют выходу на рынок новых работников или эмигрантов.

Неадекватная урбанистическая инфраструктура и социальные услуги в большинстве городов создадут условия, которые будут способствовать нестабильности и беспорядку.

Миграция с Юга на Север станет основным источником напряженности, заставляя США и европейские страны отойти от развивающихся стран.

2. Рост населения будет способствовать уменьшению площади пахотных земель, сокращению пресноводных ресурсов и биологического разнообразия. Недостаток ресурсов, в особенности пресноводных, станет основной проблемой как для стран с развитой рыночной экономикой, так и для развивающихся стран. Эта проблема приведет к сокращению сельскохозяйственной продукции и к увеличению миграции сельского населения в города.

3. Внедрение и распространение технологических нововведений будут происходить медленно вследствие экономического застоя и политической неопределенности.

Дестабилизирующие эффекты внедрения новых технологий будут преобладать, что в свою очередь может привести к распространению оружия массового поражения; информационные технологии создадут дополнительные преимущества для террористов и преступников.

Преимущества от внедрения новых технологий получат только несколько богатых стран, а большинство стран окажутся позади.

4. Экономический спад в странах ЕС и США приведет к экономическому застою. Нарушится глобальное согласие по поддержке рыночных реформ, подрывая «Американскую экономическую модель», превращая США в более чувствительное государство и, тем самым, приводя к ослаблению роли США на международной арене. Сегодня как и на ближайшую перспективу будет происходить борьба за влияние на Азиатский регион, а не за Европу, эта тенденция станет главной составляющей международной политики. Ежегодные темпы роста экономики Китая начиная с 1978 года составляют 8–10%, Индии за последние 10 лет — 8%, в ближайшие 15–20 лет эти страны разделят второе и третье место в мире по экономическому потенциалу после США. За этот же период средне годовые темпы роста экономики США — 2,1%, а развитых стран ЕС — менее 1,2%, при критическом уровне экономической безопасности не менее 2,7% должно быть обеспечение годового прироста ВВП.

Экономический застой болезненно скажется на состоянии дел в странах с развивающейся рыночной экономикой, а также в большинства развивающихся государств.

5. У многих гетерогенных государствах обострятся религиозные/этнические расхождения. Возрастет социальная напряженность и насилие в Африке, в Центральной и Южной Азии и в отдельных районах Ближнего Востока. Возрастет политическое влияние Ислама. Увеличится вероятность террористических актов против объектов, связанных с глобализацией и США. Произойдет ослабление возможностей правительства на всех уровнях среди как развитых, так и развивающихся стран. Основными не решенными на ближайшее будущее вызовами для человечества останутся:

1. Неконтролируемый рост численности народонаселения мира, что ведет к обострению дефицита воды, продовольствия, энергоресурсов, ускоряет деградацию биосферы, в целом приведет к увеличению частоты решения разного рода конфликтов — межрегиональных, региональных, внутренних.

2. Стремление США к наращению своего военного доминирования и построения нового мирового порядка, а также политика индустриально-развитых стран мира, которая тормозит развитие стран третьего мира, вызывая рост враждебности развивающихся стран по отношению к США.

3. Увеличение числа эмигрантов и беженцев из опасных регионов мира в страны Запада.

4. Усиление роли международного терроризма при решении межрегиональных споров, в противодействие военному преимуществу США и их союзников, а также процесса глобализации.

5. Являясь средством запугивания населения и давления на политических лидеров, террористические акты, при всей их жестокости, не снимают основных причин конфликтов — перенаселение, борьба за естественные ресурсы, экологическое неравновесие.

6. Негативные факторы процессов глобализации послужили толчком для криминальной революции в мире.

6. В XXI веке для стран третьего мира, как считают эксперты, в особенности со слабой экономикой и отсутствием доступа к современным военным технологиям, из всех видов оружия наиболее выгодным станет биологическое. Оно будет применяться в основном не в военных условиях, а в мирное время, и не вооруженными силами, а террористами. Главными объектами будут не только армия противника, но и гражданское население.

Биологическое оружие может стать оружием мести, а также средством осуществления политических убийств в руках диктаторских режимов и террористических организаций. Еще более опасной угрозой по масштабам поражения, особенно для индустриально развитых стран является кибер-терроризм. Правительства стран в одиночку не могут справиться с сегодняшними глобальными вызовами. Поэтому нужно формировать мировое общественное мнение и объединять усилия сообщества на преодоление глобальных угроз. Решение вышеизложенных проблем по нашему мнению лежит в плоскости внедрения информациологической экономики как основы гармонического сосуществования мирового сообщества.

Благодарю за внимание.

Часть VI
Круглый стол «Комплексная безопасность в отраслях промышленности топливно-энергетического комплекса»

Некоторые аспекты обеспечения системной защиты объектов единой системы газоснабжения

Б. Н. Антипов

Уважаемый председатель!
Уважаемые участники нашего сегодняшнего круглого стола!

Разрешите предложить Вам доклад, касающийся некоторых аспектов обеспечения системной защиты объектов единой системы газоснабжения. На сегодняшний день открытое акционерное общество «Газпром» является одним из основных поставщиков денежных средств в бюджет страны. Функционирование газовой системы общества обеспечивает жизнедеятельность промышленности, мы обеспечиваем поставку газа и выполняем зарубежные контракты и несем ответственность за выполнение этих обязательств перед иностранными партнерами. На сегодняшний день объекты газотранстпортой системы и объекты добычи газа обеспечены системой защиты, которая в менее или более полной мере обеспечивает защиту этих объектов на сегодняшний день. Но рост угрозы терроризма, который наблюдается во всем мире, огромные последствия, которые могут возникнуть для страны в случае проявления террористических актов на объектах газо-транспортной системы, вынуждают совершенствовать имеющиеся системы защиты против несанкционированного доступа к объектам и приводят к необходимости проведения комплексной реорганизации как самой системы, так и ее материально-технической части, усовершенствование ее в свете новых достижений науки и техники. На сегодняшний день служба безопасности ОАО «Газпром» разработала и предлагает новую концепцию защиты объектов газотранспортной области страны, которая включает в себя не только создание новых технических средств предотвращения несанкционированного доступа к объектам, но и впервые проводит системной разбиение этих объектов. Система предусматривает деление объектов на:

- объекты обслуживаемые;
- объекты мало обслуживаемые;
- и объекты необслуживаемые.

Если рассмотреть возможные угрозы безопасности, которые возникают на этих объектах, то можно рассмотреть два типа этих угроз: внешнее воздействие и внутреннее воздействие (каждая из этих угроз представлена в укрепленном плане на указанном плакате). Необходимо отметить, что важность каждого из этих направлений будет оцениваться в этой комплексной программе и адекватные меры, которые будут разработаны, будут касаться именно сложности и последствий возможного негативного влияния. В созданной концепции, которая будет разработана в ближайшее время, будут преследоваться следующие цели:

- предотвращение угроз безопасности;
- обеспечение охраны жизни и здоровья сотрудников;
- недопущение повреждений или уничтожения сооружений, технических средств, имущества и ценностей.

Поставленные цели будут решаться задачами своевременного выявления и устранения угроз безопасности, обеспечения высокого уровня эксплуатации ИССО и САС и повышение эффективности всех видов обеспечения охраны, уточнение и совершенствование нормативно-правовой базы ОАО «Газпром» в области обеспечения безопасности. Хочется остановиться на этом пункте. На сегодняшний день практически отсутствует эта нормативно-правовая база, которая позволяет совершенствовать

систему и обеспечивать качественную, необходимую на новом научно-техническом уровне, систему защиты. Немаловажное значение имеет и совершенствование системы подбора и подготовки кадров. Остановлюсь на нескольких организационных принципах охраны объектов.

Если рассмотреть это со стороны требований системы комплексной безопасности объекта, то необходимо учесть следующие факторы действия системы безопасности:

1) непрерывность ее действия;

2) способность системы выполнять свои функции в течение жизненного цикла объекта;

3) централизованное управление функционирования системы безопасности;

4) использование стандартных методов;

5) способность системы охраны к развитию и совершенствованию (этот пункт можно расширить тем, что концепция по создаваемой системе защиты объектов предусматривает непрерывное совершенствование этих систем в соответствие с развитием научно-технического прогресса).

Нельзя не сказать про обоснованность материальных затрат по созданию этих систем. На этом вопросе можно остановиться несколько подробнее, поскольку практически утвержденных методов расчета, касающихся возможных затрат от несанкционированного доступа до разрушения объектов газотранспортной системы, на сегодняшний день нет. Однако предварительные расчеты, проведенные службой безопасности ОАО «Газпром», позволяют сказать, что эта величина, которую уже на сегодняшний день потеряла акционерное общество Газпром за счет несанкционированного воздействия на его объекты, очень существенна. Основные принципы, которые позволяют решать эти задачи, мы считаем, заключаются в соблюдении законодательства Российской Федерации. За последние несколько лет были приняты пять нормативно-правовых актов Российской Федерации, направленных на организацию обеспечения защиты и охраны объектов. Необходим также анализ прогнозирования угроз обеспечения комплексной защиты объектов. Все действия систем, действующих на сегодняшний день, а также систем, которые будут внедряться на новых объектах или модернизироваться на старых, необходимо увязать, а на сегодняшний день это достаточно слабо реализовано, с действиями силовых структур, находящихся в данном регионе.

Два следующих пункта не требуют отдельных пояснений. Остановлюсь на формировании баз данных о состоянии охраняемых объектов. В предлагаемой концепции одно из основных новшеств будет заключаться в том, что будут создаваться региональные центры (их будет 6 или 7), в которых будет фиксироваться не только состояние систем защиты объектов, но также будет возможность постоянного контроля ее функциональной надежности, а в случае необходимости и ручного воздействия на деятельность систем безопасности. Все данные с этих 6-ти региональных центров будут стекаться в единый центральный пункт системы безопасности Газпрома, где будет обеспечена полная информация руководству Газпрома о состоянии систем защиты, о несанкционированном доступе на охраняемые объекты, а также о принятых мерах по предотвращению и устранению последствий несанкционированного доступа.

Несколько слов об обеспечении охраны объектов. Опять же начнем с тех средств, которые на наш взгляд могут это обеспечить.

Непрерывный контроль и управление доступом на объекты: охранное телевидение; охранная сигнализация периметра зданий и сооружений; контроль мобильных объектов (имеется в виду передвижение первых лиц Газпрома); досмотр и обнаружение запрещенных к проносу веществ, предметов, материалов, стационарных и портативных исполнений (здесь имеется в виду, что мы предлагаем рассматривать поэтапную защиту объектов, которая будет заключаться не только в защите от проникновения автотранспорта, но и физических лиц, которые будут на проходных пунктах проверяться специальными приборами на предмет наличия взрывчатых веществ); препятствия для несанкционированного доступа на территорию объекта (предлагается рассматривать как один из вариантов разрабатываемой концепции, возможность механической преграды несанкционированному проникновению транспорта на объект не только внутрь, но также возможность его блокировки на месте нарушения). Здесь надо сказать, что международная практика, используемая террористами в мире на сегодняшний день, отмечает два основных способа террористического акта: использование террориста-смертника, который несет с собой определенное количество взрывчатых веществ, или использование автотранспорта,

загруженного взрывчатыми веществами. Я здесь не рассматриваю возможное воздействие на объект путем ракетного удара или нанесения удара с летательного аппарата. Методы, которые должны обеспечить защиту объекта, мы предлагаем рассматривать:

- использование вооруженной охраны;
- организацию централизованной охраны;
- использование интегрированных систем безопасности, что подразумевает, что эти системы находятся во взаимосвязи друг с другом и работают синхронно;
- контроль состояния защищенность объектов Газпром на основе единой информационно-аналитической базы данных, о чем я сказал несколько выше;
- применение поражающих средств временного воздействия на периодически обслуживаемых объектах (это заключается в том, что есть объекты, на которых отсутствует постоянный контингент людей, поэтому здесь предусматривается многоуровневая система защиты, которая, допустим, на первом уровне будет сигнализировать о несанкционированном доступе, на втором, что произошел несанкционированный доступ в закрытое помещение, где и будут применяться определенные вещества, приводящие к временному воздействию на террориста, причем средства, которые будут использовать для этого временного воздействия, должны пройти медицинскую экспертизу, иметь сертификат соответствия и срок годности);
- апробация и прокладка новых систем на специализированном полигоне (это тоже один из пунктов, который заложен в концепцию, и он заключается в том, что все системы, которые мы хотим использовать, должны пройти аттестацию и сертификацию именно в составе объекта, для этого будет усовершенствован полигон, он будет заново оснащен, где и будет проводиться контроль за всеми системами и их аттестация).

Если остановиться на организационном обеспечении, то, как я сказал немного ранее, одной из основных идей является создание региональных центров. Огромное количество объектов усложняет систему централизованного контроля и наблюдения за их состоянием. Предлагаемая система региональных центров, которые потом будут связаны с центральным центром службы безопасности, позволит решить эту проблему, уменьшить риск ложных сигналов, ложных срабатываний.

- регламентация деятельности служб безопасности объектов, привлечение частных предприятий, аккредитованных в сфере охранной деятельности (здесь тоже на сегодняшний день не должно возникнуть больших проблем, наши частные охранные предприятия в большинстве имеют лицензии, но в своем лице должны будут получать особый допуск к основам безопасности Газпрома);
- применение современных методов контроля при оснащении объектов ИССО и САС.

С точки зрения работы сотрудников, физических лиц, то здесь мы предлагаем обязательное страхование сотрудников службы безопасности объектов. Как я говорил, недостатки существующей нормативно-правовой базы должны быть тоже устранены. Пункт, касающийся добровольной сертификации оборудования, тоже на сегодняшний день недостаточно нормативно обоснован, но мы считаем, что только оборудование, которые пройдет сертификацию необходимую для ее работоспособности на наших объектах, будет на них использована.

Кадровая проблема — это создание системы аттестации повышения квалификации специалистов в области эксплуатации наших объектов. Здесь надо сказать, что по имеющейся на сегодняшний день информации, кадры не совсем соответствуют не только современной технике, но мы должны непрерывно обучать наших сотрудников, потому что техника будут постоянно усложняться, и сигналы, поступающие с техники, должны восприниматься людьми, прошедшими курс обучения.

На сегодняшний день необходимо отметь отсутствие нормативно узаконенного взаимодействий служб охраны объектов ОАО «Газпром» с территориальными органами МВД, ФСБ и МЧС России. Здесь мы считаем необходимым развивать и углублять это сотрудничество, естественно, на основе двусторонних договоров. Если рассматривать конкретно области взаимодействия, как самих территориальных органов, так и службы безопасности акционерного общества, то эти все данные приведены в следующей таблице. Главное это обеспечение связи между службой безопасности объектов ОАО

«Газпром» и территориальными органами ФСБ, МВД и МЧС, для того чтобы помочь не только в устранении возможного террористического акта или несанкционированного доступа. Главная задача этого взаимодействия — это возможность предотвращения этого доступа. На сегодняшний день достаточно широко используется информация у наших силовых структур о наличие в районе подозрительных лиц и плюсом технические возможности позволяют определять визуальное, тактическое наблюдение за объектом на достаточно большом удалении. Именно взаимодействие с местными силовыми структурами и службой безопасности объектов позволит не только, на наш быстро, устранять возможные нарушения работоспособности объектов, но также и предотвращать возможный несанкционированный доступ.

Таким образом, необходимо отметить, что предлагаемая комплексно-целевая программа на 2005–2007 гг. позволит начать переоснащение систем защиты объектов, а также разработать наиболее перспективные на сегодняшний день средства и методы этой защиты. На первом этапе программа предусматривает из 30 тысяч объектов Газпрома выделение наиболее критических на сегодняшних день в понимании важности обеспечения работы всей газотранспортной системы, здесь будут выделены наиболее важные объекты, на которых в первую очередь будут проведены работы по созданию и модернизации системы защиты. Главные цели, которые хотелось бы озвучить, этой комплексной программы являются:

- снижение уровня уязвимости объектов;
- совершенствование системы защиты;
- централизация управления системами обеспечения безопасности;
- и применение унифицированных интегрированных систем охраны.

Эти цели могут быть достигнуты, хотя как я уже сказал, для всех объектов это создать не удастся, но разработать унифицированные методы, технические средства защиты объектов, мы должны это за 2 года сделать. А также разработать унифицированную документацию по пиру объектов. Задачи можно перечислить следующие:

- раннее предупреждение об угрозах;
- снижение их уровня и нейтрализация;
- создание новых систем и технологий;
- разработка новых нормативно-методических документов;
- и оснащение объектов современными интегрированными комплексными средствами.

В заключении, еще раз вернувшись к ожидаемым результатам реализации программы, необходимо сказать, что все-таки работа предстоит большая, она является важной, служба безопасности Газпрома уделяет ей огромное внимание, программа утверждена правлением. На сегодняшний день задачи, которые ставятся и которые должны быть достигнуты в результате реализации программы, самое главное, это обеспечение комплексной безопасности объектов Газпром. И здесь необходимо, опять же я прочитаю то, что является основой этого обеспечения, это на основе контроля состояния защищенности критически важных объектов, на основе защищенности периодически обслуживаемых объектов, на основе определения и состояния мобильных объектов, на основе обнаружения взрывчатых вещества и пресечения применения на рубежах охраны объектов, на основе исключения несанкционированного движения транспорта в зонах объекта и на основе контроля состояния и охраны прибрежных зон морских участков центральных газопроводов, если сами объекты линейной части в морские участки могут быть как-то защищены, то нахождение участков линейной части на глубине 50 метров, что у нас имеет место быть, они доступны для несанкционированного воздействия людьми с аквалангами.

В заключение хочется отметить, что реализации программы, строится на комплексном подходе к охране и защите ОАО «Газпром», создании базы данных состояния и функционирования системы безопасности, создании региональных центров. Любое воздействие на объекты Газпрома ко всему прочему приводит к экологическим последствиям для окружающей среды, что также является одним из тех факторов, почему создание надежной, эффективной системы защиты объектов Газпрома является на сегодняшний день актуальной задачей.

Инфраструктура систем комплексного обеспечения безопасности предприятий: проблемные вопросы

В. Ф. Пустарнаков, В. Н. Кустов

Уважаемые коллеги!

Разрешите выразить благодарность организаторам за возможность выступить на таком знаменательном форуме от имени руководства фирмы «ГазИнформСервис» и выразить надежду на достаточно быструю встречу в городе Санкт-Петербурге на форме по PKI безопасности и на выставке по проблемам информации, которая пройдет с 8 по 10 ноября в ЛенЭкспо. «ГазИнформСервис» участвует в организации данных мероприятий и организует заседание 5-й секции Газпрома.

Разрешите начать свой доклад с определения предмета защиты, который по нашему пониманию состоит из следующих компонентов: люди (персонал объекта, клиенты предприятий), материальные и финансовые ценности (оборудование, производственные объекты, административные здания, объекты инфраструктуры), информация конфиденциального характера. И иногда забывают, что предметом защиты кроме этих материальных и нематериальных объектов еще является такой объект как отношения, которые возникают в процессе производственной деятельности между персоналом, обслуживающим производственные объекты, и материальными ценностями, такими как финансовые отношения, производственные отношения и различные отношения организационного характера. Поэтому, наверное, эта схема неполна и эти отношения можно было бы добавить в предмет защиты. Если касаться задач системы комплексной безопасности, то, на наш взгляд, этими задачами являются предметы защиты, на которых говорилось на предыдущем слайде. Кроме этого: обеспечение превентивных и контр мер против физических угроз; предотвращение несанкционированного проникновения на охраняемые объекты, здания, зоны, помещения или к охраняемым предметам; организация тревожного оперативного оповещения; организация контролируемого доступа сотрудников, посетителей и транспорта на территорию объекта и в режимное помещение; обеспечение предусмотренного внутренним распорядком режима работы сотрудников; наблюдение за прилегающими к территории объекта зданиями и за транспортными потоками на ней; создание видео-архива; компьютерный анализ безопасности объекта. Принципы построения комплексной системы безопасности, на наш взгляд, выглядят следующим образом: универсальность; комплексность; разумная достаточность; оперативность; адаптивность; непрерывность; систематичность; целеустремленность; многорубежность; равнопрочность; шелонированность[1]; совместимость; простота; экологическая чистота; незаметность; дружественность; неуязвимость; документированность; правомерность. Вот, на наш взгляд, основные принципы построения системы комплексной безопасности.

Для чего нужна интеграция? Во-первых, интегрированная техническая система охраны предлагает объединение на базе современных информационных технологий, программно-аппаратной интеграции нескольких подсистем функционально и информационно связанных друг с другом. Даже при минимальном уровне интеграции взаимодействие подсистем осуществляется таким образом, что события в одной из подсистем могут воздействовать на другие и вызывать определенную реакцию. По сравнению с простой совокупностью отдельных средств защиты применение интегрированных систем обеспечивает на наш взгляд следующие преимущества: более быструю и точную реакцию на происходящие события; оптимальный анализ текущих ситуаций; значительное снижение риска, связанного с человеческим фактором; уменьшение затрат на оборудование; облегчение работы обслуживающего персонала за счет автоматизации процессов; снижение затрат на монтаж и эксплуатацию системы безопасности; сокращение персонала, затрат на его обучение и содержание. Основными компонентами

[1] Обращаю внимание, что в нашем понимании многорубежность и шелонированность понятия не равнозначные. Многорубежность просто означает некое количество рубежей. Вполне возможно, например, применить два рубежа нотификации в информационной системе, но при этом они выполняют равнозначные функции, просто нотификация становится многофакторной. Шелонированность подразумевает, когда каждый рубеж последовательно выполняет различные задачи и следующий рубеж рассчитывается из предпосылки, что предыдущий рубеж был злоумышленником пройден.

инфраструктуры системы комплексной безопасности, на наш взгляд, являются следующие: система инженерной защиты (эта одна из основных систем); система охранной сигнализации (стандартный компонент); система пожарной безопасности; система контроля и управления доступом; система телевидения охранного и промышленного при необходимости; система централизованного сбора и обработки информации; система оперативной связи; система контроля в реальном времени местоположения и состояния подвижных объектов, используемых не только для перевозки особо важных персон, но и для особо важных ценных грузов с целью оперативного принятия адекватных мер при появлении террористических угроз и противоправных действий; тревожная взрывная сигнализация с использованием поражающих средств временного воздействия, обеспечивающих нейтрализацию террористов и нарушителей, на время достаточное для прибытия оперативных сил; система визуально-звукового оповещения в случае экстренной эвакуации людей. Кроме того, необходимым компонентом, как показывают последние события в городе Москве, является система гарантированного электропитания.

В связи с упоминающимися документами, изданными в ОАО «Газпром», а это, прежде всего план и комплексно-целевая программа, на наш взгляд, из этих документов следует пять направлений работы. Прежде всего, это создание инфраструктуры и нормативно-методической базы безопасности объектов единой системы газоснабжения; оснащение объектов унифицированными современными комплексами ИТСО (инженерно-технической системы охраны) и САЗ (система антитеррористической защиты); подготовка квалифицированных специалистов по эксплуатации средств защиты; создание системы «гарантия-качество» обеспечения безопасности объектов системы газовой отрасли; создание системы взаимодействия с субъектами, осуществляющими борьбу с терроризмом.

По реализации первого направления мы видим реализацию следующих задач:

1) формирование инфраструктуры обеспечения безопасности объектов единой системы газоснабжения в сфере ИТСО;

2) создание автоматизированной системы контроля защищенности объектов;

3) разработка методически-программного обеспечения оценивания состояния защищенности;

4) создание компьютеризированной системы обмена данными;

5) разработка стандартов организации ОАО «Газпром» по категорированию объектов, по степени потенциальной опасности и по террористической уязвимости этих объектов;

6) определение критически важных объектов подлежащих первоочередному оснащению ИТСО и САЗ.

Задачами второго направления, на наш взгляд, являются:

1) оснащение объектов защиты унифицированными современными комплексами ИТСО и САЗ;

2) разработка типовых проектных решений;

3) разработка унифицированных комплексов и их интеграция в существующую систему;

Модернизация комплексов безопасности критически важных объектов выполняется на основе решения этих трех задач.

Системные аспекты обеспечения безопасности объектов ОАО «Газпром» с использованием показателей риска

В. Н. Пожарский, В. С. Сафонов, В. В. Лесных

Единая система газоснабжения России (ЕСГ) — уникальная организационно-техническая система, которая характеризуется большим числом производственных объектов и их протяженностью. Принципиальной особенностью ЕСГ России является неразрывность технологического процесса, протекающего в реальном времени и требующим полной увязки и совместимости по материальным балансам и технологиям во всех звеньях системы (добыча, транспорт, переработка, хранение). Обеспечение безопасного и устойчивого функционирования такой системы является сложной научно-технической и организационной проблемой. Острота этой проблемы существенно возрастает в последние годы в связи с ростом внешних и внутренних угроз. К числу наиболее значимых относятся угрозы противоправных действий, включая террористические акты.

Принятие решений о структуре системы обеспечения безопасности и оптимальном распределении материальных и финансовых ресурсов необходимо производить на основе результатов ранжирования производственных и административных объектов ОАО «Газпром» как объектов противоправных воздействий по величине системных рисков.

Аварии на объектах ОАО «Газпром», независимо от природы инициирующих событий, могут приводить к социальным и экономическим последствия как в отдельных регионах, так и по стране в целом. Оценка последствий аварий, включая величину ожидаемого ущерба, производится с использованием оптимизационных и имитационных моделей, описывающих распределение потоков газа в нормальных и аварийных режимах.

При распределении ресурсов, выделяемых на организацию защиты объектов ОАО «Газпром» количественная оценка системных рисков и анализ эффективности предупредительных мероприятий должны стать неотъемлемыми этапами ранжирования объектов ОАО «Газпром» как объектов охраны. Затраты на целевые предупредительные мероприятия должны рассматриваться как инвестиции в безопасность и устойчивое развитие. Соответственно, затраты на обеспечение безопасности и устойчивости ЕСГ России должны обосновываться и нормироваться в зависимости от величины ожидаемых ущербов.

Актуальные проблемы обеспечения информационной безопасности газовой сферы

А. И. Ефимов

К актуальным проблемам обеспечения информационной безопасности газовой сферы относятся:

- Защита базовых и технических средств АСУ ТП.
- Защита информационных ресурсов АСУ ТП в том числе базы данных и системы скада.
- Система защиты технологического оборудования АСУ ТП.
- Контроль используемых в АСУ ТП средств телекоммуникаций, связи, вычислительной техники, скада систем, отсутствие недекларированных возможностей.
- Обеспечение разработки АСУ ТП в защищенном исполнении и ее аттестация по требованиям безопасности информации.
- Проведения контроля установленных правил эксплуатации и настроек систем безопасности силами российских организаций, имеющих соответствующее право деятельности (в данном случае имеется в виду, что, как правило, соответствующие объекты АСУ ТП обслуживаются российскими предприятиями, но встречаются такие случаи, когда еще со времен Советского Союза те или иные объекты создавались и продолжают технологически и технически поддерживаться уже странами ближнего зарубежья, учитывая то, что вопросы информационной безопасности являются лицензированным видом деятельности, а технологические объекты находятся на территории Российской Федерации, отсюда и наше видение, что такие работы должны выполнятся российскими организациями, которые имеют соответствующие лицензии).

Особое внимание должно уделяться обеспечению физического и логического разграничения АСУ ТП с другими информационно-системными предприятиями. Например, системами управления производственно-хозяйственной деятельностью, иначе говоря, АСУ ТП — это один отдельный логический сегмент или физический сегмент сети, а бухгалтерия, учет материально-технических ресурсов, складские поставки и так далее — это другой сегмент. Между ними должно быть четкой разделение. Сюда необходимо отнести информационные системы других отраслей. Например, автоматизированные системы коммерческого учета электроэнергии. Сейчас существует требование, что производственные предприятия должны быть обеспечены АСКУЭ. В данном случае надо понимать, что АСКУЭ — это система, принадлежащая одному ведомству, например, РАО ЕС, а АСУ ТП может принадлежать предприятию по нефтедобычи, газодобычи. Между ними должно быть произведено логическое разграничение.

Принимая во внимание, что терроризм, в том числе и технологический, направлен, прежде всего, на разрушение технической инфраструктуры, в целях нанесения максимального ущерба производству и населению, то следует отметить, что выше названные подходы к обеспечению безопасности информационных систем критических производств должны быть закреплены на федеральном уровне в техническом регламенте по безопасности информации, о чем уже говорилось в начале выступления.

Вторая проблема обеспечения безопасности критических производств, на наш взгляд, это необходимость обеспечения информационной безопасности технологических объектов в условиях преобразования отрасли либо предприятия. Данная проблема стала особо явной после аварии на энергосетях Москвы и Московской области, а также близ лежащих регионов, которая произошла в мае 2005-го года. Очевидно, что при любой реструктуризации крупного предприятия, управляющего деятельностью территориально распределенных технологических объектов, вопрос об адаптации АСУ ТП и ее систем обеспечения информационной безопасности в новой организационной структуре должен выдвигаться в разряд первоочередных. Должны быть предусмотрены доработка технического облика АСУ

ТП и ее системы защиты, корректировка рабоче-конструкторской и эксплуатационной документации, переподготовка персонала, отработка взаимодействия элементов новой технологической структуры между собой, а также со смежными технологическими, в частности, диспетчерскими службами. В обязательном порядке в ходе реструктуризации должны быть решены другие организационные вопросы, к числу которых можно отнести: закрепление ответственность в сфере защиты информации за руководителями вновь образованных структур; аттестация модернизированной АСУ ТП по требованиям безопасности информации; разработка и ввод в действие положений взаимодействия со смежными органами технологического управления и другие положения.

Третья проблема обеспечения безопасности технологических производств, по нашему мнению, заключается в необходимости обеспечения минимально достаточного уровня информационной безопасности технологических объектов, построенных в 70–80-х гг. прошлого века. Мы все хорошо знаем, что подготовка газа к транспорту или какой-либо другой технологический объект — это очень дорогое удовольствие. Вся страна направляла свои усилия на реализацию таких объектов. Объекты такие служат очень долго и на данный момент такие объекты встречаются у нас в стране, отсюда и проблемы. Оперативное принятие мер по повышению мер по информационной безопасности в отношении таких технологических объектов имеет определенные сложности, которые связаны с тем, что средства информатизации объектов технологической инфраструктуры прошлого века относятся к предыдущему поколению средств защитной техники и трудно адаптированы к современному аппарату средств защиты информации. Бытует мнение, что такие объекты имеют допустимый уровень безопасности, что обусловлено низким уровнем их автоматизации, а также использованием антикварного аппаратного обеспечения. На самом деле по нашей информации, исследования возможностей по деструктивному воздействию на подобные инфраструктуры не проводились. В связи с этим угрозы по отношению к ним постоянно сохраняются, что делает актуальным задачи целевой ревизии уязвимости АСУ ТП старых технологических объектов.

Последняя проблема обусловлена необходимостью разработки конкурентоспособных отечественных элементов АСУ технологическими процессами и обеспечение независимости их потребителей от иностранных производителей. Это сразу решит целый ряд взаимосвязанных проблем информационной безопасности и, конечно же, мы должны свое хозяйство, связанное с автоматизацией технологических объектов, на которых присутствуют непрерывные циклы производства, разработать свои собственные аппараты программной базы.

Подводя итоги сказанному, хочется отметить, что проблемы имеют комплексный характер, а их решение требует не только усилий со стороны корпораций, но и государственных органов.