САНКТ-ПЕТЕРБУРГСКИЙ
ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО

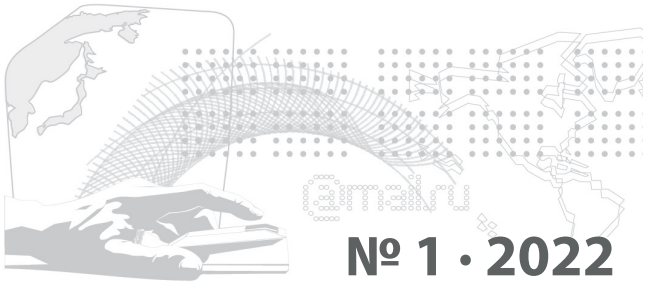
ПРОБЛЕМЫ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

КОМПЬЮТЕРНЫЕ СИСТЕМЫ

№ 1 • 2022

РАЗДЕЛЫ ВЫПУСКА:

Санкт-Петербург
2022



№ 1 · 2022

Журнал является органом Совета Регионального Северо-Западного учебно-научного центра информационной безопасности

Журнал включен в перечень изданий, утвержденных ВАК, для публикации основных результатов диссертационных исследований

АДРЕС РЕДКОЛЛЕГИИ:

195251, Санкт-Петербург,
ул. Политехническая, 29.
ФГАОУ ВО «Санкт-Петербургский политехнический университет Петра Великого».

Тел. (812) 552-76-32

e-mail: kafedra@ibks.ftk.spbstu.ru

http://jisp.ru/kontakty

Свидетельство о регистрации **№ 018607** от 17.03.99 г. выдано Государственным комитетом Российской Федерации по печати

С 1 января 2019 г. подписка на журнал «Проблемы информационной безопасности. Компьютерные системы» осуществляется через объединенный каталог «Пресса России»

https://www.pressa-rf.ru

Подписной индекс — Т18237

Выпускающий редактор **Д. А. КЛУБНИЧКИНА**

Ответственный секретарь **Н. Ю. ЛОВЧИНОВСКАЯ**

12 февраля 2022 года в Санкт-Петербурге на 82 году жизни скончался главный редактор и основатель журнала «Проблемы информационной безопасности. Компьютерные системы», доктор технических наук, профессор, Заслуженный деятель науки России, основатель и научный руководитель Института кибербезопасности и защиты информации Санкт-Петербургского политехнического университета Петра Великого Пётр Дмитриевич Зегжда.

Петр Дмитриевич посвятил свою жизнь развитию отечественной науки, занимаясь важнейшими фундаментальными и прикладными исследованиями в области информационной безопасности.

Он был одним из первых российских специалистов в этой сфере. Труды и идеи Петра Дмитриевича стали основой для десятков уникальных разработок, в числе которых средства анализа управления безопасностью, защищенная операционная система, криптографические системы защиты информации, антивирусные программы, и многие другие.

Зегжда П. Д. был не только выдающимся учёным, но и талантливым педагогом и организатором, под его началом была создана одна из первых в России кафедр в сфере информационной безопасности — кафедра «Информационная безопасность компьютерных систем» Санкт-Петербургского политехнического университета, которая стала основой для первого в России Института кибербезопасности и защиты информации.

Петр Дмитриевич активно выступал за популяризацию науки по его инициативе в 1999 году появился журнал «Проблемы информационной безопасности. Компьютерные системы». Более двух десятилетий Пётр Дмитриевич руководил редакционной деятельностью журнала, лично занимался отбором материалов, созданием рубрик и непосредственным общением с авторами, учёными и коллегами со всего

мира. Как главный редактор он всегда стремился к разнообразию тем и научных идей, желая сделать журнал современным и соответствующим высоким общемировым научным стандартам.

Уход Петра Дмитриевича Зегжды из жизни — невосполнимая потеря для сферы информационной безопасности России, для всех, кто знал его, имел честь быть его учеником, соратником, коллегой. Не стало выдающегося ученого, талантливого педагога и организатора, мудрого наставника, доброго и отзывчивого человека. Память о нем навсегда останется в наших сердцах.

Редакция журнала
«Проблемы информационной безопасности.
Компьютерные системы»

МиТСОБИ

Конференция **«Методы и технические средства обеспечения безопасности информации» (МиТСОБИ)** — это встреча профессионалов информационной безопасности, единственная и старейшая конференция, с 1991 года ежегодно проходящая в Санкт-Петербурге.

МиТСОБИ — это возможность узнать самые современные направления и поделиться опытом, это интересные доклады и горячие дискуссии, в которых молодые разработчики имеют возможность узнать мнение мэтров информационной безопасности, а руководители — выяснить, как на практике решать самые острые вопросы, оценить важность и действенность этих решений для обеспечения информационной безопасности как страны в целом, так и для каждого участника киберпространства. Особенность конференции — это диалог на пересечении теории и практики, науки и бизнеса.

Ежегодное количество участников — до 300 человек, среди которых руководство и специалисты органов государственной власти РФ, вузов, академических учреждений, разработчики и молодые ученые, представители научно-исследовательских организаций и коммерческих предприятий из различных регионов России.

Организатор конференции

НЕОБИТ

Соучредители

Комитет по информатизации и связи Правительства Санкт-Петербурга

Комитет по науке и высшей школе Правительства Санкт-Петербурга

ПОЛИТЕХ Санкт-Петербургский политехнический университет Петра Великого

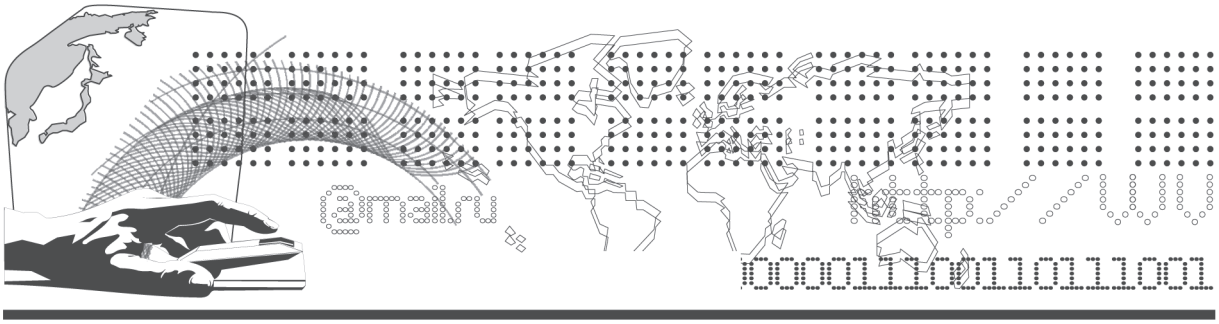
Санкт-Петербургский политехнический университет Петра Великого

При участии

Федеральной службы безопасности РФ, Федеральной службы по техническому и экспортному контролю, Управления специальной связи и информации ФСО России в СЗФО, Федеральной службы по финансовому мониторингу. Подробная информация — на сайте конференции www.mitsobi.ru.

Контактные лица:
Селиванова Анна Юрьевна — 8 (800) 222-28-06 (звонок бесплатный);
+7 (812) 535-28-06.

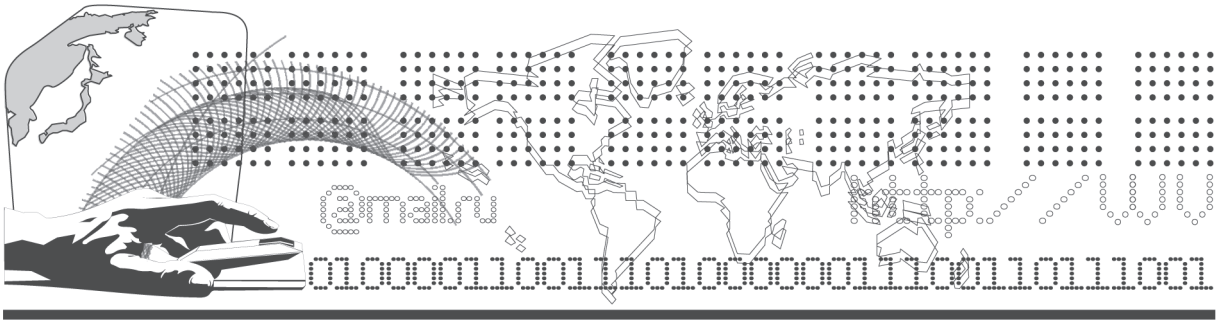
E-mail: mitsobi@neobit.ru.



СОДЕРЖАНИЕ

БЕЗОПАСНОСТЬ КИБЕРФИЗИЧЕСКИХ СИСТЕМ

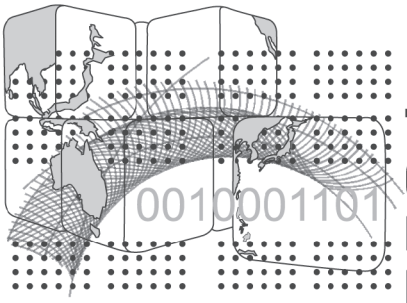
МОДЕЛИРОВАНИЕ ТЕХНОЛОГИЧЕСКИХ СИСТЕМ, АЛГОРИТМИЗАЦИЯ ЗАДАЧ И ОБЪЕКТОВ УПРАВЛЕНИЯ



CONTENTS

CYBER-PHYSIC SYSTEMS SECURITY

TECHNOLOGICAL SYSTEMS, ALGORITHMIZATION OF TASKS AND CONTROL OBJECTS MODELING



ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Научная статья
DOI 10.48612/jisp/z4at-58k4-tbp6
УДК 004.056.57

В. Г. Анисимов[1], Е. Г. Анисимов[2], Т. Н. Сауренко[2], В. П. Лось[3]

[1]Санкт-Петербургский политехнический университет Петра Великого (СПбПУ)
Россия, 195251, Санкт-Петербург, ул. Политехническая, д. 29
[2]Российский университет дружбы народов, Россия, 117198, г. Москва, ул. Миклухо-Маклая, д. 6
[3]МИРЭА — Российский технологический университет
Россия, 119454, ЦФО, г. Москва, Проспект Вернадского, д. 78

ОЦЕНКА ЭФФЕКТИВНОСТИ СИСТЕМ ЗАЩИТЫ КОМПЬЮТЕРНЫХ СЕТЕЙ ОТ ВИРУСНЫХ АТАК

Аннотация. Компьютерные сети — важная часть современной цивилизации. Они применяются буквально во всех сферах человеческой деятельности. Существенные потери вследствие отказов этих сетей обусловливают высокие требования к устойчивости их функционирования. Обеспечение необходимой устойчивости, в частности, опирается на защиту компьютерных сетей от вирусных атак. В ее интересах создаются соответствующие системы защиты. В качестве показателя эффективности таких систем предлагается использовать количество компьютеров сети, которые успевает заразить вирус до его выявления и удаления. Эмпирическую основу оценки эффективности систем защиты компьютерных сетей от вирусных атак при этом составляют данные получаемые в результате натурных испытаний и (или) предшествующий опыт эксплуатации. Эти данные имеют случайный характер, а их объем, как правило, существенно ограничен. Подход к оценке эффективности систем защиты компьютерных сетей от вирусных атак с учетом указанных особенностей эмпирических данных рассматривается в данной статье. В основу подхода положено представление эмпирических данных виде малой выборки из генеральной совокупности значений случайной величины количества компьютеров сети, которые успел заразить вирус до его выявления и удаления. В качестве модели испытаний принята функция распределения этой величины. Построение функции распределения опирается на принцип максимума неопределенности. В качестве меры неопределенности принята энтропия Шеннона.

Ключевые слова: компьютерная сеть, вирусная атака, система защиты сети, эффективность защиты.

Research article
DOI 10.48612/jisp/z4at-58k4-tbp6

V. G. Anisimov[1], E. G. Anisimov[2], T. N. Saurenko[2], V. P. Los[3]

[1]Peter the Great St. Petersburg Polytechnic University, Russia, St. Petersburg
[2]Peoples' Friendship University of Russia, Russia, Moscow
[3]MIREA — Russian Technological University, Russia, Moscow

ASSESSMENT OF THE EFFICIENCY OF PROTECTION SYSTEMS OF COMPUTER NETWORKS FROM VIRAL ATTACKS

Abstract. Computer networks are an important part of modern civilization. They are used literally in all spheres of human activity. Significant losses due to failures of these networks cause high requirements for the stability of their operation. Ensuring the necessary resilience, in particular, relies on

the protection of computer networks from virus attacks. In its interests, appropriate protection systems are being created. As an indicator of the effectiveness of such systems, it is proposed to use the number of network computers that a virus manages to infect before it is detected and removed. The empirical basis for assessing the effectiveness of systems for protecting computer networks from virus attacks is the data obtained as a result of field tests and (or) previous operating experience. These data are random in nature, and their volume, as a rule, is significantly limited. An approach to assessing the effectiveness of systems for protecting computer networks from virus attacks, taking into account the indicated features of empirical data, is considered in this article. The approach is based on the presentation of empirical data in the form of a small sample from the general population of values of a random variable of the number of computers on the network that the virus managed to infect before it was detected and removed. The distribution function of this quantity is taken as a test model. The construction of the distribution function is based on the principle of maximum uncertainty. Shannon's entropy is taken as a measure of uncertainty.

Keywords: computer network, virus attack, network protection system, protection efficiency.

Характерной особенностью современного развития всех сфер деятельности является цифровизация. Это обусловлено необходимостью высоких скоростей обработки информации, удобных форм ее хранения и передачи.

В связи с цифровизацией широкое распространение получили компьютерные сети (КС). Каждая такая сеть представляет собой объединение компьютеров и локальных сетей структурных подразделений организаций различных сфер деятельности на основе внутренних коммуникационных каналов и каналов сети Интернет. Она составляет техническую основу управления функционированием организации и, с одной стороны, способствует повышению его эффективности, но с другой — обостряет проблему информационной безопасности [1–7].

Одним из важнейших направлений обеспечения информационной безопасности КС является противодействие вредоносным воздействиям, реализуемым в форме компьютерных вирусов [8–13]. Компьютерные вирусы являются вредоносными программами, которые распространяются в компьютерной сети, внедряются в другие программы, модифицируя их и, в конечном счете, приводят к неработоспособности как отдельных компьютеров, так и сети в целом. Для предотвращения такого деструктивного воздействия создаются соответствующие системы защиты КС от вирусных атак. Обоснование решений в процессе создания, модернизации и управления функционированием этих систем опирается на соответствующую оценку их эффективности [14–22]. Поскольку основная задача систем защиты КС от вирусных атак состоит в выявлении, недопущении распространения и удалении проникнувших в сеть вирусов, в качестве показателя эффективности их функционирования может быть принято количество компьютеров сети, которые успевает заразить вирус до его выявления и удаления. Наиболее объективным источником информации, обеспечивающим оценку эффективности таких систем являются натурные испытания на стадии разработки, и предшествующий опыт на стадии эксплуатации. При этом недетерминированный характер реальных ситуаций функционирования систем защиты КС от вирусных атак обусловливает целесообразность формального представления показателя эффективности в виде функции распределения случайной величины X, характеризующей количество компьютеров сети, которые успел заразить вирус до его выявления и удаления. Важной особенностью условий усложняющих оценку эффективности рассматриваемых систем является существенная ограниченность объемов выборок эмпирических данных о значениях величины X.

Формирование методического подхода к решению задачи оценки эффективности системы защиты КС с учетом случайно-

сти результатов отражения вирусных атак и ограниченности объема эмпирической выборки составляет цель настоящей статьи.

Формализованное представление подхода

Из физического смысла случайной величины X следует, что она является дискретной и может принимать значения {0, 1, 2, ...}. Поэтому распределение величины X задается указанием вероятности того, что она примет значение x из множества {0, 1, 2, ...}, то есть

$$P(x) = P(X = x),\ x = 0,\ 1,\ 2,\ ....\qquad (1)$$

Функция (1) в указанном контексте содержит всю информацию о случайной величине X и, следовательно, является моделью оценки эффективности функционирования систем защиты компьютерных сетей от вирусных атак. В основу ее построения может быть положен принцип максимума неопределенности [23, 24]. Его суть состоит в том, что из всех возможных вариантов представления функции (1) выбирается вариант, обладающий максимальной неопределенностью при учете всей имеющейся эмпирической информации. Тем самым выбранный вариант минимизирует «домыслы» при оценке эффективности систем защиты компьютерных сетей от вирусных атак.

Если количество компьютеров в сети велико, то в качестве меры неопределенности распределения вероятностей дискретной случайной величины X принять энтропию Шеннона. При этом, вследствие ограниченности объема эмпирических данных о результатах вирусных атак, естественно полагать, что имеющаяся объективная информация исчерпывается знанием среднего количества $\bar{x}$ компьютеров сети, которые успел заразить вирус до его выявления и удаления.

В таких условиях построение функции (1) обеспечивается решением следующей экстремальной задачи

$$H = -\sum_{x=0}^{\infty} P(x)\ln P(x) \xrightarrow[P(x)]{} \max,\qquad (2)$$

$$\sum_{x=0}^{\infty} P(x) = 1,\qquad (3)$$

$$\bar{x} = \sum_{x}^{\infty} xP(x).\qquad (4)$$

Ее решение может быть получено методом Лагранжа. Для его применения введем неопределенные множители Λ_1, Λ_2 и исследуем на экстремум по $P(x)$ соответствующую задаче (2)–(4) функцию Лагранжа

$$L = -\sum_{x=0}^{\infty} P(x)\ln P(X) + \Lambda_1\left[1 - \sum_{x=0}^{\infty} P(x)\right] + \Lambda_2\left[\bar{x} - \sum_{x=0}^{\infty} xP(x)\right].\qquad (5)$$

Необходимым условием экстремума функции является равенство нулю ее первой производной. Следовательно, с учетом (5) получим:

$$\frac{dL}{dP(x)} = -\ln P(x) - 1 - \Lambda_1 - \Lambda_2 x = 0.\qquad (6)$$

Из (6) непосредственно следует, что

$$P(x) = e^{-1-\Lambda_1-\Lambda_2 x}.\qquad (7)$$

Введя обозначение

$$P_0 = e^{-1-\Lambda_1},\qquad (8)$$

соотношение (7) представим в виде

$$P(x) = P_0 e^{-\Lambda_2 x}.\qquad (9)$$

Полученное соотношение (9) содержит неизвестные параметры P_0 и Λ_2. Для их определения воспользуемся ограничениями (3), (4), накладываемыми на функцию распределения случайной величины X. С учетом (7) соотношения (3), (4) принимают вид

$$\sum_{x=0}^{\infty} P_0 e^{-\Lambda_2 x} = 1,\qquad (10)$$

$$\bar{x} = \sum_{x=0}^{\infty} xP_0 e^{-\Lambda_2 x}.\qquad (11)$$

Левая часть соотношения (10) представляет собой сумму убывающей геометрической прогрессии. С учетом этого соотношение (10) можно представить в виде

$$P_0 \frac{1}{1-e^{-\Lambda_2}} = 1 \quad . (12)$$

Правая часть соотношения (11) представляет собой сумму арифметико-геометрической прогрессии [25, 26]. Следовательно, соотношение (11) можно представить в виде

$$\overline{x} = \frac{P_0 e^{-\Lambda_2}}{\left(1-e^{-\Lambda_2}\right)^2}. \quad (13)$$

Из соотношений (12) и (13) следует, что

$$P_0 = \frac{1}{\overline{x}+1}, \quad (14)$$

$$\Lambda_2 = \ln\left(\frac{\overline{x}+1}{\overline{x}}\right). \quad (15)$$

Подставив (14), (15) в (9), получим

$$P(x) = \frac{1}{\overline{x}+1}\left(\frac{\overline{x}}{\overline{x}+1}\right)^x. \quad (16)$$

Следовательно, случайная величина X количества компьютеров сети, которые успел заразить вирус до его выявления и удаления подчиняется геометрическому распределению

$$P(x) = p(1-p)^x \quad (17)$$

с параметром

$$p = \frac{1}{1+\overline{x}}. \quad (18)$$

Вероятность же того, что количества компьютеров сети, которые успел заразить вирус до его выявления и удаления превысит величину $x_i (x_i = 0, 2, ...)$, определяется соотношением

$$Q(x_i) = 1 - \sum_{x=0}^{x_i} P(x). \quad (19)$$

Распределения (17) и (19) являются сформированными на основе малой выборки моделями оценки эффективности функционирования систем защиты компьютерных сетей от вирусных атак.

Заключение

В целом, предлагаемый подход к решению задачи оценки эффективности системы защиты КС с учетом случайности результатов отражения вирусных атак и ограниченности объема эмпирической выборки в максимальной степени учитывает полученную в результате натурных испытаний и опыта эксплуатации объективную эмпирическую информацию о последствиях вирусных атак. Практическое применение моделей (17), (19) позволяет получать приемлемые по точности оценки эффективности функционирования рассматриваемых систем даже при небольших объемах эмпирических данных.

СПИСОК ИСТОЧНИКОВ

1. **Vasil'ev Y.S., Zegzhda D. P., Poltavtseva M.A.** Problems of Security in Digital Production and its Resistance to Cyber Threats. Automatic Control and Computer Sciences. 2018. 52(8). P. 1090–1100.

2. **Anisimov V.** Digital Transformation and Optimization Models in the Sphere of Logistics / V. Anisimov [и др.] // SHS Web of Conf. 2018 т. 44. 00009. https://doi.org/10.1051/shsconf/20184400009.

3. **Лаврова Д. С., Хушкеев А. А**. Обнаружение нарушений информационной безопасности в АСУ ТП на основе прогнозирования многомерных временных рядов, сформированных из значений параметров работы конечных устройств системы // Проблемы информационной безопасности. Компьютерные системы. 2019. № 1. С. 18–30.

4. **Anisimov V. G.** The Problem of Innovative Development of Information Security Systems in the Transport Sector / V. G. Anisimov [и др.] // Automatic Control and Computer Sciences. 2018. Т. 52. № 8. С. 1105–1110. DOI: 10.3103/S0146411618080035.

5. **Зегжда П. Д**. Модель формирования программы развития системы обеспечения информационной безопасности организации / П. Д. Зегжда, Д. П. Зегжда [и др.] // Проблемы информационной безопасности. Компьютерные системы. 2021. № 2 (46). С. 109–117.

6. **Saurenko T. N**. Methodology Control Function Realization within the Electronic Government Concept Framework / T. N. Saurenko [и др.] // International Journal of Scientific and Technology Research. 2020. Т. 9. № 2. С. 6259–6262.

7. **Zegzhda P. D.** Approach to the Evaluation of the Efficiency of Information Security in Control Systems / P. D. Zegzhda [и др.] // Automatic Control and Computer Sciences, 2020, Vol. 54, No. 8, pp. 864–870. DOI: 10.3103/S0146411620080362.

8. **Anisimov V. G**. Models of Forecasting Destructive Influence Risks for Information Processes in Management Systems / V. G. Anisimov [и др.] // Информационно-управляющие системы [Information and Control Systems] 2019. № 5 (102). С. 18–23. DOI:10.31799/1684-8853-2019-5-18-23.

9. **Калинин М. О., Лаврова Д. С., Ярмак А. В**. Обнаружение угроз в киберфизических системах на основе методов глубокого обучения с использованием многомерных временных рядов // Проблемы информационной безопасности. Компьютерные системы. 2018. № 2. С. 111–117.

10. **Зегжда П. Д**. Эффективность функционирования компьютерной сети в условиях вредоносных информационных воздействий / П. Д. Зегжда [и др.] // Проблемы информационной безопасности. Компьютерные системы. 2021. № 1 (45). С. 96–101.

11. **Лаврова Д. С., Алексеев И. В., Штыркина А. А**. Анализ безопасности на основе контроля зависимостей параметров сетевого трафика с использованием дискретного вейвлет-преобразования // Проблемы информационной безопасности. Компьютерные системы. 2018. № 2. С. 9–15.

12. **Anisimov V. G.** A Risk-oriented Approach to the Control Arrangement of Security Protection Subsystems of Information Systems / V. G. Anisimov [и др.] // Automatic Control and Computer Sciences, 2016, 50(8). С. 717–721. . DOI: 10.3103/S0146411616080289.

13. **Зайцева Е. А.** Нейтрализация последствий деструктивных воздействий путём применения теории графов для переконфигурирования структуры системы / Е. А. Зайцева [и др.] // Методы и технические средства обеспечения безопасности информации. 2019. № 28. С. 7–8.

14. **Anisimov V. G**. Indices of the Effectiveness of Information Protection in an Information Interaction System for Controlling Complex Distributed Organizational Objects / V. G. Anisimov [и др.] // Automatic Control and Computer Sciences. 2017, 51(8), С. 824–828. https://doi.org/10.3103/S0146411617080053.

15. **Zegzhda P. D.** A Model of Optimal Complexification of Measures Providing Information Security / P. D. Zegzhda [и др.] // Automatic Control and Computer Sciences, 2020, т. 54, № 8. С. 930–936. DOI: 10.3103/S0146411620080374.

16. **Анисимов В. Г., Селиванов А. А., Анисимов Е. Г**. Методика оценки эффективности защиты информации в системе межведомственного информационного взаимодействия при управлении обороной государства // Информация и космос. 2016. № 4. С. 76–80.

17. **Тебекин А. В**. Эволюционная модель прогноза частных показателей инновационных проектов (на примере технических инноваций) / А. В. Тебекин [и др.] // // Журнал исследований по управлению. 2019. Т. 5. № 6. С. 55–61.

18. **Anisimov V. G**. Model and Method for Optimizing Computational Processes in Parallel Computing Systems / V. G. Anisimov [и др.] // Automatic Control and Computer Sciences, 2019, Т. 53, № . 8, С. 1038–1044. DOI: 10.3103/S0146411619080054.

19. **Сауренко Т. Н**. Методика оценки ожидаемой стоимости проектирования технических и технологических инноваций / Т. Н. Сауренко [и др.] // Управленческое консультирование. 2019. № 11 (131). С. 120–128.

20. **Ильин И. В**. Математические методы и инструментальные средства оценивания эффективности инвестиций в инновационные проекты / И. В. Ильин [и др.]. — Санкт-Петербург, 2018. — 289 с.

21. **Anisimov V.** Efficiency of Ensuring the Survivability of Logistics Information and Control Systems / V Anisimov [и др.] // E3S Web of Conferences. 2020, т. 217, 07025. https://doi.org/10.1051/e3sconf/202021707025.

22. **Zotova E.** A Model for Setting up Development Programs for Logistics Systems in the Electric Power Industry to Achieve Electric Power Security/ E. Zotova [и др.] // E3S Web of Conferences. 2021, т. 258, 2027. DOI: 10.1051/e3sconf/202125802027.

23. **Jaynes, E. T.** Information Theory and Statistical Mechanics // *Phys. Rev.,* 1957. т. 106, с.. 620.

24. **Мартыщенко Л. А**. Принцип максимума неопределенности и его приложения. — М.: МО СССР, 1984. 42 с.

25. **Ямпольский С. М**. Научно-методические основы информационно-аналитического обеспечения деятельности органов государственного и военного управления в ходе межведомственного информационного взаимодействия / С. М. Ямпольский [и др.]. — Москва: Военная академия Генерального штаба Вооруженных Сил Российской Федерации, Военный институт (управления национальной обороной). 2019. — 146 с.

26. **Балясников В. В.** Модель причинного анализа на основе использования данных об особых ситуациях / В. В. Балясников [и др.] // Вопросы оборонной техники. Серия 16: Технические средства противодействия терроризму. 2015. № 1–2. С. 31–38.

REFERENCES

1. **Vasil'ev Y.S., Zegzhda D. P., Poltavtseva M. A**. Problems of Security in Digital Production and its Resistance to Cyber Threats. Automatic Control and Computer Sciences. 2018. 52(8). P. 1090–1100.

2. **Anisimov V.** Digital Transformation and Optimization Models in the Sphere of Logistics / V. Anisimov [i dr.] // SHS Web of Conf. 2018 t. 44. 00009. https://doi.org/10.1051/shsconf/20184400009.

3. **Lavrova D. S., Hushkeev A. A**. Obnaruzhenie narushenij informacionnoj bezopasnosti v ASU TP na osnove prognozirovaniya mnogomernyh vremennyh ryadov, sformirovannyh iz znachenij parametrov raboty konechnyh ustrojstv sistemy // Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy. 2019. № 1. S. 18–30. (In Russian)

4. **Anisimov V. G**. The Problem of Innovative Development of Information Security Systems in the Transport Sector / V. G. Anisimov [i dr.] // Automatic Control and Computer Sciences. 2018. T. 52. № 8. S. 1105–1110. DOI: 10.3103/S0146411618080035.

5. **Zegzhda P. D**. Model' formirovaniya programmy razvitiya sistemy obespecheniya informacionnoj bezopasnosti organizacii / P. D. Zegzhda, D. P. Zegzhda [i dr.] // Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy. 2021. № 2 (46). S. 109–117. (In Russian)

6. **Saurenko T. N**. Methodology Control Function Realization within the Electronic Government Concept Framework / T. N. Saurenko [i dr.] // International Journal of Scientific and Technology Research. 2020. T. 9. № 2. S. 6259–6262.

7. **Zegzhda P. D.** Approach to the Evaluation of the Efficiency of Information Security in Control Systems / P. D. Zegzhda [i dr.] // Automatic Control and Computer Sciences, 2020, Vol. 54, No. 8, pp. 864–870. DOI: 10.3103/S0146411620080362.

8. **Anisimov V. G**. Models of Forecasting Destructive Influence Risks for Information Processes in Management Systems / V. G. Anisimov [i dr.] // Informacionno-upravlyayushchie sistemy [Information and Control Systems] 2019. № 5 (102). S. 18–23. DOI:10.31799/1684-8853-2019-5-18-23.

9. **Kalinin M. O., Lavrova D. S., Yarmak A. V**. Obnaruzhenie ugroz v kiberfizicheskih sistemah na osnove metodov glubokogo obucheniya s ispol'zovaniem mnogomernyh vremennyh ryadov // Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy. 2018. № 2. S. 111–117. (In Russian)

10. **Zegzhda P. D**. Effektivnost' funkcionirovaniya komp'yuternoj seti v usloviyah vredonosnyh informacionnyh vozdejstvij / P. D. Zegzhda [i dr.] // Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy. 2021. № 1 (45). S. 96–101. (In Russian)

11. **Lavrova D. S., Alekseev I. V., Shtyrkina A. A**. Analiz bezopasnosti na osnove kontrolya zavisimostej parametrov setevogo trafika s ispol'zovaniem diskretnogo vejvlet-preobrazovaniya // Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy. 2018. № 2. S. 9–15. (In Russian)

12. **Anisimov V. G.** A Risk-oriented Approach to the Control Arrangement of Security Protection Subsystems of Information Systems / V. G. Anisimov [i dr.] // Automatic Control and Computer Sciences, 2016, 50(8). S. 717–721. . DOI: 10.3103/S0146411616080289.

13. **Zajceva E. A**. Nejtralizaciya posledstvij destruktivnyh vozdejstvij putyom primeneniya teorii grafov dlya perekonfigurirovaniya struktury sistemy / E. A. Zajceva [i dr.] // Metody i tekhnicheskie sredstva obespecheniya bezopasnosti informacii. 2019. № 28. S. 7–8. (In Russian)

14. **Anisimov V. G**. Indices of the Effectiveness of Information Protection in an Information Interaction System for Controlling Complex Distributed Organizational Objects / V. G. Anisimov [i dr.] // Automatic Control and Computer Sciences. 2017, 51(8), S. 824–828. https://doi.org/10.3103/S0146411617080053.

15. **Zegzhda P. D.** A Model of Optimal Complexification of Measures Providing Information Security / P. D. Zegzhda [i dr.] // Automatic Control and Computer Sciences, 2020, t.54, № 8. S. 930–936. DOI: 10.3103/S0146411620080374.

16. **Anisimov V. G., Selivanov A. A., Anisimov E. G**. Metodika ocenki effektivnosti zashchity informacii v sisteme mezhvedomstvennogo informacionnogo vzaimodejstviya pri upravlenii oboronoj gosudarstva // Informaciya i kosmos. 2016. № 4. S. 76–80. (In Russian)

17. **Tebekin A. V**. Evolyucionnaya model' prognoza chastnyh pokazatelej innovacionnyh proektov (na primere tekhnicheskih innovacij) / A. V. Te-

bekin [i dr.] // // ZHurnal issledovanij po upravleniyu. 2019. T. 5. № 6. S. 55–61. (In Russian)

18. **Anisimov V. G**. Model and Method for Optimizing Computational Processes in Parallel Computing Systems / V. G. Anisimov [i dr.] // Automatic Control and Computer Sciences, 2019, T. 53, № . 8, S. 1038–1044. DOI: 10.3103/S0146411619080054.

19. **Saurenko T. N**. Metodika ocenki ozhidaemoj stoimosti proektirovaniya tekhnicheskih i tekhnologicheskih innovacij / T. N. Saurenko [i dr.] // Upravlencheskoe konsul'tirovanie. 2019. № 11 (131). S. 120–128. (In Russian)

20. **Il'in I. V**. Matematicheskie metody i instrumental'nye sredstva ocenivaniya effektivnosti investicij v innovacionnye proekty / I. V. Il'in [i dr.]. — Sankt-Peterburg, 2018. — 289 s. (In Russian)

21. **Anisimov V.** Efficiency of Ensuring the Survivability of Logistics Information and Control Systems / V Anisimov [i dr.] // E3S Web of Conferences. 2020, t. 217, 07025. https://doi.org/10.1051/e3sconf/202021707025.

22. **Zotova E.** A Model for Setting up Development Programs for Logistics Systems in the Electric Power Industry to Achieve Electric Power Security/ E. Zotova [i dr.] // E3S Web of Conferences. 2021, t. 258, 2027. DOI: 10.1051/e3sconf/202125802027.

23. **Jaynes, E. T.** Information Theory and Statistical Mechanics // Phys. Rev., 1957. t. 106, s.. 620.

24. **Martyshchenko L. A**. Princip maksimuma neopredelennosti i ego prilozheniya. — M.: MO SSSR, 1984. 42 s.

25. **Yampol'skij S. M**. Nauchno-metodicheskie osnovy informacionno-analiticheskogo obespecheniya deyatel'nosti organov gosudarstvennogo i voennogo upravleniya v hode mezhvedomstvennogo informacionnogo vzaimodejstviya / S.M. YAmpol'skij [i dr.]. — Moskva: Voennaya akademiya General'nogo shtaba Vooruzhennyh Sil Rossijskoj Federacii, Voennyj institut (upravleniya nacional'noj oboronoj). 2019. — 146 s. (In Russian)

26. **Balyasnikov V. V**. Model' prichinnogo analiza na osnove ispol'zovaniya dannyh ob osobyh situaciyah / V. V. Balyasnikov [i dr.] // Voprosy oboronnoj tekhniki. Seriya 16: Tekhnicheskie sredstva protivodejstviya terrorizmu. 2015. № 1–2. S.31–38. (In Russian)

Статья поступила в редакцию 09.10.2021; одобрена после рецензирования 16.12.2021; принята к публикации 12.02.2022.

The article was submitted 09.10.2021; approved after reviewing 16.12.2021; accepted for publication 12.02.2022.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

АНИСИМОВ Владимир Георгиевич — д. воен. н., профессор; Санкт-Петербургский политехнический университет Петра Великого

ANISIMOV Vladimir G. — Doctor of Military Sciences, professor; St. Petersburg Polytechnic University of Peter the Great

АНИСИМОВ Евгений Георгиевич — д.т.н., д. воен. н., профессор; Российский университет дружбы народов

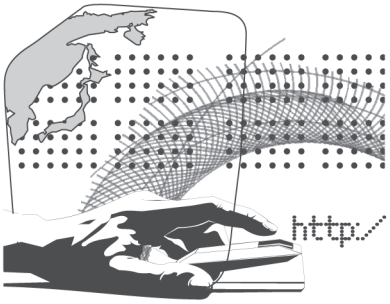
ANISIMOV Evgeny G. — Doctor of Technical Sciences, Doctor of Military Sciences, professor; Peoples' Friendship University of Russia

САУРЕНКО Татьяна Николаевна — д.э.н., профессор; Российский университет дружбы народов

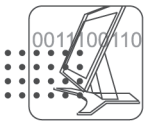
SAURENKO Tatyana N. — Doctor of Economics, Professor; Peoples' Friendship University of Russia

ЛОСЬ Владимир Павлович — д. воен. н., профессор; МИРЭА — Российский технологический университет

LOS Vladimir P. — Doctor of Military Sciences, professor; MIREA — Russian Technological University



МЕТОДЫ И СРЕДСТВА ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ



Научная статья
DOI 10.48612/jisp/pmmp-3uhf-npad
УДК 004.056

А. А. Полтавцев

Тверской государственный технический университет (ТвГТУ)
Россия, 170026, г. Тверь, наб. Афанасия Никитина, д. 22

РЕАКТИВНЫЕ И ПРОАКТИВНЫЕ МЕТОДЫ ЗАЩИТЫ БАЗ ДАННЫХ ОТ АТАК ЛОГИЧЕСКОГО ВЫВОДА

Аннотация. Если данные недоступны для внешнего мира, они бесполезны. Данные должны быть доступны, чтобы можно было проводить необходимую обработку и планирование. Регулирование и мониторинг доступа пользователей к базе данных является одной из важных задач сообщества безопасности баз данных. Защита базы данных от атак логического вывода является частью информационной безопасности, которая пытается предотвратить раскрытие закрытой информации через доступную (таблицы, отдельные записи). Необходимо иметь методы, способные поддерживать баланс между использованием информации и защитой данных. Цель работы состоит в сравнении различных методов управления логическим выводом с целью оценки результатов методов для минимизации как потери информации, так и риска раскрытия информации.

Ключевые слова: информационная безопасность, мониторинг безопасности, управление безопасностью, структуризация данных, управление данными

Research article
DOI 10.48612/jisp/pmmp-3uhf-npad

A. A. Poltavtsev

Tver State Technical University (TvSTU)
Russia, Tver

DATABASE PROTECTION AGAINST INSIGHT ATTACKS REACTIVE AND PROACTIVE METHODS

Abstract. If the data is not available to the outside world, it is useless. The data must be available so that the necessary processing and planning can be carried out. Regulating and monitoring user access to a database is one of the important tasks of the database security community. Database protection against inference attacks is a part of information security that tries to prevent the disclosure of sensitive information through available information (tables, individual records). It is necessary to have methods capable of maintaining a balance between the use of information and the protection of data. The purpose of this work is to compare different inference control methods in order to evaluate the methods results to minimize both the loss of information and the risk of information disclosure.

Keywords: Information Security, Security Monitoring, Security Control, Data Structuring, Data Engineering

Защита данных является серьезной проблемой во всех областях [1–4], и она более всего актуальна, когда злоумышленник пытается прямо или косвенно получить информацию из базы данных [5]. База данных содержит большой объем данных, критически важных для работы как коммерческих, так и государственных предприятий. Эти данные содержат и могут служить источником конфиденциальной информации, которая интересна для тех, кто является конкурентом или противником. Используя доступные системы управления безопасными базами данных, предприятие может обеспечить различные степени защиты данных. Эта защита может варьироваться от прав доступа до защиты на основе меток, когда метки безопасности назначаются данным в зависимости от их конфиденциальности. Несанкционированный доступ к этим данным осуществляется опосредовано через привилегии тех, кто пытается получить к ним доступ.

Принципиально существует два способа нарушить конфиденциальность: первый — прямой способ, при котором злоумышленник не является пользователем базы данных, но получает конфиденциальную информацию с помощью несанкционированного доступа; второй — косвенное раскрытие, когда злоумышленником является пользователь базы данных с определенным уровнем доступа, который получает конфиденциальную информацию из набора разрешенных ему не конфиденциальных запросов. Подобная проблема стояла всегда, но в настоящей момент стала еще более острой в связи с появлением хранилищ данных, содержащих большие данные [6–9], а также новых подходов к управлению данными, включая облачные технологии или blockchain [10–11].

К сожалению, даже надлежащая защита отдельных частей базы данных (прямой способ) не может обеспечить полную защиту. Конкурент или противник может иметь возможность использовать данные, которые по отдельности кажутся должным образом защищенными, для вывода конфиденциальных данных. Проблема состоит в том, чтобы обнаружить эти попытки, а затем принять необходимые контрмеры, чтобы закрыть их.

Атака логического вывода происходит в многоуровневой защищенной базе данных, когда низкоуровневый пользователь может вывести конфиденциальную информацию с помощью общих знаний и авторизованных ответов на запросы.

Косвенное обнаружение состоит в том, что пользователь-злоумышленник (далее ПЗ) выполняет несколько SQL-запросов к тем данным базы данных, к которым у него есть допуск. Анализируя результаты этих запросов, он пытается вычислить закрытые данные.

Косвенное раскрытие отличается от других проблем безопасности тем, что формально это не проблема безопасности, в отличии от первого способа. В этом случае пользователи-злоумышленники фактически не нарушают никаких правил безопасности. Они выводят информацию, обладающую высоким риском безопасности, из набора доступных запросов, имеющих низкий риск безопасности.

Контроль вывода (контроль раскрытия) направлен на защиту данных от косвенного обнаружения. Этот процесс гарантирует, что запросы неконфиденциальных данных при их объединении не раскрывают конфиденциальную информацию. Цель управления логическим выводом — помешать пользователю завершить какой-либо канал вывода.

Например, статистическая база данных содержит закрытые личные данные, а предоставлять должна только различные виды статистических сводок о населении. В частности, при переписи населения собирается информация о гражданах, но предоставляется только резюме этой информации без раскрытия каких-либо подробностей. Медицинские информационные системы должны предоставлять статистические данные о здоровье населения, но не публиковать данные о здоровье какого-либо пациента. Может оказаться,

что злоумышленник в состоянии восстановить эту информацию путем обработки достаточного количества статистической информации. Это называется *обнаружение конфиденциальной информации с помощью логического вывода* (*deduction of confidential information by inference*). Когда информация относится к человеку, обнаружение ставит под угрозу его личную жизнь.

Общее решение проблемы логического вывода крайне сложно, поскольку злоумышленник может применить обширные знания при выполнении атаки логического вывода. В данной проблеме предполагается, что противник обладает некоторыми правами, а также знаком с конкретной областью знаний. Все эти знания он применяет при выполнении атаки логического вывода. Работа по обнаружению и предотвращению логических выводов из базы данных продолжается и находится все еще на стадии исследования. Однако тот факт, что полного решения общей проблемы вывода нет, не означает, что сегодня нет доступных практических методов, которые можно было бы применить к конкретным ситуациям.

Основная задача методов контроля логических выводов (**ICTQ**) — обеспечить лучшую защиту от разглашения с минимальными потерями информации. Защита, обеспечиваемая ICTQ, обычно включает в себя некоторую степень модификации данных, которая является промежуточным вариантом между отсутствием модификации (максимальная полезность, но без защиты от раскрытия информации) и шифрованием данных (максимальная защита, но без полезности для пользователя). Задача ICTQ состоит в изменении данных таким образом, чтобы обеспечить достаточную защиту при минимальной потере информации.

Примером логического вывода может быть вывод о том, что компания внедряет новый процесс производства компьютерных микросхем на основе заказа определенных типов оборудования.

Приведём следующий пример. Конфиденциальной является информация о том, какую оценку получил студент. Злоумышленник пытается получить оценку студента Иванова. Ему доступен список оценок учащихся вместе с номером зачетки (который он не знает), и список фамилий студентов (в нем номеров зачеток нет). Если он знает, что оба списка отсортированы по номерам зачеток то он получает возможность определить оценку Иванова. Для этого он из второго списка определяет порядковый номер Иванова, а затем по этому порядковому номеру из первого списка определяется оценка.

Третий пример. Просматривая список возрастов членов пенсионного сообщества, злоумышленник замечает, что никому из них не меньше 55 лет. Из этого он может вывести правило, что членами сообщества могут быть люди не менее 55 лет.

Четвертый пример. Подготовка к нападению может быть определена по записям в базе данных о запросах материалов и перемещениях войск.

Классификация методов защиты баз данных от атак логического вывода

Прежде всего отметим, что контроль раскрытия может быть применен на одном их двух или на обоих этапах жизненного цикла программного обеспечения, называемого СУБД. Во-первых, он может быть применен как к работающей БД. Во-вторых, уязвимостям логического вывода можно противодействовать путем доработки методов проектирования баз данных, путем включения в них процедур, учитывающих логический вывод.

Методы, которые можно применить к базе данных во время ее проектирования, называются проактивными методами. Эти методы не требуют наличия данных, требуется только схема базы данных. Те методы, которые могут быть применены к существующей базе данных, называются реактивными методами. Эти методы используют экземпляры данных для своего анализа.

Классификация методов зависит и от типа данных, к которым они применяются. Некоторые методы применяются к микроданным, а некоторые — к табличным данным.

Микроданные представляют собой серию кортежей, каждый из которых содержит информацию об отдельном объекте. Это наиболее распространенный случай. В таблице 1 некоторые атрибуты являются конфиденциальными (SSN, тип диагностики, доход), и нельзя предоставлять этот тип конфиденциальной информации.

Табличные данные (tabular data). В табличном представлении дан ных каждое поле реляционного отношения представляет количество записей для набора атрибутов, связанных друг с другом. Данный случай возникает чаще всего в статистических базах данных. Атрибуты относятся к категориальному типу, и, по крайней мере для некоторых атрибутов, количество записей может быть мало. По этим небольшим значениям можно вывести информацию конкретного кортежа за несколько шагов.

В таблице 2 показано, как могут быть представлены табличные данные. Заданные значения полей — количество записей для определенной категории.

Данная работа посвящена анализу проективных методов. Тем не менее, разные группы методов имеют определенную корреляцию поэтому вначале кратко перечислим основные группы реактивных методов.

Реактивные методы

Реактивный подход борьбы с логическим выводом состоит или в ограничении множества запросов, предоставляемых пользователю, или в изменении данных, возвращаемых запросами. В обоих случаях необходимо специальное программное обеспечение — API к базе данных, которое содержит соответствующие алгоритмы борьбы с проникновением.

Как известно, реляционная модель умеет работать только с функциональными зависимостями. Основные структурные ограничения реляционной модели — первичные и внешние ключи — с математической точки зрения являются функциональными зависимостями. В первом случае — между атрибутами одной реляционной схемы, а во втором — между атрибутами двух разных реляционных схем.

Табличная защита данных. Цель — опубликовать агрегат информации таким образом, чтобы нельзя было вывести конфиденциальную информацию, содержащуюся в таблице. Пользователь может отправлять статистические запросы (суммы, средние, количество и т.д.) в базу данных. Информация, полученная пользователем в результате последовательных запросов, не должна позволять ему делать выводы о защищаемых данных.

Таблица 1. Микроданные

Table 1. Microdata

Name	SSN	Age	Zip	Diagnosis	**Income**
Ajay	11543	23	223621	AIDS	17300
Rakhi	11536	19	223622	AIDS	12456
Vivek	11524	26	223621	SARS	45213
Raj	11511	31	223625	SF	15482

Таблица 2. Табличные данные

Table 2. Tabular data

Sex_MS/Age	**0–15**	**16–25**	**26–35**	**36–50**	**>50**
Male_Married	12	9	0	4	21
Male_Single	0	5	11	8	3
Female_Married	9	13	47	25	21
Female_Single	0	14	25	7	17

Защита микроданных. Цель — минимизировать риск раскрытия информации.

Каналом вывода называется последовательность запросов, ответы на которые позволяют сделать конфиденциальный вывод. Канал вывода — это канал, в котором пользователи могут найти доступный элемент X, а затем использовать его для получения закрытого элемента Y, так как $Y = f(X)$.

Прежде чем анализировать методы, рассмотрим несколько простых примеров раскрытия информации.

Пример 1. На кафедре есть много профессоров в разных возрастных группах, и доступными являются только данные о средней зарплате профессоров заданной в запросе возрастной категории. Если есть только один профессор, возраст которого меньше 40, то из доступных средних данных можно вывести зарплату профессора, которому меньше 40 лет.

```
Q1: select avg (salary) from staff;
Q2: select avg (salary) from staff
where age > 40;
```

Два этих запроса совместно образуют канал вывода и выводят конфиденциальную информацию.

Пример 2. Рассмотрим систему управления «Корабль-порт».

```
Q1: select ship_name, port_name
from ships;
Q2: select port_name from ports
where portWeapon = 'yes';
```

Эти два запроса вместе составляют канал вывода, потому что, если оба они сделаны, можно точно определить, какие корабли несут оружие, а это может быть конфиденциальной информацией.

Рассмотрим различные методы контроля вывода, которые помогают контролировать раскрытие информации. Существует два основных подхода и набор методов, обеспечивающих защиту от косвенного раскрытия информации. Достаточно часто методы управления логическим выводом подразделяют на две широкие категории: первая — пертурбативная (добавление шума), а вторая — непертурбативная [12].

Добавление шума к данным. Применяется на этапе эксплуатации базы данных. Перед публикацией набор данных искажается. Во избежание потери информации, используемый метод возмущения должен быть таким, чтобы агрегированные данные, вычисленные на возмущенном наборе данных, не отличались существенно от данных, которые были бы получены на исходном наборе данных [12]. Добавление шума дает ответы на все вопросы, но ответы являются приблизительными. Добавление шума может быть выполнено через:

- округление (rounding techniques);
- микроагрегацию (microaggregation);
- обмен данными (data swapping);
- нарушение данных (data perturbation);
- выходное возмущение (output perturbation).

Методы округления. Методы округления заменяют исходные значения атрибутов округленными значениями. Когда таблица защищена округлением, изменение зависит от базы округления b. Защиту, обеспечиваемую округлением, можно измерить неопределенностью в отношении истинных значений. Поэтому ширина интервала, содержащего возможное истинное значение для данного округленного значения, называемого интервалом существования, принимается в качестве меры защиты.

Микроагрегация. При микроагрегации данные перед выводом группируются в небольшие агрегаты. Среднее значение группы заменяет каждое отдельное значение. Группы формируются по критерию максимального сходства. Полученные (измененные) данные могут быть выведены. Это помогает предотвратить разглашение точных значений личных данных. Как указано в [13], микроагрегация применима только тогда, когда выводятся наборы данных, причем записи в наборе соответствуют группе из k или более показателей, где ни один показатель не доминирует, т.е. вносит слишком большой вклад в группу, а k — пороговое значение. Пользователь получает результаты для любого запроса к базе данных, но результаты являются приблизительными. Первая исходная база

данных конвертируется в усредненную базу данных, и к этой базе данных пользователь делает запрос получая приблизительные результаты. Понятно, что такое применимо только к числовым значениям, например, в статистических базах данных.

Обмен данными. В этом методе к данным применяется последовательность так называемых элементарных свопов. Элементарный своп состоит из двух действий:

а) случайный выбор двух записей i и j из данных,

б) обмен значений атрибута в них.

Понятно, что такое применимо только в достаточно ограниченном наборе случаев.

Зашумление данных. В этом методе сначала к необработанным данным добавляется шум и создается возмущенная база данных. Пользователь запрашивает только возмущенную базу данных. Таким образом, и здесь пользователь получает приблизительный результат. Плюсы и минусы те же.

Возмущение на выходе. Вместо преобразования необработанных данных, как в data perturbation, искажаются только выходные данные (результаты запроса).

Без добавления шума. Методы этой категории не изменяют данные, они производят частичное уменьшение детализации в наборе данных. В этой категории, если запрос удовлетворяет условию, то будет получен точный результат. Основные техники:

- ограничение таблицы (table restriction);
- ограничение запроса (query restriction);
- подавление поля (cell suppression).

Ограничение таблицы. Ограничивается доступ ко всей таблице, которая имеет риск раскрытия информации. Этот метод приводит к большой потере информации, поскольку для защиты нескольких элементов в таблице необходимо ограничить данные всей таблицы. Реализуется путем подсчета количества атрибутов, фигурирующих в характеристической формуле запроса, если атрибутов больше d, запрос отклоняется. Порог d является параметром базы данных.

Ограничение запроса. Ограничение запроса отклоняет запрос, который может привести к компрометации данных. Основной прием — управление размером набора запросов. Запрос отклоняется, если он содержит слишком мало или слишком много записей [14]. Система управления размером набора запросов ограничивает количество записей, которые должны быть в наборе результатов. Разрешается вывод только в том случае, если размер R набора запросов (то есть количество сущностей, включенных в ответ на запрос) удовлетворяет условию. Отображаются результаты запроса только в том случае, если размер набора запросов удовлетворяет заданному условию.

При запросе данных, запрос содержит характеристическую формулу C, т.е. логическую формулу, в которой условия объединяются логическими операторами (И, ИЛИ, НЕ). Множество записей, содержание которых удовлетворяет формуле C называется множеством запроса для C (query set for C — далее QS_c). Данные, возвращаемые запросом, вычисляются из этого множества. Это может быть количество записей, сумма значений или расчетное значение, такое как максимум или медиана.

Результат является компрометирующим, если из него можно вывести конфиденциальные данные, сопоставив ответы на запросы с использованием любой предварительной информации, которую злоумышленник имеет. Наиболее часто такие методы взлома основаны на построении нужной злоумышленнику записи путем пересечения множеств результатов, возвращаемых несколькими запросами.

Рассмотрим пример медицинской базы данных позволяющей получать статистические данные. Злоумышленник хочет получить информацию о болезни своего начальника. Он запрашивает статистическую информацию, используя ту информацию, которую злоумышленник знает о нем:

```
Сколько пациентов имеют следующие
характеристики:
C = (мужчина, возраст 45-50, же-
нат. двое детей, вице-президент
банка)
Ответ [QSc]=1. (Здесь [A] как
обычно мощность множества A.)
```

Т.к. $[QS_c]=1$, то данный запрос однозначно определяет искомого индивидуума. Можно получить информацию о том, болеет ли он, например, депрессией:

Сколько пациентов имеют следующие характеристики:

C = (мужчина, возраст 45-50, женат. двое детей, вице-президент банка, депрессия)

Ответ: $[QS_c]$ = 1 или 0.

Здесь если $[QS_c]=1$, то болеет и если $[QS_c]=0$, то нет.

Принцип атаки прост. Сначала строится запрос (с некоторой формулой C), для которого множество запроса имеет мощность 1. Затем задается следующий запрос, с формулой C∧X, где X — интересующая характеристика индивидуума. Ответ 1 говорит о том, что индивидуум обладает характеристикой X, и 0 о том, что нет. Впервые такой тип атаки был описан в [21].

На этом принципе строятся и более сложные атаки. Основная идея состоит в том, чтобы дополнить небольшие множества запросов достаточным количеством дополнительных кортежей, чтобы они давали интересующую информацию, а затем вычесть эффект дополнительных записей. В [22] формула, идентифицирующая дополнительные записи, была названа трекером, потому что спрашивающий использует ее, чтобы «отследить» интересующие его характеристики.

Пусть злоумышленник знает, что объект I однозначно характеризуется формулой C, однако запрос с такой формулой ему не доступен. Пусть также он может представить этот запрос в виде C= (A∧B). Он сможет получить интересующую его информацию, если система контроля доступа разрешает ему выполнять запросы с формулами A и (A∧¬B). Последняя формула T = (A∧¬B) и была названа трекером для I. Рассмотрим использование трекера на примере.

Пусть задано реляционное отношение:

Из таблицы 3 видно, что формула C=(Доктор∧Ж) однозначно идентифицирует кортеж (Петрову). Следующие запросы показывают, как депозит Петровой может быть выведен, используя в качестве трекера формулу T = (Доктор∧Ж). Используется то, что

$\{QS_c\}$ДОКТОР = $\{QS_c\}$ДОКТОР∨Ж ∪ ∪ $\{QS_c\}$ДОКТОР∧¬Ж

1. Сколько вкладчиков имеют следующие характеристики:

C = (Доктор)

Ответ $[QS_c]_1 = 3$.

2. Сколько вкладчиков имеют следующие характеристики:

C = (Доктор∧¬Ж) = (Доктор∧М)

Ответ $[QS_c]_2 = 2$.

3. $[QS_c]_1 - [QS_c]_2 = 1$. Т.е. [$\{QS_c\}$ДОКТОР∧Ж] = 1

Злоумышленник знает, что Петрова — женщина и доктор. Таким образом, из первых двух запросов он убеждается что они однозначно идентифицируют Петрову. Поэтому на их основании он строит новые запросы — групповые (т.е. разрешенные ему), чтобы выяснить закрытую информацию о депозите Петровой:

Таблица 3. Реляционное отношение

Table 3. Relational ratio

Фамилия	Пол	Профессия	Депозит(S)
Брагин	М	Доктор	3000
Абдулов	М	Доктор	500
Иванов	М	Учитель	1
Петрова	Ж	Доктор	5000
Сидорова	Ж	Водитель	1000
Вавилов	М	Водитель	20000
Демина	Ж	Швея	2000
Фомин	М	Слесарь	10000

4. Какой суммарный депозит вкладчиков, имеющих следующие характеристики:
C = (Доктор)
Ответ $[QS_c]_4$ = 8500.
5. Какой суммарный депозит вкладчиков, имеющих следующие характеристики:
C = (Доктор∧¬Ж) = (Доктор∧М)
Ответ $[QS_c]_5$ = 3500.
6. $[QS_c]_4 - [QS_c]_5$ = 8500–3500 = 5000

Данный пример иллюстрирует атаку, которая использует комбинаторный принцип включения/исключения, чтобы изолировать запись. Возможно использование и других принципов, подходов для построения атак логического вывода.

Здесь, в частности, можно упомянуть методы интеллектуального анализа данных (data mining), которые призваны обнаруживать формально не содержащуюся информацию в базе данных, используя хранимую информацию. Хотя методы извлечения данных полезны, они могут представлять угрозу безопасности, если добываемые данные содержат информацию, которая считается «конфиденциальной». Большинство этих методов основано на использовании правил ассоциации. Данные правила могут быть использованы и при атаках на базу данных и при построении защиты от них (обнаружения несанкционированного вывода из базы данных).

Наиболее распространенный метод защиты от таких атак состоит в следующем. Пусть K — минимальное количество записей, которые должны присутствовать в наборе запроса. R — размер набора запросов. L — размер базы данных (количество сущностей, представленных в базе данных). Набор запросов отображается только в том случае, если K ≤ R ≤ L-K. K должен удовлетворять условию 0 ≤ K ≤ (L/2).

Приведём пример. Пусть K=3. Тогда если запрос COUNT (birth_place='pl1' AND street_ no='A123') возвращает 5, то он разрешен, а если 2 — запрещен.

Пороговое значение k может ограничить слишком много запросов и все же не гарантирует защиту от компрометации. База данных может быть взломана серией запросов.

Подавление поля. Основная идея — удалить поля, содержащие конфиденциальную информацию. Анализируются линейные соотношения между всеми полями таблицы (включая предельные суммы в родительских таблицах), чтобы определить, можно ли вывести данные конфиденциальных полей (точно или приблизительно) по тем, которые предоставляются в запросе [15].

Общая процедура определения поля как чувствительного (раскрывающее поле) зависит от того, могут ли данные, содержащиеся в поле, быть вычисленными менее чем за t (заданный порог) отдельных записей. Если могут, то поле называется полем раскрытия. Как только поле раскрытия X было идентифицировано, определяется интервал раскрытия I(X) [16].

Предположим, что отношение содержит положительные целые числа и после подавления полей имеет вид [17]:

В таблице выполняются следующие линейные уравнения:

с x_1: $x_1 + x_2 = 814$, $x_1 + x_3 = 2401$, $x_i \geq 0$
с x_2: $x_1 + x_2 = 814$, $x_2 + x_4 = 1517$, $x_i \geq 0$
с x_3: $x_1 + x_3 = 2401$, $x_3 + x_4 = 1830$, $x_i \geq 0$
с x_4: $x_3 + x_4 = 1830$, $x_2 + x_4 = 1517$, $x_i \geq 0$

Таблица 4. Отношение после подавления полей

Table 4. Relation after margin suppression

MS/AG	0–15	16–25	26–35	36–50	>50	Total
Single	1234	656	415	125	698	3128
Married	457	$\mathbf{x_1}$	789	896	$\mathbf{x_2}$	2956
Divorced	856	$\mathbf{x_3}$	587	621	$\mathbf{x_4}$	3894
Total	2547	3057	1791	2436	2215	8878

Наличие данных уравнений порождает риск раскрытия конфиденциальной информации (значения x_1, x_2, x_3, x_4). Формализация понятия степени этого риска и формулы для его вычисления определены в [6].

Подавление полей бывает двух типов: первичное и вторичное. При первичном подавлении подавляется только поле, которое является полем раскрытия (чувствительным). Из-за линейных соотношений между предельными суммами некоторые дополнительные нечувствительные поля (поля не раскрывающие информацию) также должны быть подавлены, чтобы обеспечить требуемую интервальную защиту для X. Определение и подавление этих дополнительных полей называется вторичным подавлением [6]. В случае вторичного подавления потеря информации больше, поэтому для того, чтобы потеря информации была меньше, количество подавляемых ячеек должно было быть минимальным.

Каждый метод обладает своей степенью защиты, эффективностью, объемом потерь информации. Риск потери информации обратно пропорционален риску раскрытия информации, потеря информации тем больше, чем меньше риск ее раскрытия и наоборот.

В пертурбативных методах ответ дается на каждый запрос, но все данные являются приблизительными. В непертурбативных методах результативные данные точны: если запрос удовлетворяет критериям, то запрос будет выполнен, а если нет, то отклонен. Может оказаться, что нет ответа на все запросы.

Правильный выбор метода контроля раскрытия информации уменьшает потерю информации и снижает риск ее раскрытия. Ни один отдельный метод управления логическим выводом не отвечает сразу нескольким целям: высокой защите, низкой потери информации, эффективности. Ни один метод не может обеспечить полной защиты и одновременно минимизировать потерю информации. Поэтому для достижения сочетания этих свойств необходимо комбинировать методы. Для одновременной минимизации риска потери информации и риска ее раскрытия необходимо объединить несколько методов.

Лучшая стратегия для конкретного приложения будет зависеть от его целей и рисков.

Проактивные методы

Проектирование реляционной базы данных основано на стратегии нормализации. Подавляющее число случаев ограничивается нормализацией только по функциональным зависимостям, приводящей к получению схемы базы данных в третьей нормальной форме или нормальной форме Бойса-Кода. Схема представляет из себя множество реляционных отношений, связанных ограничениями целостности, которые с математической точки зрения являются именно функциональными зависимостями. Поэтому исключение атак через логический вывод по функциональным зависимостям должно быть неизменной составляющей при проектировании баз данных, содержащих конфиденциальную информацию.

Тем не менее, в настоящее время нет устоявшейся, широко используемой методики учета атак логического вывода при конструировании схем баз данных. Попробуем разобраться, почему это происходит и, соответственно, предложить дальнейшие исследования в этом направлении.

В этом направлении одной из наиболее цитируемых работ является работа Tzong-An Su и Gultekin Ozsoyoglu «Controlling FD and MVD Inferences in Multilevel Relational Database Systems» [Управление выводами FD и MVD в многоуровневых системах реляционных баз данных]. Сразу отметим, что основным недостатком этой работы является то, что в ней схема базы данных представляется одним отношением. Т.е. рассматривается универсальное отношение — то чем база данных может быть представлена лишь до начала процедуры нормализации. Это очень далеко от реальной практики, чем видимо и объяс-

няется то, что результаты статьи никогда не были доведены до практического применения. Кроме того, в статье рассматриваются только схемы классификации по атрибутам и по записям.

Тем не менее, некоторые положения этой работы являются важными и должны быть приняты во внимание. В ней отмечается, что одной из важных нерешенных проблем является проблема логического вывода из-за ограничений целостности. Ограничения целостности, прежде всего ограничения ссылочной целостности, лежат и в основе любой современной разработки OLTP баз данных (процедур нормализации), так что борьба с логическим выводом через ограничения ссылочной целостности должна проводиться уже на этапе проектирования схемы базы данных. Тем не менее, в этом направлении есть еще много нерешенных вопросов.

Чтобы пояснить, приведём два простых примера из цитируемой статьи.

Пример 1. Пусть **R(SID, G_1, G_2,..., G_m, GPA)** схема отношения базы данных университета со схемой классификации по атрибутам. **SID** — это идентификационный номер студента, G_i **1<i<m** — оценка студента с **SID** в курсе G_i, а **GPA** — его совокупный средний балл. Понятно, что функциональная зависимость **(G_1G_2... G_m) → GPA** присутствует в базе данных. Если **SID** и G_i, **l<i<m** классифицируются как SECRET, а средний балл — как TOPSECRET, то пользователь u с уровнем допуска SECRET может вывести средний балл любого студента.

Пример 2: Пусть **R=(M, S, W)** будет схемой отношений в военной базе данных со схемой классификации на уровне кортежей, где **M**, **S**, **W** название миссии, название военных кораблей, участвующих в миссии, и оружие, используемое в миссии. Предположим, что все задействованные боевые корабли имеют одинаковое вооружение, а в миссии m_1 задействованы три боевых корабля и три типа оружия. Тогда в базе данных есть отношение **r**, показанное в таблице 5.

Таблица 5. Отношение R

Table 5. Ratio R

Уровень	M	S	W
1	m_1	s_1	w_1
2	m_1	s_1	w_2
1	m_1	s_1	w_3
3	m_1	s_2	w_1
4	m_1	s_2	w_2
5	m_1	s_2	w_3
4	m_1	s_3	w_1
5	m_1	s_3	w_2
6	m_1	s_3	w_3

Если пользователь **u** с уровнем допуска 4 знает тот факт, что каждый военный корабль имеет одинаковый набор вооружения, тогда он может сделать вывести кортежи (m_1, s_2, w_3), (m_1, s_3, w_2) и (m_1, s_3, w_3), которые не авторизованы для него.

Обе функциональные зависимости нарушают правила нормализации (в частности **GPA** — вычисляемый атрибут). При проведении нормализации эти зависимости будут переведены в ограничения ссылочной целостности (зависимости первичный ключ — внешний ключ). Таким образом, это будут зависимости между таблицами. Тем не менее, возможность несанкционированного логического вывода через них останется.

Приведенные выше примеры показывают, что, если ограничения целостности не отражаются должным образом, пользователи могут выводить неавторизованную информацию из многоуровневой реляционной базы данных.

Из анализа ясно, что защита баз данных от атак логического вывода должна продумываться и реализовываться уже на этапе разработки реляционной СУБД.

Единственным подходом к проектированию схемы реляционной базы данных, осуществляющей OLTP-обработку, является процедура нормализации. Но даже в случае нормализации только по функциональным зависимостям (без учета многозначных зависимостей и зависимостей проекции-соединения) она является только отчасти математически формализованной процедурой,

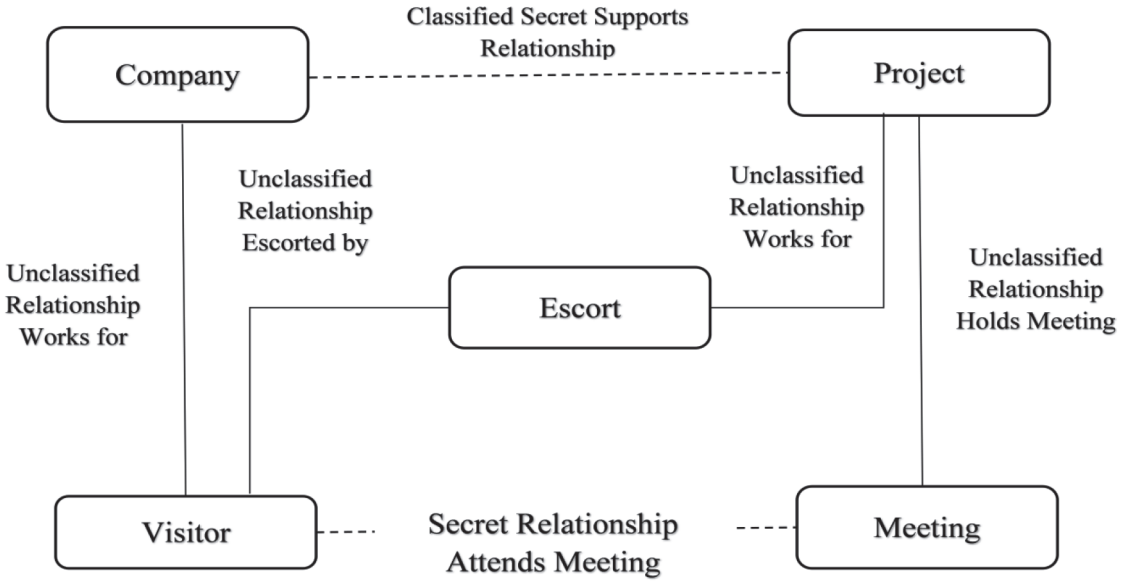
а в некотором смысле и искусством проектировщика. Например, проектировщик решит ограничиться третьей нормальной формой. Но тогда в некоторых таблицах останутся паразитные зависимости примитивных атрибутов от неключевых. Если будет принято решение нормализовать до нормальной формы Бойса-Кодда, то в таблицах останутся только зависимости от возможных ключей, но при этом проектировщику придется принимать решение от какого из двух свойств придется отказаться: сохранение информации или функциональных зависимостей. Любое из этих решений повлияет на наличие каналов нарушения безопасности через логический вывод.

Кроме того, при нормализации нормализуемые паразитные функциональные зависимости не исчезают — в процессе ее проведения они переводятся в ограничения целостности — зависимости между атрибутами разных таблиц, причем в специальный вид функциональных зависимостей, а именно в зависимости включения.

Здесь также возникают проблемы требующие решения, самой известной из которых является логический вывод, названный в [18,19] «выводом по второму пути». На рисунке приведен пример взятый из [20] (приведена схема авторов).

В БД предполагается, что для каждого проекта есть свой сопровождающий, который встречает посетителей, направленных для встречи по этому проекту. Нельзя раскрывать компании, участвующие в определенных конфиденциальных проектах. Поэтому прямая связь между проектом (project) и компанией, которая участвует в проекте (company), классифицируется в БД на уровне высокий (пунктирная линия на рисунке). Эту связь можно попытаться выявить, общаясь с БД на более низком уровне классификации (сплошные линии), найдя второй путь, который устанавливает связь между компанией и проектом.

Один второй путь использует доступные связи между проектом (project), встречей (meeting), посетителем (visitor) и компанией (company). project→meeting→visitor→company. Этот путь строит список участников собрания по проекту с указанием сотрудников компаний, участвующих в собрании, что позволяет связать компании, в которых участники



Логический вывод → Company → Project через Escort

Inference →Company → Project via Escort

работают, с проектами. Однако этот путь также не виден на низком уровне, т.к. связь между посетителем и собранием также классифицируется на уровне высокий (пунктирная линия между visitor и meeting). Таким образом, он не дает возможности получить конфиденциальную информацию.

Однако, есть другой низкий второй путь, состоящий из проекта (project), сопровождающего (escort), посетителя (visitor), компании (company). project→escort→visitor→company, который получает конфиденциальную информацию.

Наличие вывода по второму пути возникает при проектировании базы данных и должно учитываться. Схема должна быть доработана или переработана.

Заключение

Работа преследовала два цели. Первая — показать, что проактивные методы защиты баз данных не менее важны чем реактивные. По мнению автора, даже более важны. Если реактивные методы используют достаточно сложные алгоритмы и ресурсозатратные технические архитектуры, не гарантируя при этом защиту от атаки, то проактивные методы правильно используемые при разработке базы данных могут обеспечить надежную защиту данных от атак логического вывода (если конечно злоумышленник не обладает знаниями о данных, выходящими за рамки тех, которые были использованы в ее разработке).

И вторая цель работы. Несмотря на наличие определенной совокупности работ по проективным методам их результаты не используются в повседневной практике. Причина состоит в том, что они не интегрированы в практическую деятельность IT-специалистов. Задача интеграции не проста и требует дополнительных исследований. Автор надеется, что усилия по ее решению будут предприняты не только им, но и другими исследователями.

СПИСОК ИСТОЧНИКОВ

1. **Poltavtseva M.A.** Evolution of data management systems and their security // в сборнике: Proceedings — 2019 International Conference on Engineering Technologies and Computer Science: Innovation and Application, EnT 2019. 2019. С. 25–29.

2. **Aleksandrova E. B**. Applying the group signature for entity authentication in distributed grid computing networks / E. B. Aleksandrova, D. P. Zegzhda, A. S. Konoplev // Automatic Control and Computer Sciences.— 2016.— Vol. 50.— No 8.— P. 739–742.— DOI 10.3103/S0146411616080265.

3. **Александрова Е. Б**. Применение постквантовой и гомоморфной криптографии в задачах кибербезопасности / Е. Б. Александрова, Н. Н. Шенец // Неделя науки СПбПУ: Материалы научного форума с международным участием. Междисциплинарные секции и пленарные заседания институтов, Санкт-Петербург, 30 ноября — 052015 года.— Санкт-Петербург: Федеральное государственное автономное образовательное учреждение высшего образования «Санкт-Петербургский политехнический университет Петра Великого», 2015.— С. 9–17.

4. **Aleksandrova E. B**. Post-Quantum Primitives in Information Security / E. B. Aleksandrova, A. A. Shtyrkina, A. V. Iarmak // Nonlinear Phenomena in Complex Systems.— 2019.— Vol. 22.— No 3.— P. 269–276.

5. **Полтавцева М.А**., **Ожиганова М. И., Егорова А. О., Костюков А. Д., Миронова А. О.** Модели форензики и расследование инцидентов в СУБД // Защита информации. Инсайд. 2021. № 3 (99). С. 18–23.

6. **Полтавцева М.А**., **Зегжда Д. П., Калинин М. О.** Модель угроз безопасностисистем управления большими данными // Проблемы информационной безопасности. Компьютерные системы. 2019. № 2. С. 16–28.

7. **Полтавцева М.А**., **Калинин М. О.** Защита данных в системах мониторинга безопасности крупномасштабных объектов в сборнике: Управление развитием крупномасштабных систем (MLSD'2019). Материалы двенадцатой международной конференции. Под общей редакцией С. Н. Васильева, А. Д. Цвиркуна. 2019. С. 1019–1029.

8. **Полтавцева М.А**. Консистентный подход к построению защищенных систем обработки

и хранения больших данных // Проблемы информационной безопасности. Компьютерные системы. 2019. № 2. С. 29–44.

9. **Полтавцева М.А.**, **Торгов В. А.** Применение технологий распределенного реестра для аудита и расследования инцидентов в системах обработки больших данных // Проблемы информационной безопасности. Компьютерные системы. 2021. № 4 (48). С. 144–156.

10. **Zegzhda D.P.**, **Moskvin D. A., Myasnikov A. V.** Assurance of cyber resistance of the distributed data storage systems using the blockchain technology // Automatic Control and Computer Sciences. 2018. Vol. 52. No. 8. P. 1111–1116.

11. **Usov E.S.**, **Nikol'skii A.V., Pavlenko E. Y., Zegzhda D. P.** Architecture of the protected cloud data storage using intel sgx technology // Automatic Control and Computer Sciences. 2018. Vol. 52. No. 8. P. 1144–1149.

12. **Josep Domingo-Ferrer and Vicenç Torra**, "Disclosure control methods and information loss for microdata". Confidentiality, Disclosure and Data Access: Theory and Practical Applications for Statistical Agencies, pages 91–110. 2001. ISBN: 0-444-50761-2.

13. **Josep Domingo-Ferrer**, "Microaggregation for protecting individual data privacy". Proceso de Toma de Decisiones, Modelado y Agregación de Preferencias TIC-2002–11492-E, pages 171–178. 2005. ISBN: 84-934654-1-0.

14. **Adam, Nabil R.; Wortmann, John C.;** Security-Control Methods for Statistical Databases: A Comparative Study; ACM Computing Surveys, Pages: 515–556, Vol. 21, No. 4, December 1989, ISSN: 0360-0300.

15. **Dorothy E**. Denning and Jan Schlorer. Inference Control for Statistical databases. IEEE Journal, Vol. 16, No. 7: pp 69–82, July 1983.

16. **Josep Domingo-Ferrer and Josep M. Mateo-Sanz**, "On the security of cell suppression in contingency tables with quantitative factors". Proceedings of the 3rd International Seminar on Statistical Confidentiality, pages 208–217. 1996. ISBN: 86-81141-41-4.

17. **Josep Domingo-Ferrer, Anna Oganian and Vicenç Torra**, "Information-theoretic disclosure risk measures in statistical disclosure control of tabular data". Proceedings of the 14th Intl. Conf. on Scientific and Statistical DataBase Management, pages 227–231. 2002. ISBN: 0-7695-1632-7.

18. **Thomas D. Garvey**; The Inference Problem for Computer Security. IEEE, (1): pp 78–81, Computer Security Foundations Workshop V, 1992. Proceedings, ISBN: 0-8186-2850-2.

19. **Rajagopal Chakravarthi, Glenn Shafer, S. Raman**. Inference Detection Using Clustering. PHD Thesis, 13 December 2001.

20. **Thomas H. Hinke, Harry S. Delugach and Randall F. Wolf**. Protecting databases from inference attacks. Computers & Security Vol. 16, No. 8, pp. 687–708, 1997.

21. **Hoffman, L.J., and Miller, W.F.** "Getting a personal dossier from a statistical data bank," Datamation 16, 5 (May 1970), 74–75

22. **Schlorer, J.** "Confidentiality and security in statistical data banks" statistical data bank," Proc. Workshop on Data Documentation, Verlag Documentation 1977, pp. 101–123

REFERENCES

1. **Poltavtseva M.A.** Evolution of data management systems and their security // в сборнике: Proceedings — 2019 International Conference on Engineering Technologies and Computer Science: Innovation and Application, EnT 2019. 2019. P. 25–29.

2. **Aleksandrova E. B**. Applying the group signature for entity authentication in distributed grid computing networks / E. B. Aleksandrova, D. P. Zegzhda, A. S. Konoplev // Automatic Control and Computer Sciences.— 2016.— Vol. 50.— No 8.— P. 739–742. DOI 10.3103/S0146411616080265.

3. **Aleksandrova E. B**. Primenenie postkvantovoj i gomomorfnoj kriptografii v zadachah kiberbezopasnosti [Application of post-quantum and homomorphic cryptography in cybersecurity problems] / E. B. Aleksandrova, N.N. SHenec // Nedelya nauki SPbPU: Materialy nauchnogo foruma s mezhdunarodnym uchastiem. Mezhdisciplinarnye sekcii i plenarnye zasedaniya institutov, Sankt-Peterburg, 30 noyabrya — 052015 goda.— Sankt-Peterburg: Federal'noe gosudarstvennoe avtonomnoe obrazovatel'noe uchrezhdenie vysshego obrazovaniya "Sankt-Peterburgskij politekhnicheskij universitet Petra Velikogo", 2015.— S. 9–17. (In Russian)

4. **Aleksandrova E. B**. Post-Quantum Primitives in Information Security / E. B. Aleksandrova, A. A. Shtyrkina, A. V. Iarmak // Nonlinear Phenomena in Complex Systems.— 2019.— Vol. 22.— No 3.— P. 269–276.

5. **Poltavtseva M.A., Ozhiganova M. I., Egorova A. O., Kostyukov A. D., Mironova A. O.** Modeli forenziki i rassledovanie incidentov v subd [forensic models and dbms incident investigation] // Zashchita informacii. Insajd. 2021. No 3 (99). P. 18–23. (In Russian)

6. **Poltavceva M.A., Zegzhda D. P., Kalinin M. O.** Model' ugroz bezopasnostisistem upravleniya bol'shimi dannymi [security threats model for big data management systems] // Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy. 2019. No 2. P. 16–28. (In Russian)

7. **Poltavceva M.A., Kalinin M. O.** Zashchita dannyh v sistemah monitoringa bezopasnosti krupnomasshtabnyh ob"ektov [data protection in security monitoring systems for large-scale facilities] // v sbornike: Upravlenie razvitiem krupnomasshtabnyh sistem (MLSD'2019). Materialy dvenadcatoj mezhdunarodnoj konferencii. Pod obshchej redakciej S. N. Vasil'eva, A. D. Cvirkuna. 2019. P. 1019–1029. (In Russian)

8. **Poltavceva M.A.** Konsistentnyj podhod k postroeniyu zashchishchennyh sistem obrabotki i hraneniya bol'shih dannyh [consistent approach to building secure systems for processing and storing big data] // Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy. 2019. No 2. P. 29–44. (In Russian)

9. **Poltavceva M.A., Torgov V. A.** Primenenie tekhnologij raspredelennogo reestra dlya audita i rassledovaniya incidentov v sistemah obrabotki bol'shih dannyh [application of distributed register technologies for audit and investigation of incidents in big data processing systems] // Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy. 2021. No 4 (48). P. 144–156. (In Russian)

10. **Zegzhda D.P., Moskvin D. A., Myasnikov A. V.** Assurance of cyber resistance of the distributed data storage systems using the blockchain technology // Automatic Control and Computer Sciences. 2018. Vol. 52. No 8. P. 1111–1116.

11. **Usov E.S., Nikol'skii A.V., Pavlenko E. Y., Zegzhda D. P.** Architecture of the protected cloud data storage using intel sgx technology // Automatic Control and Computer Sciences. 2018. Vol. 52. No 8. P. 1144–1149.

12. **Josep Domingo-Ferrer and Vicenç Torra**, "Disclosure control methods and information loss for microdata". Confidentiality, Disclosure and Data Access: Theory and Practical Applications for Statistical Agencies, pages 91–110. 2001. ISBN: 0-444-50761-2.

13. **Josep Domingo-Ferrer**, "Microaggregation for protecting individual data privacy". Proceso de Toma de Decisiones, Modelado y Agregación de Preferencias TIC-2002–11492-E, pages 171–178. 2005. ISBN: 84-934654-1-0.

14. **Adam, Nabil R.; Wortmann, John C.** Security-Control Methods for Statistical Databases: A Comparative Study; ACM Computing Surveys, Pages: 515–556, Vol. 21, No. 4, December 1989, ISSN: 0360-0300.

15. **Dorothy E**. Denning and Jan Schlorer. Inference Control for Statistical databases. IEEE Journal, Vol. 16, No. 7: pp 69–82, July 1983.

16. **Josep Domingo-Ferrer and Josep M. Mateo-Sanz**, "On the security of cell suppression in contingency tables with quantitative factors". Proceedings of the 3rd International Seminar on Statistical Confidentiality, pages 208–217. 1996. ISBN: 86-81141-41-4.

17. **Josep Domingo-Ferrer, Anna Oganian and Vicenç Torra**, "Information-theoretic disclosure risk measures in statistical disclosure control of tabular data". Proceedings of the 14th Intl. Conf. on Scientific and Statistical DataBase Management, pages 227–231. 2002. ISBN: 0-7695-1632-7.

18. **Thomas D. Garvey**; The Inference Problem for Computer Security. IEEE, (1): pp 78–81, Computer Security Foundations Workshop V, 1992. Proceedings, ISBN: 0-8186-2850-2.

19. **Rajagopal Chakravarthi, Glenn Shafer, S. Raman**. Inference Detection Using Clustering. PHD Thesis, 13 December 2001.

20. **Thomas H. Hinke, Harry S. Delugach and Randall F. Wolf**. Protecting databases from inference attacks. Computers & Security Vol. 16, No. 8, pp. 687–708, 1997.

21. **Hoffman, L.J., and Miller, W.F.** "Getting a personal dossier from a statistical data bank," Datamation 16, 5 (May 1970), 74–75

22. **Schlorer, J.** "Confidentiality and security in statistical data banks" statistical data bank," Proc. Workshop on Data Documentation, Verlag Documentation 1977, pp. 101–123

Статья поступила в редакцию 05.02.2022; одобрена после рецензирования 14.02.2022; принята к публикации 20.02.2022.

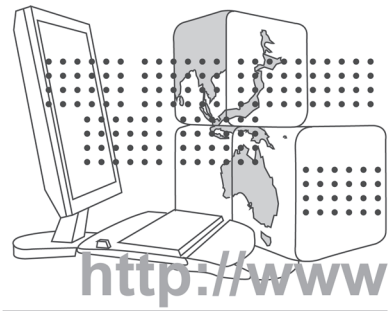
The article was submitted 05.02.2022; approved after reviewing 14.02.2022; accepted for publication 20.02.2022.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

ПОЛТАВЦЕВ Анатолий Алексеевич — к.т.н., доцент; Тверской государственный технический университет

POLTAVTSEV Anatolij Al.— Ph.D., Associate Professor; Tver State Technical University

ORCID 0000-0003-2003-7014



БЕЗОПАСНОСТЬ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Научная статья
DOI 10.48612/jisp/3d3r-etdx-nfu9
УДК 004.056

А. А. Криулин, М. А. Еремеев, Г. Ю. Потерпеев

ФГБОУ ВО «МИРЭА — Российский технологический университет»,
119454, Российская Федерация, г. Москва

ПОДХОД К АНАЛИЗУ ПРОГРАММНЫХ СРЕДСТВ ХАКЕРСКИХ ГРУППИРОВОК С ИСПОЛЬЗОВАНИЕМ БАЗЫ ЗНАНИЙ MITRE AT&T

Аннотация. В статье рассматриваются вопросы возможности применения базы знаний Mitre AT&T при разработке систем обнаружения компьютерных атак с участием вредоносных программ. С использованием API Mitre проводится статистический анализ вредоносных программ, а также техник и тактик, применяемых хакерскими группировками для составления дополнительных индикаторов компрометаций атак.

Ключевые слова: компьютерная безопасность, Mitre, APT, вредоносные программы, исполняемый файл, статистический анализ.

Research article
DOI 10.48612/jisp/3d3r-etdx-nfu9

A. A. Kriulin, M. A. Eremeev, G. Yu. Poterpeev

Federal State Budget Educational Institution of Higher Education «MIREA — Russian Technological University» Russia, Moscow

ANALYSIS OF HACKER GROUPS SOFTWARE TOOLS USING AT&T'S MITRE KNOWLEDGE BASE

Abstract. The article discusses the possibility of using the Mitre AT&T knowledge base in the development of systems for detecting computer attacks involving malware. Using the Mitre API, a statistical analysis of malware is carried out, as well as techniques and tactics used by hacker groups to compile additional indicators of compromised attacks.

Keywords: computer security, Mitre, APT, malware, executable file, statistical analysis.

В последнее время увеличивается не только количество кибератак в информационном пространстве, но и растет наносимый ущерб (экономический, политический, технологический и т.д.) от компьютерных инцидентов, которые все больше принимают целевую форму. Термин APT (Advanced Persistent Threat) или «развитая устойчивая угроза», который изначально использовался для описания компьютерных атак на военные организации, более не ограничен и харак-

теризуется длительной подготовкой к взлому конкретной цели, сопровождаемому эксплуатацией уязвимостей, социальной инженерией, подготовкой инфраструктуры и т.д. Одновременно с ростом целевых компьютерных атак увеличивается число атрибутов, их в той или иной степени характеризующих [1].

В рамках статьи предлагается использовать средства базы знаний Mitre AT&T для исследования тактик и техник различных APT-группировок с целью выявления дополнительных индикаторов компрометаций.

В настоящее время база Mitre AT&T содержит записи о 113 хакерских группировках (ХГ), 13 тактиках, более 240 техниках и подтехниках и свыше 500 программных средствах (рис. 1).

Тактики в базе Mitre AT&T представляют собой основные этапы компьютерной атаки, такие как:

– разведка (Reconnaissance), включает 10 техник;

– подготовка ресурсов (Resource Development), включает 7 техник;

– получение доступа (Initial Access), включает 9 техник;

– исполнение кода (Execution), включает 12 техник;

– закрепление (Persistence), включает 19 техник;

– повышение привилегий (Privilege Escalation), включает 13 техник;

– обход защиты (Defense Evasion), включает 39 техник;

– сбор паролей (Credential Access), включает 15 техник;

– исследование (Discovery), включает 27 техник;

– горизонтальное перемещение (Lateral Movement), включает 9 техник;

– сбор данных (Collection), включает 17 техник;

– управление и контроль (Command and Control), включает 16 техник;

– вывод данных (Exfiltration), включает 9 техник;

– воздействие (Impact), включает 13 техник.

Каждая тактика может быть реализована различными техниками, которые в свою очередь могут быть либо едиными, либо состоять из подтехник. Например, техника фишинг (Phishing) относится к тактике получения доступа (Initial Access), и может быть реализована тремя способами: целевые фишинговые вложения (Spearphishing Attachment), целевые фишинговые ссылки (Spearphishing Link) и целевые фишинговые сервисы (Spearphishing Service).

Компания Mitre AT&T предоставляет прикладной программный интерфейс (API) к своей базе знаний. С использованием Mitre API можно получить различные структурированные данные, например, обо всех техниках, которые использует та или иная хакерская группировка. Для описания данных Mitre использует формат STIX (Structured Threat Information Expression), представляющий собой информационную модель для сериализации и анализа данных о киберугрозах (Cyber Threat Intelligence, CTI) [2]. Объекты в модели STIX классифицируются по следующим атрибутам:

– шаблон атаки (Attack Pattern) — описывает способы, которыми злоумышленники проводят атаку;

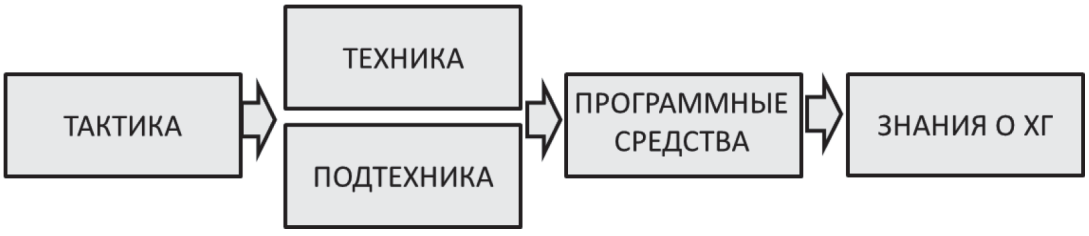


Рис. 1. Представление знаний в базе Mitre AT&T

Fig. 1. Knowledge representation in the AT&T Miter database

– кампания (Campaign) — набор вредоносных действий или атак, которые происходят в течение определенного периода времени в отношении атакуемых целей;

– курс действий (Course of Action) — действие, предпринимаемое для предотвращения атаки или реагирования на нее;

– индикатор (Indicator) — шаблон, который можно использовать для обнаружения подозрительной или злонамеренной кибер-активности;

– набор вторжений (Intrusion Set) — сгруппированный набор техник и ресурсов с общими свойствами, которые используются для проведения атаки;

– вредоносное программное обеспечение (Malware) — программы, используемые для нарушения конфиденциальности, целостности или доступности данных в атакуемой системе;

– наблюдаемые данные (Observed Data) — информация, наблюдаемая в системе или сети (например, IP-адрес);

– инструменты (Tool) — легитимное программное обеспечение, которое может использоваться в рамках атаки;

– уязвимости (Vulnerability) — ошибки в программном обеспечении, непосредственно используемые для получения доступа к системе или сети.

С использованием прикладного API можно выделить техники, которые наиболее часто применяют различные хакерские группировки (рис. 2).

Из диаграммы, представленной на рисунке 2 видно, что больше половины всех хакерских группировок используют целевые фишинговые вложения (Spearphishing Attachment) и вредоносные файлы (Malicious File) при получении доступа.

Для закрепления хакерские группировки предпочитают использовать ключи реестра автозагрузки (Registry Run Keys / Startup Folder), например, Run. Также APT-группы обфусцируют файлы и данные (Obfuscated Files or Information) при обходе средств защиты, используют PowerShell для исполнения сценариев в системе и применяют легитимные программные средства (Ingress Tool Transfer) передачи данных.

Целесообразно проводить анализ групп, которые используют наибольшее число различных техник или наоборот (рис. 3). Значительное число техник, используемых группой APT32, позволит более детально отображать ее цифровой профиль. На каждом этапе компьютерной атаки хакеры из группы APT32 используют самые различные техники и программные средства, что дает

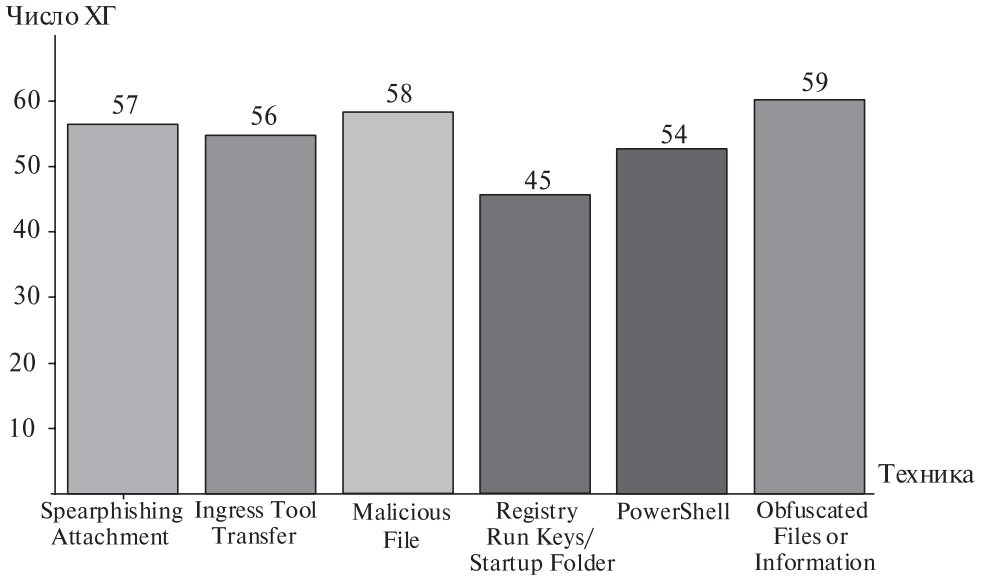


Рис. 2. Техники, которые используют большинство ХГ

Fig. 2. Techniques used by the majority of hacker groups

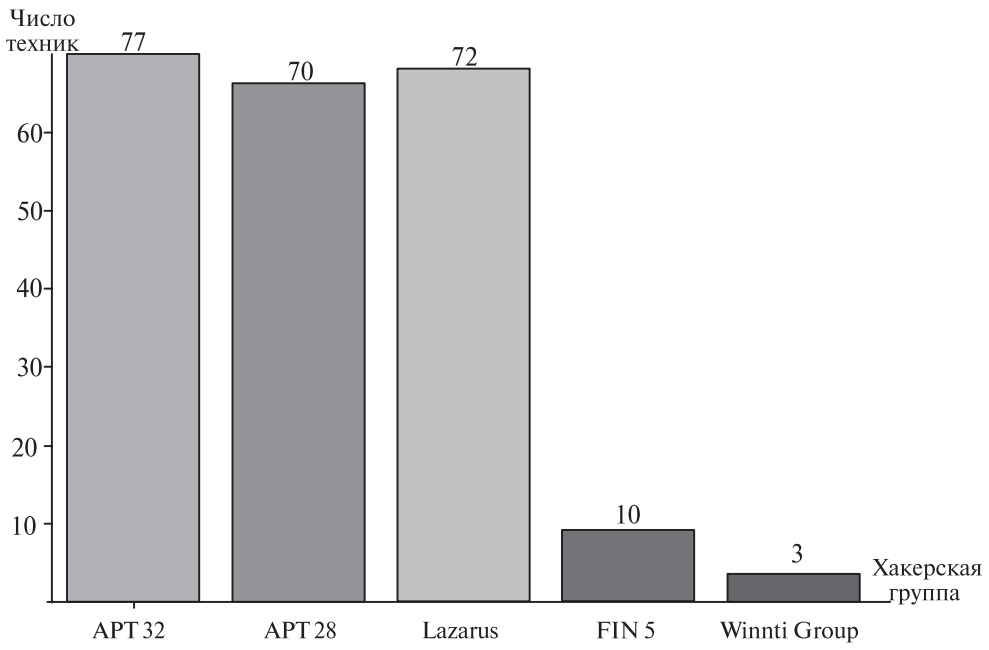


Рис. 3. Статистика по числу используемых техник

Fig. 3. Statistics on the number of techniques used

большой выбор для их идентификации. Напротив, малое число техник и программных средств, используемых группой Winnti, говорит об её уникальности среди других, что также может быть использовано при разработке средств защиты.

Хакерские группировки используют программные средства (ПС) различных форматов, охватывая большинство современных операционных систем.

Наиболее распространенным из них является формат исполняемых файлов Portable Executable (PE), предназначенный для работы в ОС Windows. Формат PE является разработкой корпорации Microsoft и согласно спецификации [3] содержит описание основных заголовков и секций исполняемых и объектных файлов. Поля заголовков, секций и других служебных структур представлены количественными и категориальными значениями, характеризуя отдельный исполняемый файл среди множества других. В частности, в исследовании [4], установлено, что вредоносные и безвредные исполняемые файлы могут значительно отличаться по структуре и содержимому. В рамках основной концепции исследования полагается, что структура и содержимое программных средств различных хакерских группировок являются уникальными атрибутами, анализ которых необходимо провести.

К характеристикам формата файла программных средств хакерских групп, которые можно исследовать как уникальные атрибуты, относятся:

– размер файлового заголовка;
– размер опционального заголовка;
– виртуальный размер;
– размер исполняемого кода;
– наличие контрольной суммы;
– файловый размер;
– виртуальный размер секций;
– файловый размер секций;
– число импортируемых функций;
– время и дата компиляции;
– размер инициализированных данных и другие.

Обработка исполняемых файлов и сбор статистических данных были выполнены с использованием разработанного анализатора [5].

На графиках, изображенных на рисунке 4, отражено распределение числа импортируемых функций в программных средствах различных хакерских группировках. Из анализа этого графика можно сделать ряд выводов, например, большинство программных средств группировки Indrik Spider содержат меньше 50-ти импортируемых функций, что говорит либо об использовании «упаковщиков», либо о не значительных функциональных возможностях программ этой ХГ.

На графиках, изображенных на рисунке 5, отражено распределение значений байтовой энтропии в программных средствах различных хакерских группировок. Большинству программных средств ХГ Indrik Spider характерны высокие значе-

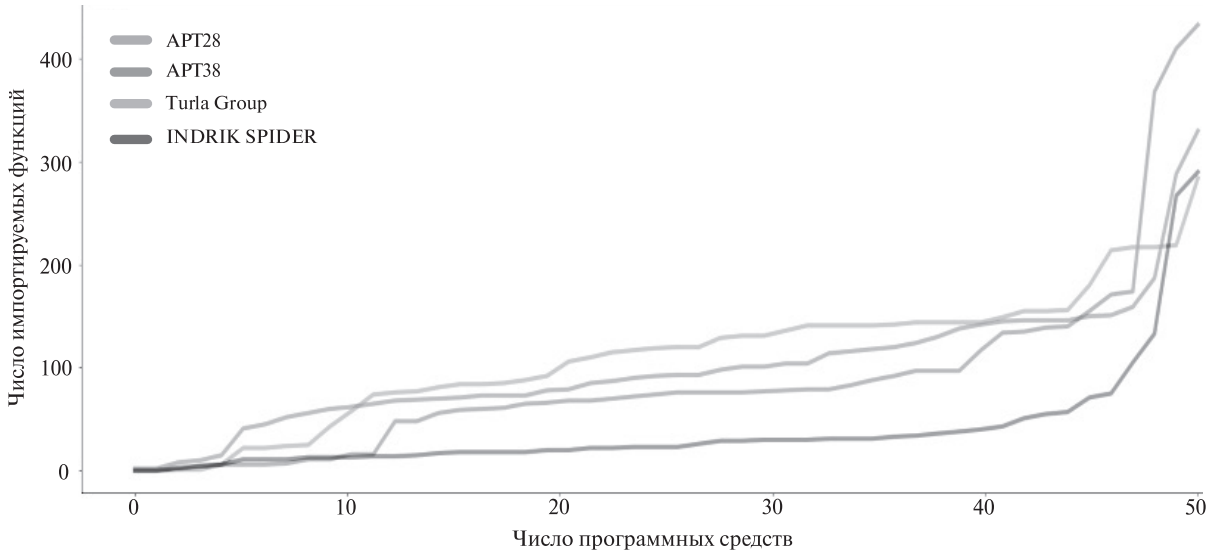


Рис. 4. Распределение импортируемых функций в ПС различных ХГ

Fig. 4. Distribution of imported functions in PS of various hacker groups

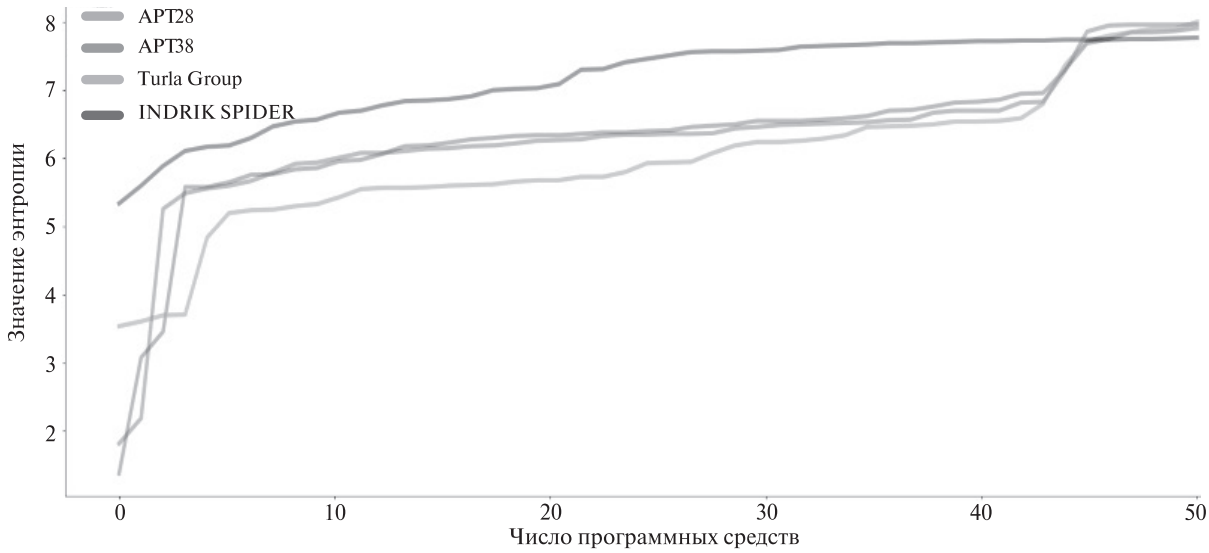


Рис. 5. Распределение значений байтовой энтропии в ПС различных ХГ

Fig. 5. Distribution of byte entropy values in PS of various hacker groups

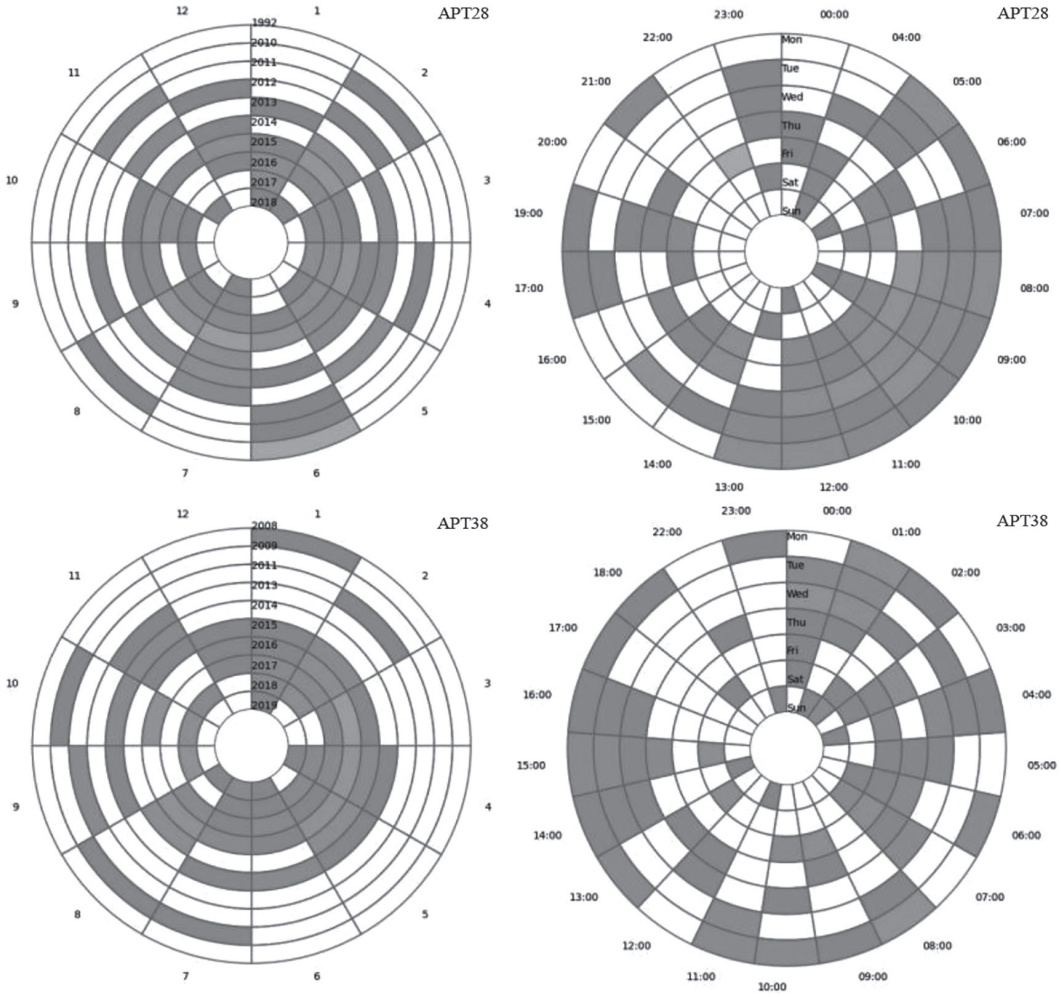


Рис. 6. Представление даты и времени компиляции программ различных ХГ

Fig. 6. Date and time presentation of programs compilation of various hacker groups

ния этой характеристики, что в совокупности с предыдущими графиками (рис. 4) дополняет предположение об использовании «упаковщиков».

Из анализа отечественных и зарубежных отчетов о компьютерных инцидентах, связанных с атаками на информационно-телекоммуникационные системы объектов различного сектора (государственного, промышленного, информационного и т.д.) в качестве уникального атрибута выделяется временная и суточная активности хакерских группировок.

В частности, суточная активность может характеризовать рабочий календарь группировки с учетом выходных и праздничных дней конкретного региона. Временная активность отражает распределение рабочего времени в течение суток. Суточную и временную активности хакерских группировок наглядно можно представить в виде специфического графика «временное колесо» (рис. 6).

На рисунке 6 с помощью графика «временное колесо» представлено сравнение даты и времени компиляции программных средств хакерских группировок APT28 и APT38. Можно заметить, что для APT28 более характерна сборка программ в период с 07:00 до 13:00 в будни, что соответствует привычной всем рабочей неделе. Компиляция файлов APT38 в основном выполняется во второй половине дня с 13:00 до 18:00.

Для графического отображения некоторых статистических показателей различных атрибутов программных средств можно использовать диаграмму размаха, называемую также «ящик с усами». Такой вид диаграммы в удобной форме позволяет представить медиану, максимальное и минимальное значения атрибута, нижний и верхний квартили, а также выбросы. На рисунке 7 представлена диаграмма размаха размеров программных средств различных хакерских группировок. Из рисунка 7 видно, что размер программных средств может использовать как уникальный атрибут хакерской группировки. В частности, программы группировки Wizard Spider значительно отличаются от программных средств APT1 по размеру.

Разработчики Mitre AT&T также разработали фреймворк, позволяющий в графическом виде сравнить тактики и техники, нескольких хакерских группировок (рис. 8).

На рисунке 8 представлено сравнение тактик и техник хакерской группировки APT28 и FIN7. Зеленым цветом выделены техники, которые использует группа APT28, желтым цветом выделены техники, используемые группой FIN7, а красным цветом выделяются техники, которые используют обе хакерские группы. Как правило, техники выделенные красным цветом являются наиболее популярными и зачастую могут использоваться большинством хакерских группировок.

Заключение

В рамках статьи проведен анализ программных средств, а также методов, которые применяют хакерские группировки с использованием средств базы знаний Mitre AT&T. Результаты анализа позволяют сделать выводы о структурно-параметрических отличиях программных средств различных хакерских группировок,

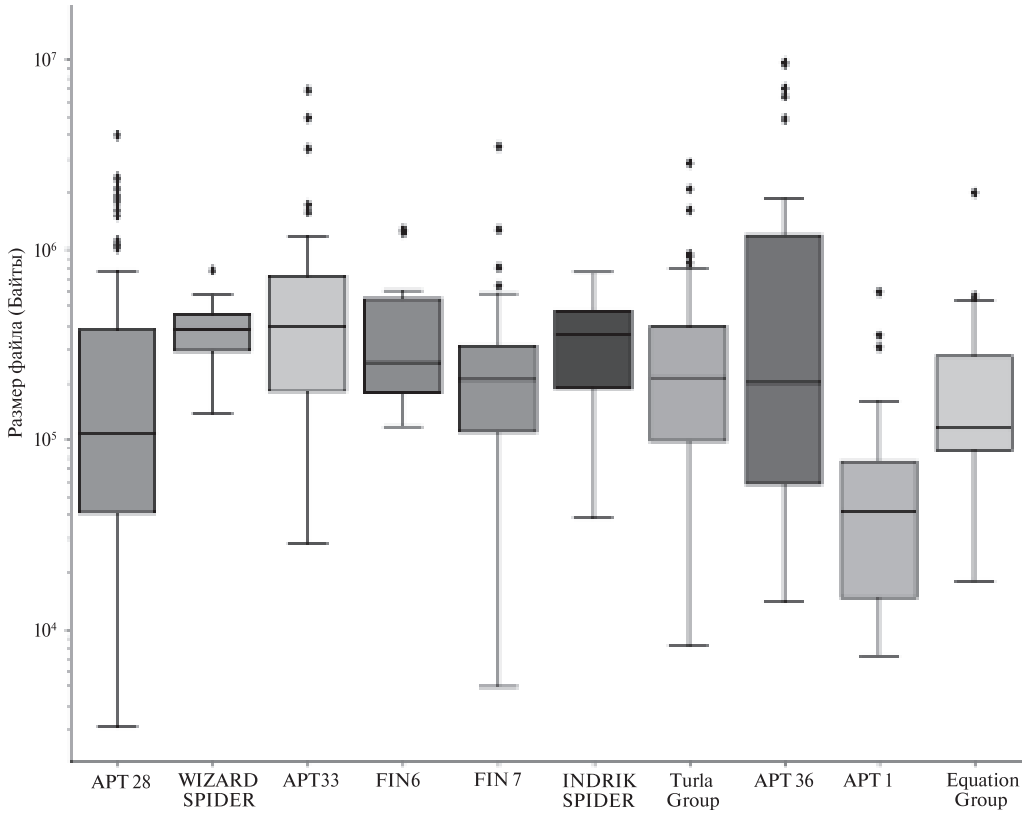


Рис. 7. Диаграммы размаха размеров программ различных ХГ

Fig. 7. Box-and-whiskers diagrams of program sizes for various hacker groups

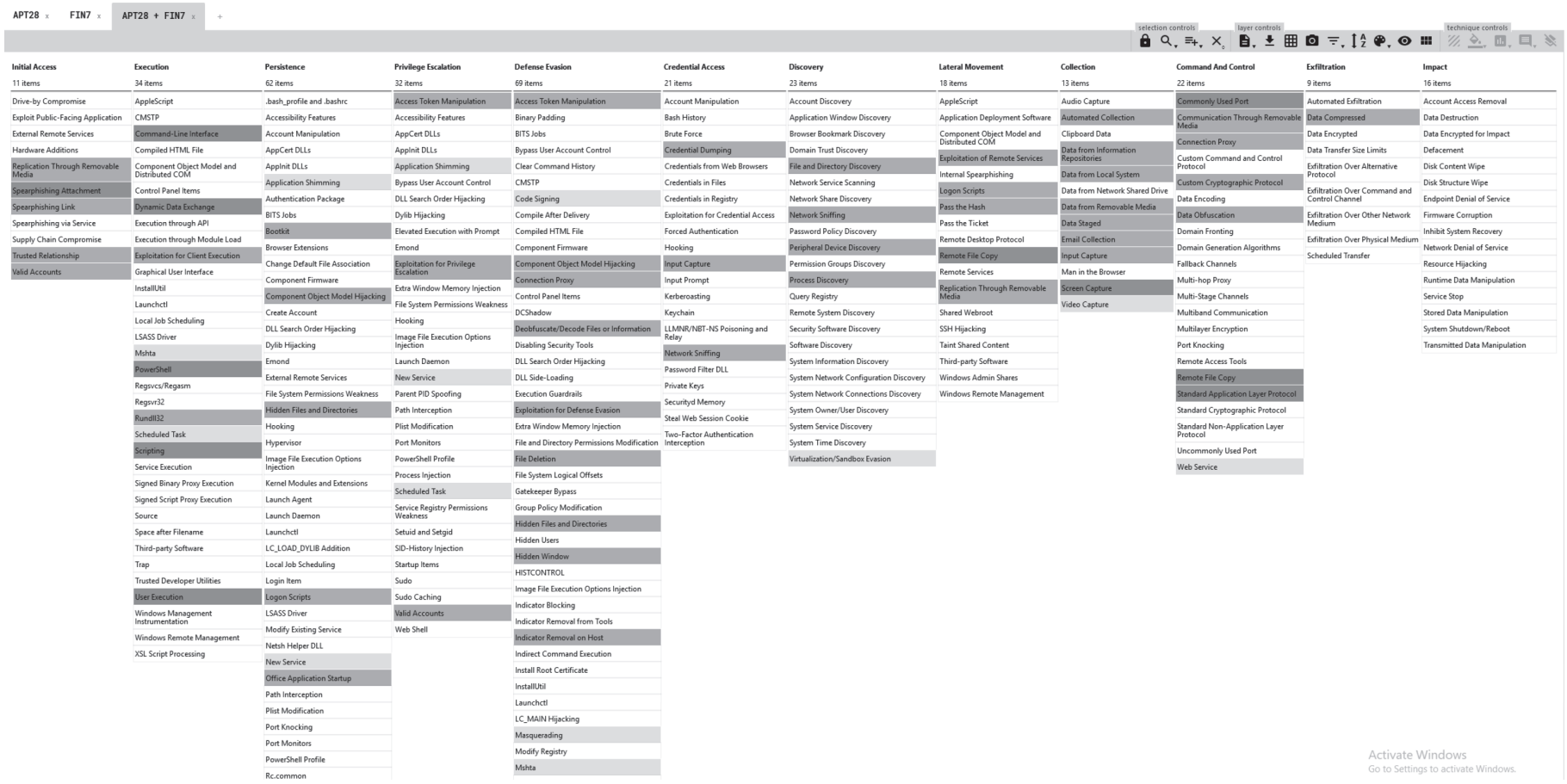


Рис. 8. Сравнение тактик и техник хакерских групп с использованием Mitre AT&T

Fig. 8. Comparison of hacker group tactics and techniques using Mitre AT&T

а также методов их работы. Проведенный анализ позволяет выделить статистические особенности и уникальные индикаторы компрометации APT-групп, которые могут быть использованы при разработке автоматизированной системы обнаружения хакерских атак в режиме реального времени.

СПИСОК ИСТОЧНИКОВ

1. **Нефедов В. С., Криулин А. А., Еремеев М. А.** Подход к выявлению вредоносных серверов сети анонимизации TOR на основе методов кластерного анализа / Проблемы информационной безопасности. Компьютерные системы. 2021. № 1 (45). С. 55–61.

2. Спецификация формата STIX [Электронный ресурс] — Режим доступа: https://docs.oasis-open.org/cti/stix/v2.1/cs01/stix-v2.1-cs01.html

3. Microsoft Portable Executable and Common Object File Format Specification [Электронный ресурс] // Microsoft Corp.— 1999.— Режим доступа: https://msdn.microsoft.com/en-us/library/ms809762.aspx

4. **Абашева И. В., Еремеев М. А., Криулин А. А., Нефедов В. С., Потерпеев Г. Ю.** Применение методов машинного обучения к задачам обнаружения вредоносного программного обеспечения / Труды Военно-космической академии имени А. Ф. Можайского. 2020 № 675 С. 164–171.

5. **Криулин А. А.** Анализатор исполняемых файлов формата PE / Свидетельство о государственной регистрации программы для ЭВМ № 2019614655 / правообладатель Криулин А. А.; заявл. 01.03.2019; зарег. 10.04.2019, Бюл. № 4.

REFERENCES

1. **Nefedov V. S., Kriulin A. A., Eremeev M. A.** Podhod k vyyavleniyu vredonosnyh serverov seti anonimizacii TOR na osnove metodov klasternogo analiza [Approach to Detect Maliscous Servers on Anonimous Network TOR Based on Clasterization Methods]/ Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy. 2021. № 1 (45). S. 55–61. (In Russian)

2. Specifikaciya formata STIX [STIX Format Specification].

3. Microsoft Portable Executable and Common Object File Format Specification // Microsoft Corp.— 1999.

4. **Abasheva I. V., Eremeev M. A., Kriulin A. A., Nefedov V. S., Poterpeev G. Yu.** Primenenie metodov mashinnogo obucheniya k zadacham obnaruzheniya vredonosnogo programmnogo obespecheniya [Applying Machine Learning Methods to Malware Detection Problems] / Trudy Voenno-kosmicheskoj akademii imeni A. F. Mozhajskogo. 2020 № 675 S. 164–171. (In Russian)

5. **Kriulin A. A.** Analizator ispolnyaemyh fajlov formata PE [PE format executable file analyzer] / Svidetel'stvo o gosudarstvennoj registracii programmy dlya EVM № 2019614655 / pravoobladatel' Kriulin A. A.; zayavl. 01.03.2019; zareg. 10.04.2019, Byul. № 4. (In Russian)

Статья поступила в редакцию 03.02.2022; одобрена после рецензирования 16.02.2022; принята к публикации 11.03.2022.

The article was submitted 03.02.2022; approved after reviewing 16.02.2022; accepted for publication 11.03.2022.

СВЕДЕНИЯ ОБ АВТОРАХ/ THE AUTHORS

КРИУЛИН Артур Андреевич — к.т.н., доцент; Российский технологический университет –МИРЭА

ЕРЕМЕЕВ Михаил Алексеевич — д.т.н., профессор; Российский технологический университет –МИРЭА

ПОТЕРПЕЕВ Герман Юрьевич — к.т.н., доцент; Российский технологический университет –МИРЭА

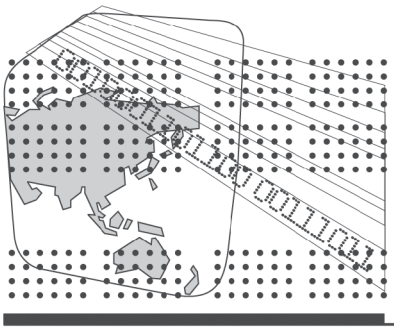
KRIULIN Artur A.— Associate Professor, Russian Technological University — MIREA

E-mail: kriulin@mirea.ru

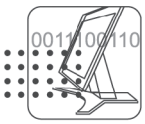
EREMEEV Mihail A.— Professor, Russian Technological University — MIREA

ORCID: 0000-0002-5511-4000

POTERPEEV German Yu.— Associate Professor, Russian Technological University — MIREA



БЕЗОПАСНОСТЬ РАСПРЕДЕЛЕННЫХ СИСТЕМ И ТЕЛЕКОММУНИКАЦИЙ



Научная статья
DOI 10.48612/jisp/mp53-t3hn-1pt3
УДК 004.056

А. Д. Фатин, Е. Ю. Павленко

Санкт-Петербургский политехнический университет Петра Великого (СПбПУ)
Россия, 195251, Санкт-Петербург, ул. Политехническая, д. 29

ИММУНИЗАЦИЯ СЛОЖНЫХ СЕТЕЙ: ТОПОЛОГИЯ И МЕТОДЫ

Аннотация. В работе рассматриваются основные способы иммунизации современных компьютерных сетей. Наибольшее внимание уделяется топологиям и видам рассматриваемых графовых структур. Также проводится оценка и сравнение существующих стратегий иммунизации и способов решения задач оптимального выбора узлов для иммунизации в контексте рассматриваемых топологий. Выделяются основные преимущества, недостатки и области применения выбранных топологий и методов решения задач иммунизации.

Ключевые слова: компьютерные сети, иммунизация, киберфизические системы, сетевая безопасность, безмасштабные сети.

Финансирование: Исследование выполнено в рамках гранта Президента РФ для государственной поддержки молодых российских ученых – кандидатов наук МК-3861.2022.1.6.

Research article
DOI 10.48612/jisp/mp53-t3hn-1pt3

A. D. Fatin, E. Yu. Pavlenko

Peter the Great St. Petersburg Polytechnic University, Russia, St. Petersburg

IMMUNIZATION OF COMPLEX NETWORKS: TOPOLOGY AND METHODS

Abstract. This paper discusses the main methods of immunization of modern computer networks. The greatest attention is paid to topologies and types of graph structures under consideration. An assessment and comparison of existing immunization strategies and methods for solving the problems of optimal selection of nodes for immunization in the context of the considered topologies is also carried out. The main advantages, disadvantages and areas of application of the selected topologies and methods for solving immunization problems are highlighted.

Key words: computer networks, immunization, cyber-physical systems, network security, scale-free networks.

Acknowledgements: The study was carried out within the framework of a grant from the President of the Russian Federation for state support of young Russian scientists – candidates of sciences MK-3861.2022.1.6.

В настоящее время, в связи с ростом степени автономности и адаптивности киберфизических систем (КФС), остро стоит вопрос об иммунизации компьютерных сетей. Методы и способы детектирования аномалий в компьютерных сетях [1], как и вопросы кластеризации и изоляции [2–4] узлов компьютерных сетей, в настоящий момент хорошо изучены, однако одновременная применимость ряда этих методов

в КФС остается отрытым вопросом. Также стоит отметить дополнительный интерес к совокупному использованию подобных подходов, который только вызывает возрастающая популярность иммунного анализа компьютерных сетей.

В данной работе рассматриваются основные виды топологий компьютерных сетей, существующие как на практике, так и применяемые для моделирования искусственной иммунизации сетей:

1. Иерархическая и кластерная топология.
2. Параллельная (коррелированная) топология.
3. ER-, SF- и бимодальные сети.

Также рассматриваются основные стратегии иммунизации:

1. Случайная.
2. Селективная.
3. Пропорциональная
4. Целевая.
5. Родственная (по знакомству).
6. Иммунизирующий агент.

Основной целью работы является обобщение современных существующих данных, создание теоретического базиса для последующего упрощения исследований и работ в рассматриваемой области, а также подведение итогов: рекомендательные оценки, сравнение основных преимуществ и недостатков каждого метода, выделение областей применения.

1. Иерархическая и кластерная топология: отсутствующий, случайный и селективный случаи иммунизация

В работе [5] авторами основной акцент сделан на результатах имитационных исследований того, как вирусные инфекции распространяются через иерархическую и кластерную топологии сетей и как выборочная иммунизация оказывает на это воздействие.

Были проанализированы две модели вирусной инфекции: одиночное заражение и множественные заражения (под моделями вирусной инфекции авторы подразумевают количество копий, которые одна копия вируса может создать на узлах, подключенных к хосту, на котором она выполняется).

Для обеспечения выполнимости эксперимента было сделано несколько допущений и упрощений. Во-первых, хотя несколько копий вируса могут существовать одновременно на одном хосте, авторы предполагают, что количество вирусов на одном хосте не превышает 100. Во-вторых, заражение каждый раз начиналось с произвольного узла, выбираемого случайным образом, а случайность процесса заражения была смоделирована за счет многократного повторения модельных расчетов.

Изначально авторы проводили тесты на сетях с отсутствием иммунизации, чтобы получить представление о характеристиках инфекции, а также для получения контрольных значений для будущих экспериментов по иммунизации.

В случае с иерархической топологией были отмечены следующие важные факты:

1. Огромные различия во времени, которое необходимо для заражения всей сети. Данный эффект обосновывается случайностью протекания процесса распространения инфекции.

2. В модели одиночного заражения в зависимости от процесса заражения (например, пути заражения и т.д.) вирус может потратить большую часть своего времени на заражение и повторное заражение небольшой части сети, и может пройти значительное время, прежде чем ему удастся выйти на другие участки сети.

3. В модели множественного заражения наблюдалось существенно более быстрое распространение инфекции. Кроме того, по сравнению с базовым исследованием, в экспериментах с множественной моделью заражения наблюдается значительно меньше вариаций и отклонений.

Результаты этих экспериментов показали, что при высокой вероятности заражения снижается чувствительность динамики заражения к стохастической изменчивости.

В случае с кластерной топологией были отмечены следующие важные факты:

1. В кластерной топологии сети наблюдается значительно большая скорость роста заражения, однако заражение по-

следних 10 % сети занимает много больше времени, чем в случае с иерархической топологией сети.

2. Как и в иерархическом случае, в модели множественного заражения наблюдалось существенно более быстрое распространение инфекции, чем в случае одиночного заражения.

Также авторами подчеркивается тот факт, что определенные узлы в сети во время заражения подвергались атакам существенно чаще своих соседей. Данное явление имело место в обеих рассмотренных топологиях. Исходя из этого, авторами далее рассматривается идея иммунизации отдельных узлов сети.

Как отмечают авторы, в их намерения не входит исследование способов проведения иммунизации. Скорее, если предположить, что методы иммунизации существуют, их цель состоит в том, чтобы изучить наиболее эффективную стратегию иммунизации.

В случае с иммунизацией случайных узлов были отмечены следующие важные факты:

1. С ростом количества иммунизированных узлов падает скорость заражения системы.

2. Случайная иммунизация показывает лучшие результаты в случае иерархической топологии сети. Данное явление обосновывается тем, что в иерархической топологии, в случае удачной иммунизации ключевого узла, намного проще разделить сеть на несколько независимых участков и изолировать зараженный участок.

В случае с селективной иммунизацией были отмечены следующие важные факты:

1. В случае с иерархической топологией селективная иммунизация работает значительно лучше, чем иммунизация случайных узлов. Данное явление (как и в случае со случайной иммунизацией) обусловлено тем, что в иерархической топологии, в случае удачной иммунизации ключевого узла, намного проще разделить сеть на несколько независимых участков и изолировать зараженный участок. В селективной иммунизации выбирались преимущественно такие узлы.

2. В случае с кластерной топологией отмечается, что иммунизация узлов с самой высокой частотой заражения дала результат лучше, чем иммунизация узлов с наибольшим количеством связей.

3. В случае с кластерной топологией иммунизация узлов с наибольшим количеством связей по эффективности сопоставима с иммунизацией случайных узлов, однако показывает более удовлетворительные результаты в случаях малого процента иммунизации, нежели в случайном случае.

Авторы работы интерпретируют результаты экспериментов следующим образом:

1. Селективная иммунизация работает эффективнее, чем случайная иммунизация.

2. Селективную иммунизацию достаточно просто реализовать в иерархической топологии — достаточно иммунизировать разделяющие (корневые) узлы.

3. Иммунизация узлов в кластерной топологии сети не является тривиальной задачей.

4. По сравнению с иерархической моделью, в кластерной топологии для достижения аналогичного эффекта требуется больший процент иммунизированных узлов.

Положительными сторонами селективной иммунизации иерархической и кластерной топологии можно считать:

1. Повышенный уровень защищенности системы (в сравнении с прочими рассмотренными подходами).

2. Простоту моделирования иерархической и кластерной топологий.

Однако, стоит отметить и отрицательные факторы данного подхода:

1. Сложность обобщения полученных результатов на прочие, приближенные к реальным, топологии сетей.

2. Необходимость наличия всей информации об узлах и топологии сети, что на практике не всегда является возможным.

Область применения данного подхода: моделирование теоретических случаев и анализ динамики распространения вирусной инфекции, а также сети, чья конфигурация достоверно известна и в должной степени гомогенна.

2. Параллельная honeypot топология: иммунизирующий агент

Авторы работы [6] полагают, что предыдущие стратегии иммунизации (статические и централизованные) за прошедшие годы не смогли в полной мере удовлетворить необходимые требования по сдерживанию ВПО (вредоносного программного обеспечения); в следствии чего предлагается использовать распределенный иммунизирующий агент и распространять его по сети так же, как это происходит и с ВПО.

Основная проблема распространения иммунизирующего агента, по мнению авторов, заключается в том, что он неизбежно будет отставать от ВПО. Авторами предлагается к рассмотрению теоретический иммунизирующий агент, способный перескакивать зараженные ВПО сегменты сети. Обоснование своей гипотезы авторы подкрепляют тем фактом, что разработка иммунизирующего агента может быть завершена в течение времени, сопоставимого со временем распространения ВПО в сети.

Целью работы авторов является попытка найти удачные стратегии иммунизации сети, основываясь на минимизации зараженных ВПО сегментов. Под размер зараженного (вирусного) кластера авторы понимают количество инфицированных узлов при времени моделирования, устремленному к бесконечности. Под размером иммунизированного кластера авторы понимают общее число иммунизированных узлов.

По своей сути, описанная авторами динамика включает конкуренцию между двумя типами ветвящихся процессов в сети, где первый тип создает связанный вирусный кластер, а второй создает набор иммунизированных кластеров. В отличие от централизованных подходов, этот подход сводит на нет необходимость в глобальном знании топологии.

Авторами принимается допущение о том, что время распространения ВПО и иммунизации сети является постоянным, а иммунизация и заражение узлов в сети происходит за один модельный шаг.

В качестве демонстрации эффективности разработанного подхода, авторы предлагают изначальную модификацию задачи: изначальная топология сети расширяется дополнительными ребрами. Дополнительно введенные ребра представляют собой ребра иммунизации и используются для транспортировки иммунизирующих агентов. Топология ребер базируется как на существующей сети, так и нововведенных узлах — honeypot'ах.

Иммунизирующие ребра соединяют вырабатывающий иммунизирующий агент узел с узлами, находящимися за пределами его непосредственной близости, как определено исходной сетью. С практической точки зрения это означает, что иммунизирующие агенты эффективно развертываются за пределами вирусного кластера, минуя границу заражения ВПО. Оказавшись в этом положении, они могут изменить топологию пространства, оставшегося в распоряжении ВПО, иммунизируя узлы, которые в противном случае принадлежали бы зараженному кластеру.

Затем авторы изменяют динамику распределенной иммунизации следующим образом: вирус распространяется через исходную сеть *G1*, тогда как иммунизирующий агент развертывается через частично коррелированную сеть *G2*, которая получается путем случайного добавления ребра к *G1*. Данное решение позволяет преодолевать зараженные кластеры иммунизирующему агенту и иммунизировать еще не зараженные участки, дополнительно выигрывая во времени.

Таким образом, авторы приходят к выводу, что динамическая иммунизация, которая используется в частично коррелированных сетях, может значительно уменьшить размер вирусного кластера, расходуя при этом лишь малую пропускную способность коррелированных сетей.

Далее авторы видоизменяют топологию модели, дополняя коррелированные сети honeypot системами. Такая архитектура имеет два основных преимущества по сравнению со «случайным» решением:

1. Более реалистично и технически осуществимо.

2. Значительно более эффективно, чем случайное развертывание иммунизирующих ребер.

Архитектура honeypot системы строится следующим образом: внедряется возможность разработки иммунизирующего агента в набор узлов — honeypot'ов, которые случайным образом встраиваются в сеть, так что любое ВПО, которое распространяется по сети, скорее всего, быстро доберется до них.

По задумке авторов заражение должно распространяться по сети свободно ровно до того момента, пока не заразит honeypot. После заражения любого из honeypot'ов начинается развитие иммунизирующего агента. К моменту начала развития иммунизирующего агента, ожидаемый размер зараженного кластера сети должен быть приблизительно равен размеру сети, деленному на количество honeypot'ов. По мере распространения заражения по сети все honeypot'ы должны получить друг от друга информацию о вирусе, и самостоятельно начать производство иммунизирующего агента.

Авторами работы доказано и показано с помощью численного моделирования, что при условии зависимости количества honeypot'ов от размера сети N и при условии скорости роста количества honeypot'ов $\geq\sqrt{N}$ размер зараженного кластера будет стремиться к нулю при стремлении размера всей сети к бесконечности.

Хотя представленная авторами архитектура внедрения и наращивания honeypot'ов с иммунизирующими агентами является лишь концепцией, математические выкладки и результаты численного моделирования являются весьма воодушевляющими и стимулирующими к дальнейшему изучению подобного материала.

Положительными сторонами параллельной honeypot топологии можно считать:

1. Простоту и скорость реализации данного метода.

2. Доказанную эффективность (в пределе) для растущих сетей.

Однако, стоит отметить и отрицательные факторы данного подхода:

1. Невозможность (на данный момент) автономной разработки иммунизирующего агента.

2. Необходимость наращивания значительной мощности для поддержания параллельной сети при масштабировании главной сети.

3. Типичные ограничения для динамических систем иммунизации: «запоздалую» реакцию и повышенные требования к вычислительным ресурсам иммунизирующих систем.

Область применения данного подхода: одноранговые сети (за счет возможности безболезненной кластеризации) с достаточной ресурсоемкостью для поддержания работоспособности параллельной иммунизирующей сети.

3. Безмасштабные сети: пропорциональный, целевой и родственный случаи иммунизации

Авторами работы [7] предлагается рассматривать современные сети как безмасштабные сети. Под безмасштабной сетью (или масштабно-инвариантной сетью) (scale-free network, SF-сети) обычно понимают граф, степени вершин которого распределены по степенному закону, такому что доля вершин со степенью k примерно или асимптотически пропорциональна $k^{-\gamma}$.

Основной нюанс этих сетей заключается в высокой степени локальной кластеризации и относительно малой длине пути между любыми двумя вершинами графа.

Данную интерпретацию классических компьютерных сетей весьма удобно использовать для прогнозирования эпидемий ВПО, т.к. SF-сети склонны к распространению и сохранению инфекций, какой бы вирулентностью ни обладал инфекционный агент.

Ввиду описанной слабости поиск оптимальных стратегий иммунизации, направленных на минимизацию риска эпидемических вспышек в SF-сетях, становится главной задачей исследования авторов.

Также авторами отмечается, что единообразно применяемые стратегии иммунизации эффективны только в сложных сетях с ограниченными флуктуациями связности. Однако, в SF-сетях инфекция не ликвидируется даже при наличии чрезмерно высокой доли иммунизированных узлов. Согласно приведенным исследованиям, авторы полагают, что в SF-системах нет какой-либо критической доли иммунизированных узлов: только при условии полной иммунизации можно достигнуть остановки распространения заражения. Чтобы разрешить вопрос полезной иммунизации сети, авторами далее определяются оптимальные стратегии иммунизации SF-сетей.

Первоначально авторами рассматриваются однородные сложные сети, т.к. широкий класс сетевых моделей имеет экспоненциально ограниченные флуктуации связности. В конечном итоге аналитические решения и численное моделирование сводится к выводу о том, что существует вычислимый эпидемический порог, при котором в сети либо со временем наступает полное заражение, либо полное выздоровление.

Однако, рассматривая SF-сети, авторы утверждают, что для малых SF-сетей при любом положительном значении параметра инфицирования инфекция может проникнуть в систему с конечной скоростью распространения, отличной от нуля. Численно решив уравнения в явном виде для описанных SF-сетей, авторы обобщают свое заключение на SF-сети любого размера. Этот результат можно обобщить и для SF-сетей с произвольным распределением связности, которые показывают исчезновение эпидемического порога как поведение степенного закона с показателем степени, зависящим от показателя связности γ.

Далее, численно анализируя стационарные свойства плотности зараженных узлов (распространенность инфекции) при различных значениях иммунизации, авторы отмечают, что инфекция в SF-сетях всегда достигает эндемического состояния, если размер сети достаточно велик. Это указывает на отсутствие порога иммунизации для SF-сетей. Таким образом, авторы приходят к выводу о необходимости создания оптимальной стратегии иммунизации SF-сетей.

Одним из простейших примеров оптимальной стратегии иммунизации является метод пропорциональной иммунизации, подразумевающий под собой условие наличия иммунитета у большей части узлов в группах с большей связностью. В конечном счете, на основе полученных данных, рекомендация автором сводится к лечению с пропорционально более высокими скоростями наиболее подключенных узлов.

Однако авторы также учитывают топологию SF-сетей и предлагают более эффективный метод иммунизации на основе иерархии узлов — целевую иммунизацию.

В частности, авторы ссылаются на работы, в которых было показано, что классические сети обладают заметной устойчивостью к случайным сбоям соединения, но в то же время, SF-сети сильно подвержены избирательному повреждению (если удалить несколько наиболее связанных узлов, сеть резко уменьшит свою способность передавать информацию). Применяя этот аргумент к случаю распространения эпидемии, авторы проводят численную оценку и моделирование конструктивности целевой иммунизации, в которой постепенно делают иммунными узлы с наиболее высокой степенью связи, то есть те, которые с большей вероятностью распространяют болезнь.

В SF-сетях данный подход приводит к резкому повышению устойчивости сети к инфекциям за счет незначительной доли иммунных индивидуумов.

Подводя итог, авторы отмечают, что, хотя SF-сети особенно слабы в борьбе с инфекциями, существуют возможности разработки чрезвычайно эффективных стратегий иммунизации, не подразумевающих под собой иммунизацию всей или большей части сети.

Далее, рассматривая тему эффективной иммунизации SF-сетей, стоит отметить работу [8]. В ней рассматривается математи-

ческая модель SF-сети, базирующаяся на иммунизации знакомых узлов. Согласно данной модели, иммунизации подвергается только малое количество тесно связанных узлов, а заражение сети, при соблюдении данной стратегии, не достигает критического порога заражения.

В описываемом авторами подходе выбирается случайная доля p из N узлов и ищется случайный сосед, с которым они находятся в контакте, то есть подразумевается иммунизация именно знакомых узлов от изначально рассмотренных — изначально выбранные узлы остаются нетронутыми. Преимуществом данной стратегии является локальность и отсутствие необходимости знаний о топологии сети.

Далее авторами расширяется выбранный подход до «родственной стратегии иммунизации», которая требует иммунизации знакомых, на которых ссылаются по крайней мере n узлов (ранее авторы полагали, что количество ссылок на выбранный узел $n = 1$).

Хотя авторами и не отмечается достаточно высокая скорость остановки распространения инфекции, данный метод имеет среднюю эффективность, находящуюся между затратным методом пропорциональной иммунизации и методом целевой иммунизации, требующим дополнительных знаний о топологии сети, при этом сам метод не требует ни излишней иммунизации прочих узлов, ни дополнительной информации о топологии системы. Также авторами отмечается, что данный подход можно использовать еще до начала распространения эпидемии, поскольку он не требует знания цепочки заражения.

Таким образом, положительными сторонами родственной иммунизации (в сравнении с пропорциональной и целевой) можно считать:

1. Отсутствие необходимости знаний о топологии сети.

2. Средние требования к вычислительным мощностям системы.

3. Возможность использовать данный подход заранее (до начала распространения вирусной инфекции в системе).

Однако, стоит отметить и отрицательный фактор применения данного подхода: пониженную, но все еще достаточно высокую эффективность иммунизации (в сравнении с пропорциональной иммунизацией).

Область применения данного подхода: статические (в должной мере) SF-сети, позволяющие провести первые этапы иммунизации до начала распространения вирусной инфекции, достаточно устойчивые к изоляции опорных узлов.

4. SF- и ER-сети: целевой случай иммунизация

В работе [9] авторы изучают целевую иммунизацию в сетях с ограниченными знаниями (под ограниченными знаниями понимается полное или частичное отсутствие информации о топологии сети).

Авторами предложены конкретные механизмы иммунизации сетей с ограниченной информацией об их топологии. Также авторами дополнительно рассматривается эффективность применения их подходов в ER- и SF-сетях. ER-сети представляют собой графовые топологии, сгенерированные по модели Эрдёша-Реньи. В модели Эрдёша-Реньи все графы с фиксированным набором вершин и фиксированным набором рёбер одинаково вероятны.

В общем случае, авторы позднее дополняют свой подход следующим шагом: как только получена информация об узле, можно также получить частичную информацию о соседних узлах и использовать эту информацию для иммунизации узла с наивысшей степенью связности.

Авторы отмечают, что добавление этих знаний в конечном итоге не сильно меняет ключевые результаты предлагаемой модели, а потому предлагают остановиться на первоначальном результате.

Далее, в ходе численного моделирования на графах с разной топологией доказывается эффективность метода: отмечается снижение скорости распространения вируса в большей степени, чем при использовании подхода случайной иммунизации.

Стоит отметить, что целевая иммунизация SF-сетей рассматривалась и ранее, однако действительный интерес представляет именно эффективность применения данного подхода для объединенных ER-сетей, которым ранее уделялось намного меньшее внимание исследователей.

5. SF- и бимодальные сети: перколяция связей

Авторами работы [10] изучается вопрос возникновения и распространения вирусной инфекции в сетях. Особый акцент авторы делают на SF- и бимодальных сетях (сети, подразумевающие двухуровневую разнородность и поддерживающие отношение субъектов одного класса ко второму, пример: отношение множества людей к событиям как участников и не участников). В качестве базовой модели распространения авторами используется SIR («восприимчивый — инфицированный — удаленный»).

Также авторы приводят результаты моделирования динамики распространения эпидемии. Сопоставляя модель со статической моделью перколяции связей, были получены аналитические результаты для общего числа инфицированных узлов. Таким образом, модель перколяции связей может успешно применяться для замещения базовой модели, упрощая расчеты, но не теряя в точности.

Дополнительно рассмотренная модель изучается с различными стратегиями иммунизации, включая случайную, выборочную иммунизацию и иммунизацию по знакомству (схожую с родственной иммунизацией).

Особое внимание авторы уделяют модели иммунизации по знакомству. Основная проблема, связанная с целевым подходом к иммунизации, заключается в том, что он требует знания топологии сети — зачастую подобная информация может отсутствовать или быть недостоверной. Стратегия иммунизации по знакомству устраняет необходимость в глобальных знаниях топологии рассматриваемой сети.

В качестве результатов работы следует отметить следующие факты: стратегия иммунизации по знакомству эффективна для любой широкомасштабной распределенной сети (здесь авторы приводят примеры безмасштабных и бимодальных распределений, которые распространены во многих естественных сетях). Также авторами рассматривается пример «ассортативно смешанной сети» (узлы высокой степени связности стремятся соединиться с другими узлами высокой степени связности). Дополнительно обсуждается эффективность рассмотренной стратегии иммунизации в сочетании с эпидемиологической моделью SIR.

Следует отметить следующие положительные стороны замещения классической модели распространения инфекции моделью перколяции связей:

1. Упрощения расчетов и моделирования за счет ухода от топологии сети к среде наблюдения явления, внешним источникам явления и замещению уравнений способа протекания заражения.

2. Возможность упрощенного рассмотрения бимодальных сетей с помощью теории перколяции.

Также стоит отметить и отрицательный фактор данного подхода: избыточное влияние вероятностных процессов (как результат использования теории перколяции).

Область применения данного подхода в конечном итоге сводится к вероятностной оценке протекания заражения в сетях и возможности превентивной реконфигурации уже используемых сетей.

Заключение

В работе были рассмотрены основные виды и топологии сетей (классическая, иерархическая и кластерная, безмасштабная, параллельная и прочие), а также основные способы иммунизации описанных видов сетей (случайная, селективная, целевая и другие). Были выделены положительные и отрицательные стороны для каждого случая, описаны основные области применения и степени применимости каждого из подходов.

В качестве дальнейшего направления исследований в заданной области авторами рассматривается возможность классификации основных параметров распространения вирусных инфекций в сетях, а также дополнительные исследования динамических топологий сети (периодически изменяющиеся, растущие и прочие).

СПИСОК ИСТОЧНИКОВ

1. **Фатин А. Д., Павленко Е. Ю.** Анализ моделей представления киберфизических систем в задачах обеспечения информационной безопасности / А. Д. Фатин, Е. Ю. Павленко // Проблемы информационной безопасности. Компьютерные системы. 2020.— № 2.— С. 109–121.

2. Probabilistic partition of unity networks: clustering based deep approximation / N. Trask et al. // 2021, arXiv:2107.03066.

3. Adaptation of Vehicular Ad hoc Network Clustering Protocol for Smart Transportation / M. Ahmad et al. // Computers, Materials & Continua. Vol. 67. No. 2. 2021. P. 1353–1368. DOI:10.32604/cmc.2021.014237.

4. Adaptive Network Automata Modelling of Complex Networks / A. Muscoloni, U. Michieli, C. V. Cannistraci // Preprints 2020, 2020120808. DOI: 10.20944/preprints202012.0808.v1.

5. On computer viral infection and the effect of immunization / Ch. Wang, J. C. Knight, M. C. Elder // Proceedings 16th Annual Computer Security Applications Conference (ACSAC'00), 2000. P. 246–256. DOI: 10.1109/ACSAC.2000.898879.

6. Distributive immunization of networks against viruses using the 'honey-pot' architecture / Goldenberg, J. et al. // Nature Physics. Vol. 1, 2005. P. 184–188. DOI: 10.1038/nphys177.

7. **Pastor-Satorras R., Vespignani A.** Immunization of complex networks / R. Pastor-Satorras, A. Vespignani // Phys. Rev. E65, 036104. DOI: 10.1103/PhysRevE.65.036104.

8. Efficient Immunization Strategies for Computer Networks and Populations / R. Cohen, S. Havlin, D. ben-Avraham // Physical Review Letters 91(24):247901. DOI: 10.1103/PhysRevLett.91.247901

9. Modelling Artificial Immunization Processes to Counter Cyberthreats / D. Zegzhda, E. Pavlenko, E. Aleksandrova // Symmetry 2021, 13, 2453. DOI: 10.3390/sym13122453.

10. Virus Propagation on Time-Varying Networks: Theory and Immunization Algorithms / B. A. Prakash et al. // Machine Learning and Knowledge Discovery in Databases. ECML PKDD2010. Lecture Notes in Computer Science, Vol. 6323, 2010.— P. 99–114. DOI: 10.1007/978-3-642-15939-8_7.

11. **Zegzhda, D. P.** Aspects of Information Security of Computer Systems / D. P. Zegzhda, I. Y. Zhukov // CEUR workshop proceedings: Selected Papers of XI International Scientific and Technical Conference on Secure Information Technologies (BIT 2021), Moscow, 06–07 апреля 2021 года.— Moscow: CEUR Workshop Proceedings, 2021.— P. 214–228.

REFERENCES

1. **Fatin A. D., Pavlenko E. Yu.** Analiz modelej predstavleniya kiberfizicheskih sistem v zadachah obespecheniya informacionnoj bezopasnosti [Analysis of models of representation of cyber-physical systems in the problems of ensuring information security] / A. D. Fatin, E. Yu. Pavlenko // Problems of information security. Computer systems. 2020.— No 2.— P. 109–121. (In Russian)

2. Probabilistic partition of unity networks: clustering based deep approximation / N. Trask et al. // 2021, arXiv:2107.03066.

3. Adaptation of Vehicular Ad hoc Network Clustering Protocol for Smart Transportation / M. Ahmad et al. // Computers, Materials & Continua. Vol. 67. No. 2. 2021. P. 1353–1368. DOI:10.32604/cmc.2021.014237.

4. Adaptive Network Automata Modelling of Complex Networks / A. Muscoloni, U. Michieli, C. V. Cannistraci // Preprints 2020, 2020120808. DOI: 10.20944/preprints202012.0808.v1.

5. On computer viral infection and the effect of immunization / Ch. Wang, J. C. Knight, M. C. Elder // Proceedings 16th Annual Computer Security Applications Conference (ACSAC'00), 2000. P. 246–256. DOI: 10.1109/ACSAC.2000.898879.

6. Distributive immunization of networks against viruses using the 'honey-pot' architecture / Goldenberg, J. et al. // Nature Physics. Vol. 1, 2005. P. 184–188. DOI: 10.1038/nphys177.

7. Pastor-Satorras R., Vespignani A. Immunization of complex networks / R. Pastor-Satorras,

A. Vespignani // Phys. Rev. E65, 036104. DOI: 10.1103/PhysRevE.65.036104.

8. Efficient Immunization Strategies for Computer Networks and Populations / R. Cohen, S. Havlin, D. ben-Avraham // Physical Review Letters 91(24):247901. DOI: 10.1103/PhysRevLett.91.247901

9. Modelling Artificial Immunization Processes to Counter Cyberthreats / D. Zegzhda, E. Pavlenko, E. Aleksandrova // Symmetry 2021, 13, 2453. DOI: 10.3390/sym13122453.

10. Virus Propagation on Time-Varying Networks: Theory and Immunization Algorithms / B. A. Prakash et al. // Machine Learning and Knowledge Discovery in Databases. ECML PKDD2010. Lecture Notes in Computer Science, Vol. 6323, 2010.— P. 99–114. DOI: 10.1007/978-3-642-15939-8_7.

11. **Zegzhda D. P.** Aspects of Information Security of Computer Systems / D. P. Zegzhda, I. Y. Zhukov // CEUR workshop proceedings: Selected Papers of XI International Scientific and Technical Conference on Secure Information Technologies (BIT 2021), Moscow, 06–07 апреля 2021 года.— Moscow: CEUR Workshop Proceedings, 2021.— P. 214–228.

Статья поступила в редакцию 09.02.2022; одобрена после рецензирования 16.02.2022; принята к публикации 22.02.2022.

The article was submitted 09.02.2022; approved after reviewing 16.02.2022; accepted for publication 22.02.2022.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

ФАТИН Александр Денисович — ассистент; Санкт-Петербургский Политехнический университет Петра Великого

ПАВЛЕНКО Евгений Юрьевич — к.т.н., доцент; Санкт-Петербургский Политехнический университет Петра Великого

FATIN Aleksander D.— Assistant Professor; Peter the Great St. Petersburg Polytechnic University

E-mail: sasha-fatin@mail.ru

PAVLENKO Evgeny Yu.— Senior Research Officer, Associate Professor; Peter the Great St. Petersburg Polytechnic University

E-mail: pavlenko_eyu@spbstu.ru

ORCID: 0000-0003-1345-1874

Научная статья
DOI 10.48612/jisp/vxp5-pvhu-m4a2
УДК № 004.056

Т. М. Татарникова[1], А. В. Сверликов[1], И. А. Сикарев[2]
[1]ФГАОУ ВО «Санкт-Петербургский государственный университет аэрокосмического приборостроения»
[2]Российский государственный гидрометеорологический университет

МЕТОДИКА ВЫЯВЛЕНИЯ АНОМАЛИЙ В ТРАФИКЕ ИНТЕРНЕТА ВЕЩЕЙ

Аннотация. Показано, что для маломощных устройств интернета вещей недоступны технологии защиты данных, применяемые в проводных сетях связи. Поэтому поиск атаки на устройства интернета вещей может быть реализован средствами анализа трафика, несущего атаку и вследствие этого отнесенного к аномальному. Предлагается методика поиска аномалии сетевого трафика интернета вещей. Рассматривается последовательность шагов выделения из трафика, генерируемого сенсорными устройствами интернета вещей случайной составляющей, оставшиеся после исключения основных характеристик и в которой может содержаться аномалия. Программная реализация предложенной методики может стать частью системы обнаружения вторжений для сетей интернета вещей.

Ключевые слова: Интернет вещей, аномальный трафик, безопасность данных, методика анализа трафика, система обнаружения вторжений

Research article
DOI 10.48612/jisp/vxp5-pvhu-m4a2

T. M. Tatarnikova[1], A. V. Sverlikov[1], I. A. Sikarev[2]
[1]St. Petersburg State University of Aerospace Instrumentation
[2]Russian State Hydrometeorological University

METHODOLOGY FOR DETECTING ANOMALIES IN THE TRAFFIC OF THE INTERNET OF THINGS

Abstract. It is shown that data protection technologies used in wired communication networks are not available for low-power devices of the Internet of things. Therefore, the search for an attack on IoT devices can be implemented by means of analyzing the traffic that carries the attack and, as a result, is classified as anomalous. A technique for searching for an anomaly in the network traffic of the Internet of things is proposed. A sequence of steps is considered to isolate a random component from the traffic generated by the IoT sensor devices, remaining after the exclusion of the main characteristics and which may contain an anomaly. The software implementation of the proposed technique can become part of the intrusion detection system for the Internet of things.

Keywords: Internet of things, anomalous traffic, data security, traffic analysis technique, intrusion detection system

Развитие технологий построения измерительных систем на основе интернета вещей (Internet of Things, IoT) является трендом современных инфокоммуникационных сетей.

Устройства IoT, как правило используют батареи в качестве источника питания, обладают низкой вычислительной мощностью, ограниченным объемом памяти [1]. Эти ограничения обусловили разработку протоколов, обеспечивающих передачу и обработку данных с минимальной вычислительной и коммуникационной нагрузкой, что делает устройства интернета

Издатель: Санкт-Петербургский политехнический университет Петра Великого

вещей уязвимыми к аномальному трафику — различным атакам, реализация которых может нанести серьезный ущерб эксплуатируемому оборудованию IoT и даже физический ущерб людям [2].

Известно, что системы обнаружения вторжений (Intrusion Detection System, IDS) не увеличивают нагрузку на сенсорные устройства и поэтому могут быть использованы для своевременного обнаружение аномального трафика, генерируемого интернетом вещей [3].

IDS для IoT представляет собой программное обеспечение, которое позволяет выявить несанкционированное получение доступа или вредоносную атаку на данные и оповестить о нарушении безопасности. Методы, используемые в системах обнаружения вторжения, относятся к активным и реализуют [4]:

- поиск по сигнатуре;
- поиск по аномалии.

Поиск по сигнатуре — способ, позволяющий определить атаку по некоторому шаблону, составленному по признакам известных на данное время атак. При совпадении признаков трафика и одного из шаблонов создается оповещение о нарушении безопасности. Такой метод позволяет определять только те вторжения, которые были ранее.

Поиск по аномалии — способ, при котором система сравнивает активность трафика с некоторой моделью стандартного (корректного) поведения того или иного устройства. При отклонении от нормы система так же оповестит о нарушении безопасности.

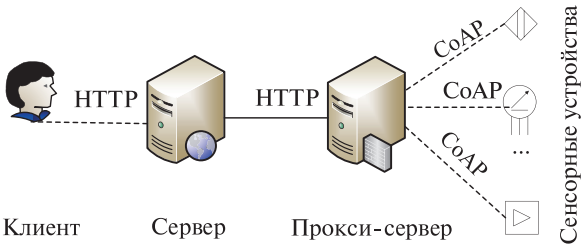


Рис. 1. Схема взаимодействия пользователя с сенсорным устройством

Fig. 1. Scheme of user interaction with a touch device

Исходя из скорости развития информационных технологий в целом и отрасли интернета вещей в частности, есть проблема быстрого устаревания систем обнаружения вторжения. IDS нуждаются в постоянных доработках и обновлениях, чтобы корректно реагировать на появление новых угроз, что в свою очередь требует больших временных и финансовых затрат.

В работе предлагается интеграция метода машинного обучения на основе самообучаемой нейронной сети в анализ трафика, генерируемого интернетом вещей. Самообучаемая нейросеть способна самостоятельно адаптироваться к обновляемой среде и оставаться актуальной на протяжении длительного времени.

1. Структура сети IoT

Взаимодействие Machine-to-Machine (M2M) является основой интернета вещей и подключения новых устройств в единую сеть [5]. Организация M2M взаимодействия основано на протоколе CoAP (Constrained Application Protocol). Этот протокол разработан специально для маломощных устройств, совместим с HTTP (HyperText Transfer Protocol) и реализует передачу данных с минимальными энергозатратами такими, как они есть — без модификаций и инкапсулирования [6].

Протокол CoAP является бинарным и работает поверх UDP (User Datagram Protocol — протокол пользовательских датаграмм), что позволяет работать с любым типом данных. Передача данных происходит эффективнее, так как бинарный код подразумевает короткие и маловесные пакеты — уменьшается объем, требуемый для передачи данных и появляется гибкость при работе с различными устройствами.

В общем случае, взаимодействие клиента с умным устройством интернета вещей выглядит, как на рис. 1, где подразумевается, что пакеты данных с сенсорного устройства проходят через прокси-сервер, где инкапсулируются в HTTP-пакеты.

Протокол CoAP не поддерживает режим шифрования, предусмотренный в прото-

коле TCP (Transmission Control Protocol — протокол управления передачей), а поддерживает более упрощенный вариант шифрования DTLS (Datagram Transport Layer Security — протокол датаграмм безопасности транспортного уровня).

Протокол DTLS может работать в одном из четырех режимов:

– NoSec — в этом режиме шифрование отключено;

– PreSharedKey — шифрование включено, добавляется список общих ключей шифрования AES (Advanced Encryption Standard — симметричный алгоритм блочного шифрования) и узлов, между которыми ведется конфиденциальный обмен данными;

– RawPublicKey — шифрование включено, используются асимметричные пары ключей с шифрованием по алгоритму AES;

– Сертификат — шифрование включено, устройство использует сертификаты X.509 с цифровыми подписями для распределения открытых ключей.

2. Методика анализа трафика

Модель трафика представляет собой закономерность изменения его основных характеристик во времени. В описании трафика обычно учитываются три характеристики [7]:

– тренд — описание поведения трафика во времени;

– сезонность — закономерность роста или падения объема трафика, связанных, например, с конкретным промежутком времени;

– случайная составляющая — остальные характеристики, оставшиеся после исключения основных характеристик, собственно, в которых и следует искать аномалии.

После выбора способа анализа, необходимо трафик разложить на составляющие:

– выделить тренд и сгладить исходные данные, например, при помощи скользящего окна, экспоненциального сглаживания или регрессии;

– вычесть или разделить сезонную составляющую из исходных данных, что зависит от выбранного способа;

– после удаления сезонного фактора и тренда остается случайная составляющая, которая и принимается за аномалию.

Примерами аномалий могут быть выбросы, сдвиги, изменения распределения значений, отклонения от среднего и совместные аномалии.

В качестве примера можно рассмотреть анализ трафика, основанный на циклическом алгоритме, состоящий из:

– отбора данных;

– сглаживания тренда;

– поиска возможных циклов;

– удаления трендовых компонент;

– проверки циклов;

– предсказания будущих циклов.

На этапе отбора данных определяется анализируемый временной ряд X_q (рис. 2). На рис. 2 изображен временной ряд, где по оси ординат отложен объем трафика V, а по оси абсцисс — период наблюдения T. Для дальнейшего анализа, временная шкала дискретизируется равными интервалами Δt наблюдения динамики изменения объема трафика.

Каждый интервал Δt содержит агрегацию данных, объем которых зафиксирован в этом интервале

$$X_q(\Delta t) = \sum_{j=1}^{R} V_j$$

где q — номер интервала ($q = 1, 2, ..., Q$); R — число пакетов в интервале Δt; X_q — ряд данных.

Для сглаживания необходимо исключить одиночные выбросы — $(L - 1)$ точку. Применяя, например, метод краткосрочной центрированной скользящей средней осуществляется переход к новому сглаженному ряду X_k, длина которого равна $N = Q - (L - 1)$, $k = \overline{1, N}$.

$$X_k = \frac{1}{L} \sum_{j=k}^{k+L-1} X_j.$$

После удаления случайных колебаний и выравнивания значений переходят

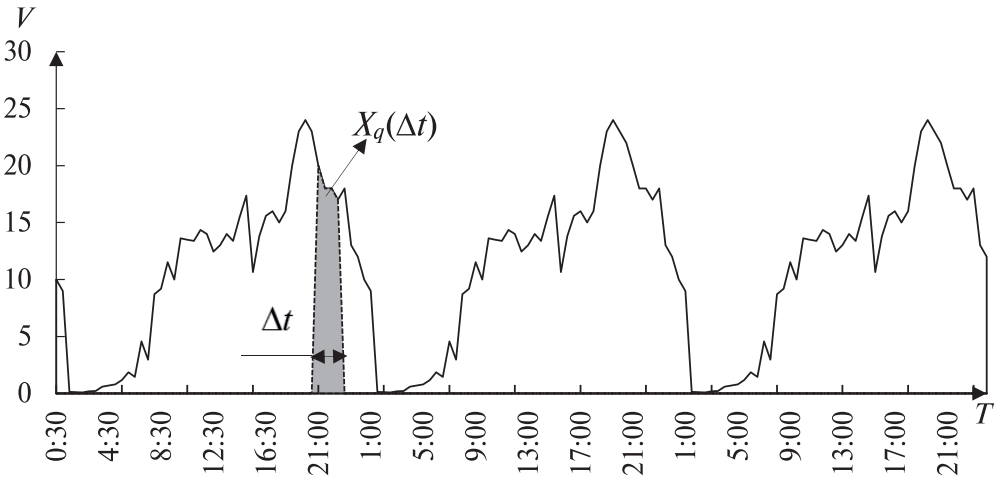


Рис. 2. Временной ряд трафика

Fig. 2. Traffic time series

к поиску циклов. Здесь применяется метод спектрального анализа. На графике спектра мощности (рис. 3) видны пики, образующиеся возле определенных частот. Пики указывают на возможные циклы.

Пики свидетельствует только о вероятности наличия циклов. Поэтому проводят еще несколько шагов для проверки наличия циклов. Одним из таких критериев является удаление трендовых компонентов. Применяя метод скользящей средней, удаляются силы роста в данных, а затем исходя из статистической значимости методами *F*-коэффициентов или хи-квадрат ищется отклонение от распределения спектра мощности.

3. Архитектура IDS интернета вещей

IDS для интернета вещей, имеет иерархическую структуру, как и сама сеть IoT — три компонентных уровня (рис. 4).

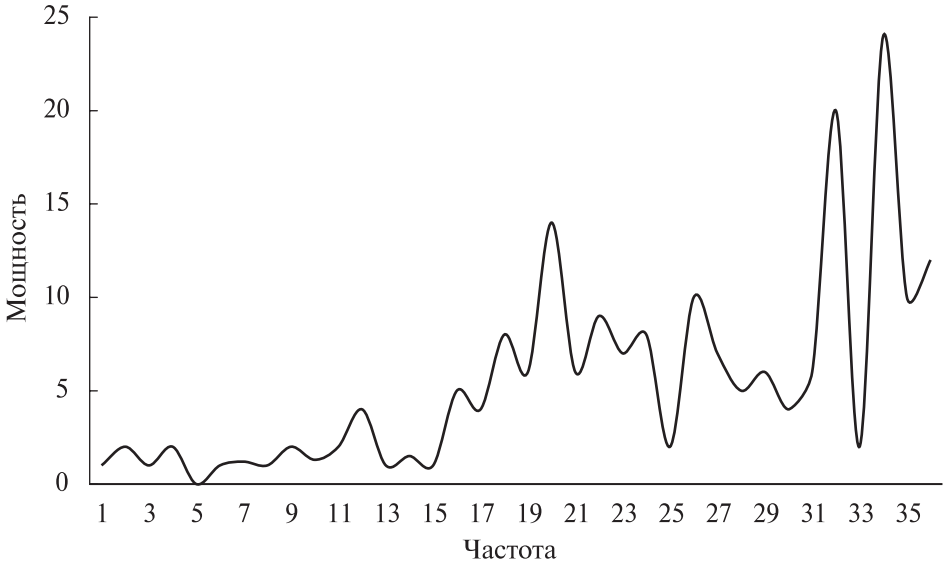


Рис. 3 Спектр мощности

Fig. 3 Power spectrum

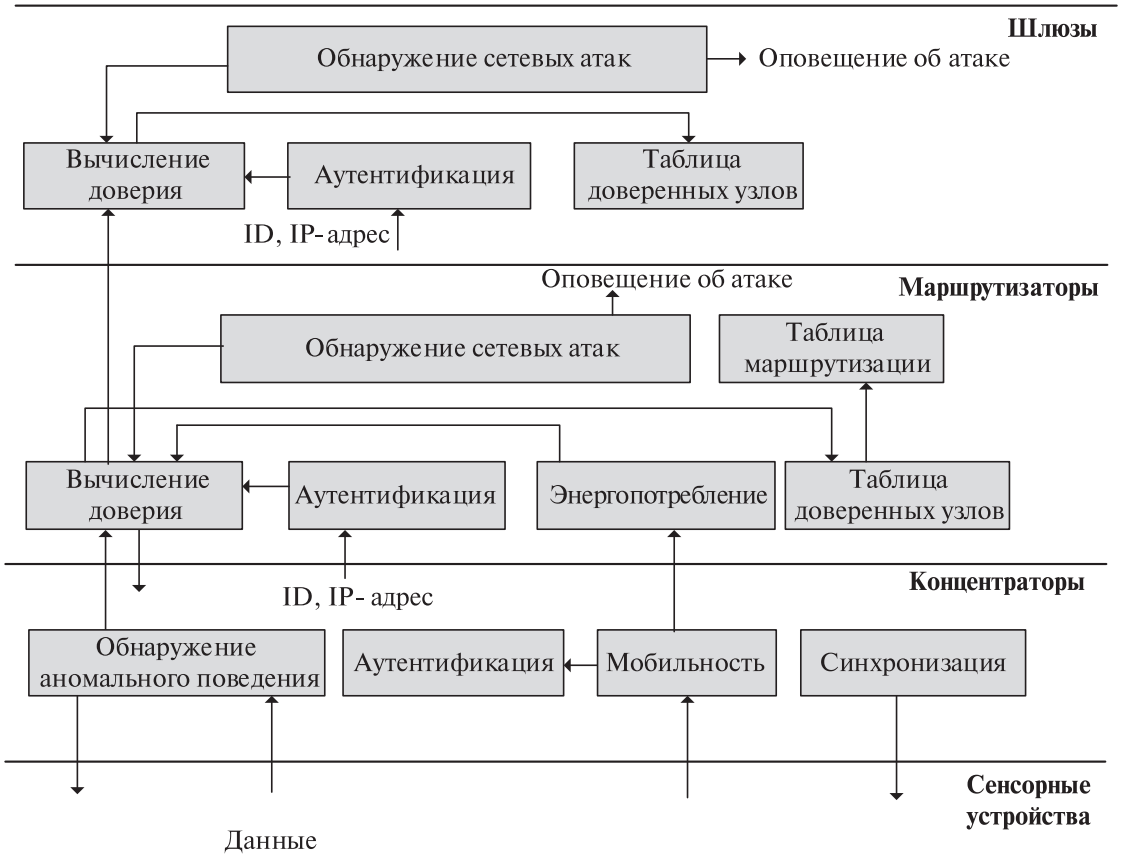


Рис. 4. Компоненты IDS интернета вещей

Fig. 4. IDS components of the Internet of Things

На нижнем уровне иерархии находятся кластерные беспроводные сенсорные сети [8]. Головной узел кластера отвечает за авторизацию других членов кластера, маршрутизатор отвечает за авторизацию головных узлов кластера, а шлюз за авторизацию маршрутизаторов.

Модуль обнаружения аномального поведения установлен в каждом кластере, интеллектуальная система обнаружения сетевых атак на маршрутизаторах и шлюзах.

IoT-устройства оповещают о своем присутствии в конкретном кластере, направляя идентификатор (ID), например, MAC-адрес, в модуль мобильности. Модуль мобильности отвечает за регистрацию вновь прибывших в кластер устройств и удаление вышедших из кластера устройств. Любой вновь прибывший узел проходит процедуру аутентификации [9].

Модуль синхронизации представляет собой тактовый генератор — формирование временных окон (слотов) и соответственно раундов беспроводной сенсорной сети, во время которых происходит опрос IoT-устройств.

Модуль обнаружения аномального поведения реализуется программно по приведенной в статье методике анализа трафика.

На уровне сети обнаружение атак реализуется программной моделью глубокого обучения, например нейронной сетью или случайным лесом [10].

Модуль вычисления доверия позволяет в онлайн режиме удалять скомпрометированные узлы из списка доверенных узлов, на основании которого формируется таблица маршрутизации.

Заключение

Предложенная в статье методика анализа трафика с целью поиска аномалий, несущих атаку на устройства интернета

вещей, является частью системы обнаружения вторжений и реализует свои функции на нижнем уровне архитектуры интернета вещей, то есть на уровне сенсорных устройств и кластеров беспроводной сенсорной сети.

Последовательность шагов методики направлена на выделение из трафика, генерируемого сенсорными устройствами интернета вещей случайной составляющей, оставшиеся после исключения основных характеристик, в которой может содержаться аномалия.

Работа имеет продолжение в части методики вычисления доверия узлам интернета вещей с учетом обнаруженной аномалии.

СПИСОК ИСТОЧНИКОВ

1. **Wang C., Lin H., & Jiang H.** CANS: Towards congestion-adaptive and small stretch emergency navigation with wireless sensor networks // IEEE Transactions on Mobile Computing. 2015. Vol. 15(5). P. 1077–1089.

2. **Татарникова Т.М., Богданов П.Ю., Краева Е.В**. Предложения по обеспечению безопасности системы умного дома, основанные на оценке потребляемых ресурсов // Проблемы информационной безопасности. Компьютерные системы. 2020. № 4. С. 88–94.

3. **Baddar S.A.-H., Merlo A., Migliardi M.** Anomaly Detection in Computer Networks: A State-of-the-Art Review // Journal of Wireless Mobile Networks, Ubiquitous Computing and Dependable Applications. 2014. Vol. 5. No. 4. P. 29–64.

4. **Шелухин О.И., Сакалема Д.Ж., Филинова А.С**. Обнаружение вторжений в компьютерные сети. Сетевые аномалии.— М.: Горячая линия — Телеком, 2013.— 220 с.

5. **Bogatyrev V.A., Bogatyrev A.V., Bogatyrev S.V**. Redundant Servicing of a Flow of Heterogeneous Requests Critical to the Total Waiting Time During the Multi-path Passage of a Sequence of Info-Communication Nodes," Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). 2020. Vol. 12563. P. 100–112.

6. **Lee Perry**. Internet of Things Architects. Packt Publishing, 2018. 514 p.

7. **Сахаров Д. В., Козлов Д. С.** Обнаружение аномального поведения устройства IoT в сети на основе модели трафика. 2018. С. 51–55.

8. **Basford P. J., Johnston S. J., Perkins C. S., Garnock-Jones T., Tso F. P., Pezaros D., Cox S. J.** Performance analysis of single board computer clusters. Future Generation Computer Systems. 2020. Vol. 102. P. 278–291.

9. **Dziubenko I. N. and Tatarnikova T. M.** Algorithm for Solving Optimal Sensor Devices Placement Problem in Areas with Natural Obstacles. 2018 Wave Electronics and its Application in Information and Telecommunication Systems (WECONF). 2018. P. 1–5. DOI: 10.1109/WECONF.2018.8604325.

10. **Татарникова Т.М., Бимбетов Ф., Богданов П.Ю**. Выявление аномалий сетевого трафика методом глубокого обучения // Известия СПбГЭТУ ЛЭТИ. 2021. № 4. С. 36–41.

11. **Дахнович А. Д.** Принципы феноменологического подхода в обеспечении кибербезопасности промышленного Интернета вещей / А. Д. Дахнович, Д. А. Москвин // Методы и технические средства обеспечения безопасности информации.— 2021.— № 30.— С. 97–98.

REFERENCES

1. **Wang C., Lin H., & Jiang H.** CANS: Towards congestion-adaptive and small stretch emergency navigation with wireless sensor networks // IEEE Transactions on Mobile Computing. 2015. Vol. 15(5). P. 1077–1089.

2. **Tatarnikova T.M., Bogdanov P.YU., Kraeva E.V**. Predlozheniya po obespecheniyu bezopasnosti sistemy umnogo doma, osnovannye na ocenke potreblyaemyh resursov [Smart home system security proposals based on the assessment of consumed resources] // Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy. 2020. № 4. S. 88–94. (In Russian)

3. **Baddar S.A.-H., Merlo A., Migliardi M.** Anomaly Detection in Computer Networks: A State-of-the-Art Review // Journal of Wireless Mobile Networks, Ubiquitous Computing and Dependable Applications. 2014. Vol. 5. No. 4. P. 29–64.

4. **Sheluhin O.I., Sakalema D.ZH., Filinova A.S**. Obnaruzhenie vtorzhenij v komp'yuternye seti. Sete-

vye anomalii. [Intrusion detection in computer networks. Network anomalies] — M.: Goryachaya liniya — Telekom, 2013.— 220 c. (In Russian)

5. **Bogatyrev V.A., Bogatyrev A. V., Bogatyrev S. V**. Redundant Servicing of a Flow of Heterogeneous Requests Critical to the Total Waiting Time During the Multi-path Passage of a Sequence of Info-Communication Nodes," Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). 2020. Vol. 12563. P. 100–112.

6. **Lee Perry**. Internet of Things Architects. Packt Publishing, 2018. 514 p.

7. **Sakharov D. V., Kozlov D. S.** Obnaruzhenie anomal'nogo povedeniya ustrojstva IoT v seti na osnove modeli trafika [Detection of abnormal behavior of an IoT device in a network based on a traffic model]. 2018. P. 51–55. (In Russian)

8. **Basford P. J., Johnston S. J., Perkins C. S., Garnock-Jones T., Tso F. P., Pezaros D., Cox S. J.** Performance analysis of single board computer clusters. Future Generation Computer Systems. 2020. Vol. 102. P. 278–291.

9. **Dziubenko I. N. and Tatarnikova T. M.** Algorithm for Solving Optimal Sensor Devices Placement Problem in Areas with Natural Obstacles. 2018 Wave Electronics and its Application in Information and Telecommunication Systems (WECONF). 2018. P. 1–5. DOI: 10.1109/WECONF.2018.8604325.

10. **Tatarnikova T.M., Bimbetov F., Bogdanov P. Yu.** Vyyavlenie anomalij setevogo trafika metodom glubokogo obucheniya [Network traffic anomaly detection using deep learning] // Izvestiya SPbGETU LETI. 2021. № 4. S. 36–41. (In Russian)

11. **Dahnovich A. D.** Principy fenomenologicheskogo podhoda v obespechenii kiberbezopasnosti promyshlennogo Interneta veshchej [Principles of the Phenomenological Approach in Ensuring Cybersecurity of the Industrial Internet of Things] / A. D. Dahnovich, D. A. Moskvin // Metody i tekhnicheskie sredstva obespecheniya bezopasnosti informacii.— 2021.— No 30.— P. 97–98. (In Russian)

Статья поступила в редакцию 07.02.2022; одобрена после рецензирования 15.02.2022; принята к публикации 22.02.2022.

The article was submitted 07.02.2022; approved after reviewing 15.02.2022; accepted for publication 22.02.2022.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

ТАТАРНИКОВА Татьяна Михайловна — д.т.н., профессор; Санкт-Петербургский государственный университет аэрокосмического приборостроения

TATARNIKOVA Tatyana M. — D. Sc. in Engineering, Professor; St. Petersburg State University of Aerospace Instrumentation

ORCID: 0000-0002-6419-0072

СВЕРЛИКОВ Александр Владимирович — студент; Санкт-Петербургский государственный университет аэрокосмического приборостроения

SVERLIKOV Alexander V. — student; St. Petersburg State University of Aerospace Instrumentation

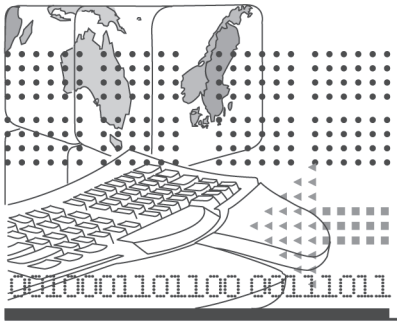
ORCID: 0000-0002-2447-2737

СИКАРЕВ Игорь Александрович — д.т.н., профессор; Российский государственный гидрометеорологический университет

SIKAREV Igor A. — D. Sc. in Engineering, Professor, Russian State Hydrometeorological University

Email: sikarev@yandex.ru

ORCID: 0000-0001-6289-3295



ПРАКТИЧЕСКИЕ АСПЕКТЫ КРИПТОГРАФИИ

Научная статья
DOI 10.48612/jisp/b94e-kfnm-8atd
УДК 519.718

Т. В. Стариков, К. Ю. Сопин, С. А. Диченко, Д. В. Самойленко
Краснодарское высшее военное училище имени С. М. Штеменко
Россия, 350063, Краснодар, ул. Красина, д. 4

КРИПТОГРАФИЧЕСКИЙ КОНТРОЛЬ ЦЕЛОСТНОСТИ ДАННЫХ ПО ПРАВИЛАМ ПОСТРОЕНИЯ КОДА РИДА-СОЛОМОНА

Аннотация. Рассматривается актуальная проблема оптимизации контроля целостности информации в системах хранения данных, функционирующих в условиях непрерывного роста ее объёмов и деструктивных воздействий злоумышленника. Представлен способ криптографического контроля целостности многомерных массивов данных на основе правил построения кодов Рида-Соломона.

Ключевые слова: защита информации, контроль целостности данных, криптографические методы, код Рида-Соломона.

Research article
DOI 10.48612/jisp/b94e-kfnm-8atd

T. V. Starikov, K. Yu. Sopin, S. A. Dichenko, D. V. Samoylenko
Krasnodar Higher Military School named after S. M. Shtemenko, Russia, Krasnodar

CRYPTOGRAPHIC CONTROL OF DATA INTEGRITY ACCORDING TO THE RULES OF CONSTRUCTION OF THE REED-SOLOMON CODE

Abstract. The actual problem of optimization of information integrity control in data storage systems functioning in conditions of continuous growth of its volumes and destructive influences of an attacker is considered. A method of cryptographic integrity control of multidimensional data arrays based on the rules for constructing Reed-Solomon codes is presented.

Keywords: information protection, data integrity control, cryptographic methods, Reed-Solomon code.

Современные условия функционирования систем хранения данных (СХД) характеризуются непрерывным ростом объёмов накапливаемой в них информации. При этом деструктивные воздействия злоумышленника и возмущения среды функционирования в условиях непрерывного роста объёмов накапливаемой информации порождают новую проблему, связанную с обеспечением защищенности больших объемов данных [1—3]. В современных СХД данная проблема разрешается, в ос-

новном, при помощи введения высокой избыточности при использовании методов теории надежности и методов из области обеспечения безопасности информации [4, 5]. Подобные решения приводят к преждевременному израсходованию ресурсов СХД и, как следствие, к снижению их работоспособности или полной дисфункции. Таким образом, разработка новых подходов к контролю целостности больших объёмов непрерывно накапливаемой в СХД информации без введения высокой избыточности является особенно актуальным в настоящее время [6–9].

Способ контроля целостности данных на основе правил построения кода Рида-Соломона

Блок данных **M** многомерного массива, подлежащего защите, представленный в виде вектора:

$$M = [\mu_1 \mu_2 \ldots \mu_\tau],$$

где $\mu_\vartheta \in \{0,1\}$ ($\vartheta = 1, 2, ..., \tau$), для осуществления контроля целостности по правилам, аналогичным правилам построения кода Рида-Соломона (РС), фрагментируется на подблоки данных фиксированной длины m:

$$\mathbf{M} = \{\mathbf{M}_0 \| \mathbf{M}_1 \| \ldots \| \mathbf{M}_\theta\},$$

где «||» обозначает операцию конкатенации (объединения) [10–12].

Для построения кода РС используется расширение двоичного поля Галуа GF(2), именуемое полем расширения и обозначаемое как $GF(2^m)$. Каждый ненулевой элемент поля $GF(2^m)$ представляется как степень α. При этом бесконечное множество элементов образуется из начального множества $\{0, 1, \alpha\}$, дополненного новыми элементами, полученными в результате последовательного умножения последнего элемента на α.

Описание конечного поля осуществляется с помощью примитивного полинома [13–17].

Неприводимый полином $f(x)$ порядка ξ будет примитивным, если наименьшим положительным целым числом n, для которого $x^n - 1$ делится на $f(x)$, будет $n = 2^m - 1$. Заметим, неприводимый полином — это полином который нельзя представить в виде произведения полиномов меньшего порядка [13].

По примитивному полиному $f(x)$ задаётся правило понижения степени $f(\alpha) = 0$ и строится множество элементов поля расширения $GF(2^m)$.

Рассмотрим характеристики кода РС:

$n = 2^m - 1$ — кодовая длина (количество подблоков блока данных, подлежащих защите, и подблоков с контрольной информацией);

t — количество исправляемых ошибок (обнаруживаемых и локализуемых подблоков блока данных с признаками нарушения целостности);

$k = 2^m - 1 - 2t$ — информационная длина (количество подблоков блока данных, подлежащих защите);

$r = 2t$ — количество контрольных символов (подблоков с контрольной информацией).

Порождающий полином кода РС $g(x)$ задаётся формулой:

$$g(x) = (x - \alpha^1) \cdot (x - \alpha^2) \ldots (x - \alpha^{2t}),$$

где «·» — знак умножения.

Исходный блок данных **M** многомерного массива, подлежащий защите, записывается с помощью примитивных элементов.

В зависимости от информационной длины $k = 2^m - 1 - 2t$ выбранного кода РС блок данных **M** при необходимости дополняется $\eta = k - \theta - 1$ нулевыми элементами (подблоками) для получения требуемой размерности и обозначается как:

$$\mathbf{M}' = \{\mathbf{M}_0 \| \mathbf{M}_1 \| \ldots \| \mathbf{M}_\theta \| \mathbf{M}_{\theta+1} \| \ldots \| \mathbf{M}_{\theta+\eta}\},$$

где $\mathbf{M}'$ — расширенный блок данных.

Подблоки блока данных $\mathbf{M}'$ представляются элементами $GF(2^m)$:

$$u(x) = M'_0 x^\rho + M'_1 x^{\rho-1} + \\ + M'_2 x^{\rho-2} + \ldots + M'_{\rho-1} x^1 + M'_\rho x^0,$$

где x — фиктивная переменная, $u(x)$ — информационное слово, $\rho = \theta + \eta$.

Для того чтобы реализовать контроль целостности данных осуществляется процесс кодирования, результатом которого является кодовый полином $c(x)$.

Для возможности осуществления контроля целостности данных выполняется кодирование, при котором вычисляется кодовый полином $c(x)$.

При несистематическом кодировании кодовый полином $c(x)$ вычисляется по формуле:

$$c(x) = u(x) \cdot g(x).$$

В случае выполнения такого кодирования кодовое слово не содержит в явном виде информационное слово [18]. То есть подблоки исходного блока данных **M**, подлежащего защите, записываются при помощью примитивных элементов a.

При систематическом кодировании выполняется типовой алгоритм кодирования для систематических циклических кодов:

– осуществляется сдвиг информационного полинома $u(x)$ в крайние старшие разряды кодового слова при помощи умножения полинома $u(x)$ на x^r;

– полученный полином $x^r \cdot u(x)$ делится на порождающий полином $g(x)$ для того, чтобы вычислить остаток от деления $p(x)$;

– искомый кодовый полином $c(x)$ определяется по формуле:

$$c(x) = x^r \cdot u(x) + p(x).$$

В случае систематического кодирования кодовое слово в явном виде содержит информационное слово (подблоки исходного блока данных **M**) [19, 20].

Кодовое слово записывается следующим образом:

$$c = \left\{ \mathbf{M}_0 \| \mathbf{M}_1 \| \ldots \| \mathbf{M}_\rho \| \mathbf{H}_0 \| \mathbf{H}_1 \| \ldots \| \mathbf{H}_\omega \right\},$$

где $\mathbf{H}_i$ — контрольные элементы, $i = 0, 1, \ldots, \omega$; $\omega = r - 1$.

После выполнения кодирования блока данных **M**, подлежащего защите, защищённые данные могут быть отправлены на хранение.

Декодирование выполняется при запросе защищённых данных на использование.

Пусть полином ошибки равен:

$$e(x) = e_{n-1}x^{n-1} + e_{n-2}x^{n-2} + \ldots + e_1x^1 + e_0.$$

Предполагается, что произошло ν ошибок, где $0 \le \nu \le t$, позиции ошибок обозначаются как $\iota_1, \iota_2, \ldots, \iota_\nu$.

В таком случае, полином ошибок примет вид:

$$e(x) = e_{\iota_1}x^{\iota_1} + e_{\iota_2}x^{\iota_2} + \ldots + e_{\iota_\nu}x^{\iota_\nu},$$

где ι_φ — значение φ-й ошибки.

Необходимо найти:

– количество ошибок ν (подблоков блока данных с признаками нарушения целостности);

– позиции ошибок $\iota_1, \ldots, \iota_\nu$ (подблоков блока данных с признаками нарушения целостности);

– значения ошибок $e_{\iota_1}, \ldots, e_{\iota_\nu}$ (при необходимости восстановления целостности подблоков блока данных).

Пусть $\upsilon(x) = c(x) + e(x)$ — полином, полученный при запросе данных, подлежащих защите, на использование.

Вычислим значения синдромов S_ϖ в точках α^ϖ:

$$S_\varpi = \upsilon(\alpha^\varpi) = c(\alpha^\varpi) + e(\alpha^\varpi) = e(\alpha^\varpi) =$$
$$= e_{\iota_1}(\alpha^\varpi)^{\iota_1} + e_{\iota_2}(\alpha^\varpi)^{\iota_2} + \ldots + e_{\iota_\nu}(\alpha^\varpi)^{\iota_\nu} =$$
$$= e_{\iota_1}(\alpha^{\iota_1})^\varpi + e_{\iota_2}(\alpha^{\iota_2})^\varpi + \ldots + e_{\iota_\nu}(\alpha^{\iota_\nu})^\varpi,$$

где $\omega = 1, 2, \ldots, 2t$.

Вводятся обозначения $Y_l = e_{\iota_l}$ и $X_l = \alpha^{\iota_l}$. Вычисляются значения X_l и логарифмируются по основанию α. Определяются все позиции ошибок ι_l. Получим систему:

$$\begin{cases} S_1 = Y_1X_1 + Y_2X_2 + \ldots + Y_\nu X_\nu, \\ S_2 = Y_1X_1^2 + Y_2X_2^2 + \ldots + Y_\nu X_\nu^2, \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ S_{2t} = Y_1X_1^{2t} + Y_2X_2^{2t} + \ldots + Y_\nu X_\nu^{2t} \end{cases}$$

из $2t$ нелинейных уравнений относительно ν неизвестных локаторов ошибок $X_1, \ldots, X_\nu$ и ν неизвестных значений ошибок $Y_1, \ldots, Y_\nu$. При решении данной системы обнаружим ν ошибок, в случае необходимости исправим их. При этом изначально найдём

значения локаторов ошибок, подставим их в систему и получим линейную систему относительно значений ошибок.

Рассмотрим полином:

$$\Lambda(x) = \Lambda_\nu x^\nu + \Lambda_{\nu-1} x^{\nu-1} + \ldots + \Lambda_1 x^1 + 1$$

и подберём его коэффициенты так, чтобы его корнями являлись значения X_l^{-1} $l = 1$, ..., ν для выполнения равенства:

$$\Lambda(x) = (1 - xX_1) \cdot (1 - xX_2) \ldots (1 - xX_\nu).$$

Выразим Λ_1, ..., Λ_ν через компоненты синдрома S_1, ..., S_{2t} и получим систему линейных уравнений, состоящих из:

$$\Lambda_1 S_{j+\nu-1} + \Lambda_2 S_{j+\nu-2} + \ldots + \Lambda_\nu S_\nu = -S_{j+\nu},$$

где $j = 1$, ..., ν. Запишем её в матричном виде:

$$\begin{pmatrix} S_1 & S_2 & S_3 & \ldots & S_{\nu-1} & S_\nu \\ S_2 & S_3 & S_4 & \ldots & S_\nu & S_{\nu+1} \\ S_3 & S_4 & S_5 & \ldots & S_{\nu+1} & S_{\nu+2} \\ & \ldots & & \ldots & & \ldots \\ S_\nu & S_{\nu+1} & S_{\nu+2} & \ldots & S_{2\nu-2} & S_{2\nu-1} \end{pmatrix} \begin{pmatrix} \Lambda_\nu \\ \Lambda_{\nu-1} \\ \Lambda_{\nu-2} \\ \ldots \\ \Lambda_1 \end{pmatrix} = \begin{pmatrix} -S_{\nu+1} \\ -S_{\nu+2} \\ -S_{\nu+3} \\ \ldots \\ -S_{2\nu} \end{pmatrix}.$$

Если произошло ν ошибок, то матрица:

$$A = \begin{pmatrix} S_1 & S_2 & S_3 & \ldots & S_{\nu-1} & S_\nu \\ S_2 & S_3 & S_4 & \ldots & S_\nu & S_{\nu+1} \\ S_3 & S_4 & S_5 & \ldots & S_{\nu+1} & S_{\nu+2} \\ & \ldots & & \ldots & & \ldots \\ S_\nu & S_{\nu+1} & S_{\nu+2} & \ldots & S_{2\nu-2} & S_{2\nu-1} \end{pmatrix}$$

невырождена, следовательно, система разрешима.

Способ контроля целостности данных на основе правил построения кода Рида-Соломона иллюстрируется на рисунке.

Пример. Дан блок данных **M** многомерного массива

$$\mathbf{M} = [1011011111000101].$$

Фрагментируем блок данных **M** на подблоки данных фиксированной длины m.

При $m = 4$ получим следующий набор подблоков:

$$\mathbf{M}_0 = [1011],$$
$$\mathbf{M}_1 = [0111],$$
$$\mathbf{M}_2 = [1100],$$
$$\mathbf{M}_3 = [0101].$$

Полученные подблоки данных имеют четырёхбитную размерность ($m = 4$), следовательно, для построения кода РС необходимо поле расширения GF(2^m) двоичного поля Галуа GF(2).

Определим характеристики кода РС:

$n = 2^m - 1 = 2^4 - 1 = 15$ — кодовая длина (количество подблоков блока данных, подлежащих защите, и подблоков с контрольной информацией);

$t = 2$ — количество исправляемых ошибок (обнаруживаемых и локализуемых подблоков блока данных с признаками нарушения целостности);

$k = n - 2t = 15 - 4 = 11$ — информационная длинна (количество подблоков блока данных, подлежащих защите);

$r = 2t = 4$ — количество контрольных символов (подблоков с контрольной информацией).

Опишем конечное поле с помощью примитивного полинома:

$$x^{15} - 1 = (1 + x) \cdot (1 + x + x^2) \times \times (1 + x + x^2 + x^3 + x^4) \cdot (1 + x + x^4) \times \times (1 + x^3 + x^4).$$

Проверим неприводимый полином:

$$f(x) = (1 + x + x^4)$$

на примитивность, то есть будет ли он делителем $x^{15} - 1$ для любого $n \in (1,15)$.

Так как $x^{15} - 1$ делится на $1 + x + x^4$, то повторив вычисления, можно проверить, что для любого $n \in (1,15)$ полином $x^n - 1$ не делится на полином $f(x)$, следовательно, полином $1 + x + x^4$ является примитивным.

Зададим правило понижения степени:

$$f(\alpha) = 0,$$

Тогда правило примет вид:

$$1 + \alpha + \alpha^4 = 0,$$

Вынесем примитивный элемент старшей степени за знак равно:

$\alpha^4 = 1 + \alpha$ (так как в двоичном поле $-1 = 1$).

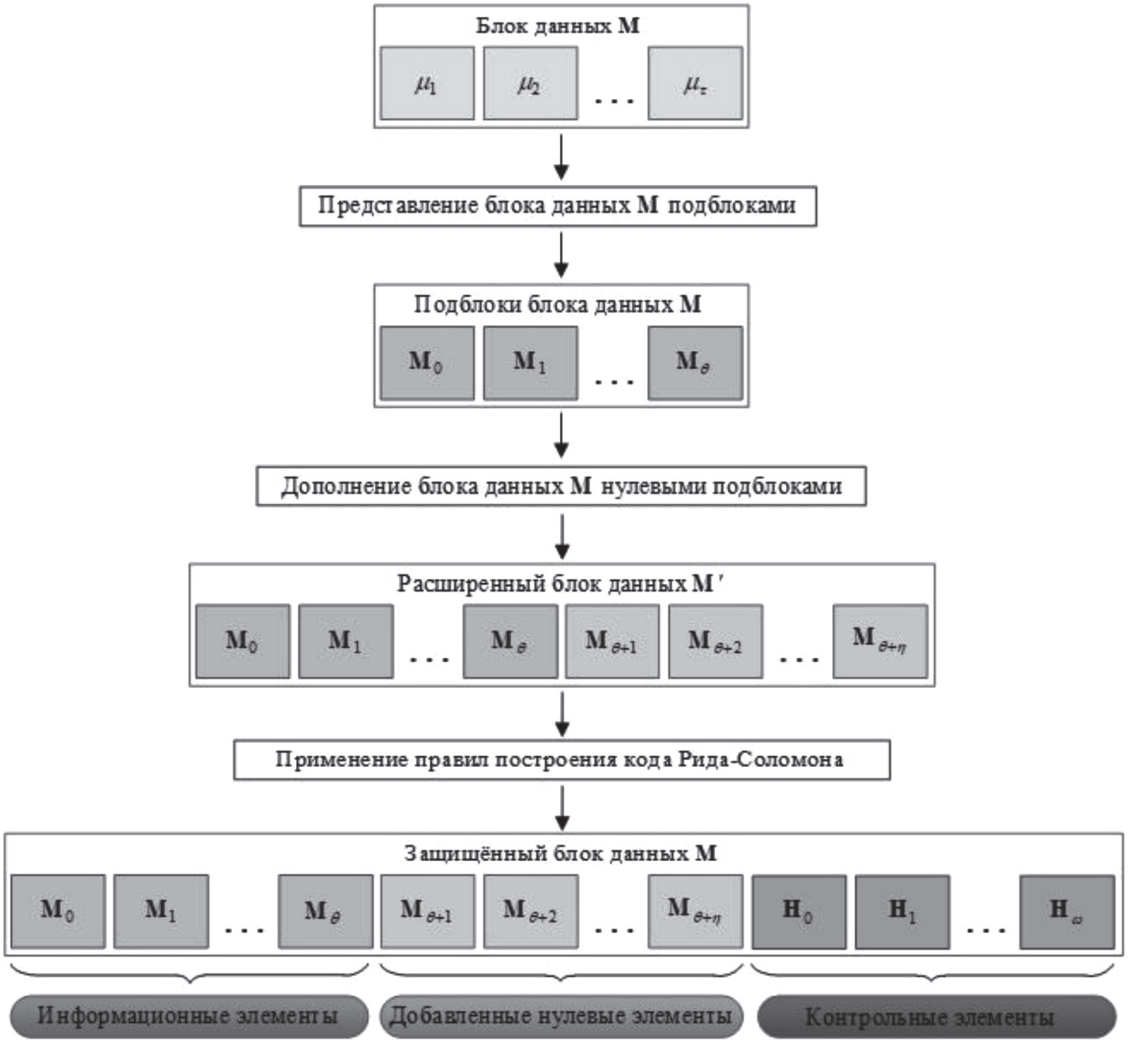


Рис. 1. Схема, иллюстрирующая порядок контроля целостности данных на основе правил построения кода Рида-Соломона

Fig. 1. Scheme illustrating the procedure for data integrity control based on the rules for constructing the Reed-Solomon code

Выразим примитивные элементы поля через базисные элементы α^0, α^1, α^2, α^3 и правило понижения степени.

Получим:

$$\alpha^0 = \alpha^0 \to (0001);$$
$$\alpha^1 = \alpha^1 \to (0010);$$
$$\alpha^2 = \alpha^2 \to (0100);$$
$$\alpha^3 = \alpha^3 \to (1000);$$
$$\alpha^4 = 1 + \alpha \to (1000);$$
$$\alpha^5 = \alpha^4\alpha = \alpha(1+\alpha) = \alpha + \alpha^2 \to (0110);$$
$$\alpha^6 = \alpha^5\alpha = \alpha(\alpha+\alpha^2) = \alpha^2 + \alpha^3 \to (1100);$$
$$\alpha^7 = \alpha^6\alpha = \alpha(\alpha^2+\alpha^3) = 1 + \alpha + \alpha^3 \to (1011);$$
$$\alpha^8 = \alpha^7\alpha = \alpha(1+\alpha+\alpha^3) = 1 + \alpha^2 \to (0101);$$
$$\alpha^9 = \alpha^8\alpha = \alpha(1+\alpha^2) = \alpha + \alpha^3 \to (1010);$$
$$\alpha^{10} = \alpha^9\alpha = \alpha(\alpha+\alpha^3) = 1 + \alpha + \alpha^2 \to (0111);$$
$$\alpha^{11} = \alpha^{10}\alpha = \alpha(1+\alpha+\alpha^2) =$$
$$= \alpha + \alpha^2 + \alpha^3 \to (1000);$$
$$\alpha^{12} = \alpha^{11}\alpha = \alpha(\alpha+\alpha^2+\alpha^3) =$$
$$= 1 + \alpha + \alpha^2 + \alpha^3 \to (1111);$$
$$\alpha^{13} = \alpha^{12}\alpha = \alpha(1+\alpha+\alpha^2+\alpha^3) =$$
$$= 1 + \alpha^2 + \alpha^3 \to (1101);$$

$$\alpha^{14} = \alpha^{13}\alpha = \alpha(1+\alpha^2+\alpha^3) = 1+\alpha^3 \to (1001);$$

$$0 \to (0000).$$

Кодирование.

Порождающий полином равен:

$$g(x) = (x-\alpha)\cdot(x-\alpha)\ldots(x-\alpha^{2t}) =$$

$$= x^4 + \alpha^{13}x^3 + \alpha^6x^2 + \alpha^3x + \alpha^{10}.$$

Запишем исходный блок данных **M** с помощью примитивных элементов:

$$\mathbf{M} = \left\{\alpha^7 \quad \alpha^{10} \quad \alpha^6 \quad \alpha^8\right\}.$$

В рассматриваемом (15,11)-коде РС информационная длина равна $k = 11$.

Так как блок данных **M**, подлежащий защите, имеет четыре элемента (подблока данных), дополним его $\eta = 11-3-1 = 7$ нулевыми элементами.

Получим:

$$\mathbf{M}' = \left\{\alpha^7\ \alpha^{10}\ \alpha^6\ \alpha^8\ 0\ 0\ 0\ 0\ 0\ 0\ 0\right\}.$$

Подблоки блока данных **M′** представим элементами GF(2^4):

$$u(x) = \alpha^7x^{10} + \alpha^{10}x^9 + \alpha^6x^8 + \alpha^8x^7.$$

Выполним несистематическое кодирование, где полином кодового слова вычисляется по формуле:

$$c(x) = u(x)\cdot g(x),$$

и равен:

$$c(x) = \alpha^7x^{14} + x^{13} + \alpha^2x^{12} + \alpha^4x^{11} +$$
$$+ \alpha^9x^{10} + \alpha^8x^9 + \alpha^6x^8 + \alpha^3x^7,$$

а соответствующий ему вектор:

$$c(x) =$$
$$= \left(\alpha^7, 1, \alpha^2, \alpha^4, \alpha^9, \alpha^8, \alpha^6, \alpha^3, 0, 0, 0, 0, 0, 0, 0\right).$$

Исходный блок данных **M** = [1, 0, 1, 1, 0, 1, 1, 1, 1, 1, 0, 0, 0, 1, 0, 1] преобразован посредством применения алгоритма кодирования РС и может быть отправлен на хранение.

При запросе данных для использования выполним декодирование.

Предположим, что была нарушена целостность данных, искажению подверглись следующие элементы кодового слова:

$$c = \left(\underline{\alpha^7}, 1, \alpha^2, \alpha^4, \underline{\alpha^9}, \alpha^8, \alpha^6, \alpha^3, 0, 0, 0, 0, 0, 0, 0\right).$$

Принятая комбинация определена формулой:

$$\upsilon = c + e.$$

Зададим вектор ошибки:

$$e = \left(\underline{\alpha^4}, 0, 0, 0, \underline{\alpha^9}, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\right),$$

тогда принятая кодовая комбинация:

$$\upsilon = \left(\underline{\alpha^3}, 1, \alpha^2, \alpha^4, \underline{0}, \alpha^8, \alpha^6, \alpha^3, 0, 0, 0, 0, 0, 0, 0\right).$$

При этом в первом случае произошла подмена элемента, во втором случае его удаление.

Перейдём к полиномиальной записи:

$$e(x) = \alpha^4x^{14} + \alpha^9x^{10};$$

$$\upsilon(x) = \alpha^3x^{14} + x^{13} + \alpha^2x^{12} + \alpha^4x^{11} + \alpha^8x^9 +$$
$$+ \alpha^6x^8 + \alpha^3x^7.$$

Вычислим компоненты синдрома S_ϖ в точках α^ϖ, $\varpi = 1, \ldots, 2t$ (дополнительно проверим компоненты синдрома S_ϖ по полиному ошибок $e(\alpha^\varpi)$):

$$S_1 = \upsilon(\alpha^1) = \alpha^7; \qquad S_1 = e(\alpha^1) = \alpha^7;$$
$$S_2 = \upsilon(\alpha^2) = \alpha^{13}; \qquad S_1 = e(\alpha^2) = \alpha^{13};$$
$$S_3 = \upsilon(\alpha^3) = \alpha^3; \qquad S_1 = e(\alpha^3) = \alpha^3;$$
$$S_4 = \upsilon(\alpha^4) = \alpha^1; \qquad S_4 = e(\alpha^4) = \alpha^1.$$

Определим число произошедших ошибок, предположим, что произошло $\nu = 2$ ошибки и сформируем матрицу A:

$$A = \begin{pmatrix} S_1 & S_2 \\ S_2 & S_3 \end{pmatrix} = \begin{pmatrix} \alpha^7 & \alpha^{13} \\ \alpha^{13} & \alpha^3 \end{pmatrix},$$

$$\det A = \begin{vmatrix} \alpha^7 & \alpha^{13} \\ \alpha^{13} & \alpha^3 \end{vmatrix} =$$

$$= \alpha^7\alpha^3 - \alpha^{13}\alpha^{13} = \alpha^{10} - \alpha^{11} = \alpha^{14} \neq 0,$$

значит, при хранении, действительно, произошло две ошибки.

Локализация ошибки выполняется путём поиска коэффициентов полинома Λ из системы:

$$\begin{pmatrix} S_1 & S_2 \\ S_2 & S_3 \end{pmatrix}\begin{pmatrix} \Lambda_2 \\ \Lambda_1 \end{pmatrix} = \begin{pmatrix} -S_3 \\ -S_4 \end{pmatrix};$$

$$\begin{pmatrix} \alpha^7 & \alpha^{13} \\ \alpha^{13} & \alpha^3 \end{pmatrix}\begin{pmatrix} \Lambda_2 \\ \Lambda_1 \end{pmatrix} = \begin{pmatrix} -\alpha^3 \\ -\alpha^1 \end{pmatrix};$$

$$A^{-1} = (\alpha^{14})^{-1}\begin{pmatrix} \alpha^3 & \alpha^{13} \\ \alpha^{13} & \alpha^7 \end{pmatrix} =$$

$$= \alpha^1\begin{pmatrix} \alpha^3 & \alpha^{13} \\ \alpha^{13} & \alpha^7 \end{pmatrix} = \begin{pmatrix} \alpha^4 & \alpha^{14} \\ \alpha^{14} & \alpha^8 \end{pmatrix};$$

$$\begin{pmatrix} \Lambda_2 \\ \Lambda_1 \end{pmatrix} = \begin{pmatrix} \alpha^4 & \alpha^{14} \\ \alpha^{14} & \alpha^8 \end{pmatrix}\begin{pmatrix} \alpha^3 \\ \alpha^1 \end{pmatrix} = \begin{pmatrix} \alpha^7 + \alpha^0 \\ \alpha^2 + \alpha^9 \end{pmatrix} = \begin{pmatrix} \alpha^9 \\ \alpha^{11} \end{pmatrix}.$$

Полином равен:

$$\Lambda(x) = \alpha^0 + \Lambda_1 X + \Lambda_2 X^2 = 1 + \alpha^{11} X + \alpha^9 X^2.$$

Подставим все степени α и найдём корни многочлена L(x):

$$\Lambda(\alpha^0) = 1 + \alpha^{11}\alpha^0 + \alpha^9\alpha^0 = \alpha^8;$$

$\Lambda(\alpha^1) = 1 + \alpha^{11}\alpha^1 + \alpha^9\alpha^2 = 0 \Rightarrow$ произошла ошибка;

$$\Lambda(\alpha^2) = 1 + \alpha^{11}\alpha^2 + \alpha^9\alpha^4 = \alpha^0;$$
$$\Lambda(\alpha^3) = 1 + \alpha^{11}\alpha^3 + \alpha^9\alpha^6 = \alpha^{14};$$
$$\Lambda(\alpha^4) = 1 + \alpha^{11}\alpha^4 + \alpha^9\alpha^8 = \alpha^2;$$

$\Lambda(\alpha^5) = 1 + \alpha^{11}\alpha^5 + \alpha^9\alpha^{10} = 0 \Rightarrow$ произошла ошибка;

$$\Lambda(\alpha^6) = 1 + \alpha^{11}\alpha^6 + \alpha^9\alpha^{12} = \alpha^{14};$$
$$\Lambda(\alpha^7) = 1 + \alpha^{11}\alpha^7 + \alpha^9\alpha^{14} = \alpha^6;$$
$$\Lambda(\alpha^8) = 1 + \alpha^{11}\alpha^8 + \alpha^9\alpha^{16} = \alpha^8;$$
$$\Lambda(\alpha^9) = 1 + \alpha^{11}\alpha^9 + \alpha^9\alpha^{18} = \alpha^3;$$
$$\Lambda(\alpha^{10}) = 1 + \alpha^{11}\alpha^{10} + \alpha^9\alpha^2 = \alpha^2;$$
$$\Lambda(\alpha^{11}) = 1 + \alpha^{11}\alpha^{11} + \alpha^9\alpha^{22} = \alpha^3;$$
$$\Lambda(\alpha^{12}) = 1 + \alpha^{11}\alpha^{12} + \alpha^9\alpha^{24} = \alpha^6;$$
$$\Lambda(\alpha^{13}) = 1 + \alpha^{11}\alpha^{13} + \alpha^9\alpha^{26} = \alpha^{13};$$
$$\Lambda(\alpha^{14}) = 1 + \alpha^{11}\alpha^{14} + \alpha^9\alpha^{28} = \alpha^{13}.$$

Расположение ошибок является обратной величиной к корням многочлена $\Lambda(x)$, то есть корень получаем при $X^{-1} = \alpha$.

Отсюда локаторы ошибок равны $X_1^{-1} = (\alpha^1)^{-1} = \alpha^{14}$ и $X_2^{-1} = (\alpha^5)^{-1} = \alpha^{10}$ и совпадают с позициями заданных ошибок $e(x) = \alpha^4 \underline{x^{14}} + \alpha^9 \underline{x^{10}}$.

Таким образом, были найдены позиции ошибок (подблоки данных с признаками нарушения целостности).

Заключение

Представлен подход к контролю целостности многомерных массивов данных, основанный на применении правил построении кодов РС. Разработанный способ позволяет обеспечить контроль целостности данных с возможностью локализации блоков данных с признаками нарушения целостности при одновременном сокращении объёма вводимой избыточности по сравнению с существующими решениями. Использование представленных механизмов защиты позволяет повысить работоспособность СХД в современных условиях их функционирования.

СПИСОК ИСТОЧНИКОВ

1. **Omondi A.** Residue Number System: Theory and Implementation. Imperial Collegt Press, London, 2007.— 296 p.

2. **Schneier B.** Applied Cryptography Second Edition: protocols, algorithms and source code in C. John Wiley & Sons, Inc, 2016.— 653 p.

3. **Knuth D. E**. The Art of Computer Programming: Volume 3: Sorting and Searching. 2nd edn, 2018.— 803 p.

4. **Шевцов В. Ю**. Анализ современных систем хранения данных / В. Ю. Шевцов, Е. С. Абрамов // НБИ технологии.— 2019.— Т. 13. № 1.— С. 25–30.

5. **Борзенкова С. Ю**. Обеспечение безопасности системы хранения данных / С. Ю. Борзенкова, И. В. Савин // Известия ТулГУ. Технические науки.— 2017.— № 10.— С. 196–200.

6. **Диченко С. А**. Контроль и восстановление целостности многомерных массивов данных посредством криптокодовых конструкций / С. А. Диченко, О. А. Финько // Программирование.— 2021.— № 6.— С. 3–15.

7. **Диченко С. А**. Концептуальная модель обеспечения целостности информации в современных системах хранения данных / С. А. Диченко // В сборнике: Информатика:

проблемы, методология, технологии. Сборник материалов XIX международной научно-методической конференции. Под ред. Д. Н. Борисова.— 2019.— С. 697–701.

8. **Клеменков П. А**. Большие данные: современные подходы к хранению и обработке / П. А. Клеменков, С. Д. Кузнецов // Труды Института системного программирования РАН.— 2012.— Т. 23.— С. 143–158.

9. **Ямашкин С. А**. Интеграция, хранение и обработка больших массивов пространственно-временной информации в цифровых инфраструктурах пространственных данных / С. А. Ямашкин, А. А. Ямашкин // Современные наукоемкие технологии.— 2021.— № 5.— С. 108–113.

10. **Tchernykh A.** AC-RRNS: Anti-Collusion Secured Data Sharing Scheme for Cloud Storage / A. Tchernykh, M. Babenko, N. Chervyakov // International Journal of Approximate Reasoning. Special Issue on Uncertainty in Cloud Computing: Concepts, Challenges and C. Solutions.— 2018.— Vol. 102. P.— 60–73.

11. **Диченко С. А**. Модель контроля целостности многомерных массивов данных / С. А. Диченко // Проблемы информационной безопасности. Компьютерные системы.— 2021.— № 2(46).— С. 97–103.

12. **Диченко С. А.** Снижение вводимой избыточности при обеспечении устойчивости информационно-аналитических систем в условиях компенсации последствий деструктивных воздействий злоумышленника / С. А. Диченко, О. А. Финько // Автоматизация процессов управления.— 2020.— № 4 (62).— С. 38–48.

13. **Скляр Б.** Цифровая связь. Теоретические основы и практическое применение. М.: Вильямс, 2016.— 1099 с.

14. **Wang X.** How to break MD5 and Other Hash Function. EUROCRYPT. LNCS3494.— Springer-Verlag, 2005.— P. 19–35.

15. **Biham E.** A framework for iterative hash functions. HAIFA. ePrint Archive, Report 2007/278. 2007.— P. 1–20.

16. **Bellare M.** New Proofs for NMAC and HMAC: Security without Collision-Resistance. CRYPTO. ePrint Archive, Report 2006/043. 2006.— P. 1–16.

17. **Диченко С. А**. Контроль и восстановление целостности данных в защищенных информационно-аналитических системах / С. А. Диченко, О. А. Финько // Труды Военно-космической академии имени А. Ф. Можайского.— 2021.— № 676.— С. 36–49.

18. **Tilborg H.** Fundamentals of cryptology: a professional reference and interactive tutorial. Kluwer Academic Publishers. London, 2014.— 414 p.

19. **Диченко С. А**. Реализация двоичных псевдослучайных последовательностей линейными числовыми полиномами / С. А. Диченко, А. К. Вишневский, О. А. Финько // Известия ЮФУ. Технические науки.— 2011.— № 12 (125).— С. 130–140.

20. **Тимофеев Г. С**. Аппаратная реализация кодирования информации систематическими полярными кодами / Г. С. Тимофеев // Вестник Сибирского государственного аэрокосмического университета им. академика М. Ф. Решетнева.— 2017.— Т. 18.— № 1.— С. 97–104.

REFERENCES

1. **Omondi A.** Residue Number System: Theory and Implementation. Imperial Collegt Press, London, 2007.— 296 p.

2. **Schneier B.** Applied Cryptography Second Edition: protocols, algorithms and source code in C. John Wiley & Sons, Inc, 2016.— 653 p.

3. **Knuth D. E**. The Art of Computer Programming: Volume 3: Sorting and Searching. 2nd edn, 2018.— 803 p.

4. **Shevcov V. Yu**. Analiz sovremennyh sistem hraneniya dannyh / V.YU.SHevcov, E. S. Abramov [Analysis of modern data storage systems] // NBI tekhnologii.— 2019.— T. 13. № 1.— S. 25–30. (In Russian)

5. **Borzenkova S. Yu**. Obespechenie bezopasnosti sistemy hraneniya dannyh [Securing the storage system] / S. Yu. Borzenkova, I. V. Savin // Izvestiya TulGU. Tekhnicheskie nauki.— 2017.— No 10.— P. 196–200. (In Russian)

6. **Dichenko S. A**. Kontrol' i vosstanovlenie celostnosti mnogomernyh massivov dannyh posredstvom kriptokodovyh konstrukcij [Control and restoration of the integrity of multidimensional data arrays using cryptocode structures] / S. A. Dichenko, O. A. Fin'ko // Programmirovanie.— 2021.— No 6.— P. 3–15. (In Russian)

7. **Dichenko S. A**. Konceptual'naya model' obespecheniya celostnosti informacii v sovremennyh sistemah hraneniya dannyh [Conceptual model for ensuring the integrity of information in modern data storage systems] / S. A. Dichenko // V sbornike: Informatika: problemy, metodologiya, tekhnologii.

Sbornik materialov XIX mezhdunarodnoj nauchno-metodicheskoj konferencii. Pod red. D. N. Borisova.— 2019.— P. 697–701. (In Russian)

8. **Klemenkov P. A**. Bol'shie dannye: sovremennye podhody k hraneniyu i obrabotke [Big data: modern approaches to storage and processing] / P. A. Klemenkov, S. D. Kuznecov // Trudy Instituta sistemnogo programmirovaniya RAN.— 2012.— Vol. 23.— P. 143–158. (In Russian)

9. **Yamashkin S. A.** Integraciya, hranenie i obrabotka bol'shih massivov prostranstvenno-vremennoj informacii v cifrovyh infrastrukturah prostranstvennyh dannyh [Integration, storage and processing of large arrays of spatio-temporal information in digital infrastructures of spatial data] / S. A. Yamashkin, A. A. Yamashkin // Sovremennye naukoemkie tekhnologii.— 2021.— No 5.— P. 108–113. (In Russian)

10. **Tchernykh A.** AC-RRNS: Anti-Collusion Secured Data Sharing Scheme for Cloud Storage / A. Tchernykh, M. Babenko, N. Chervyakov // International Journal of Approximate Reasoning. Special Issue on Uncertainty in Cloud Computing: Concepts, Challenges and C. Solutions.— 2018.— Vol. 102. P.— 60–73.

11. **Dichenko S. A**. Model' kontrolya celostnosti mnogomernyh massivov dannyh [Integrity Control Model for Multidimensional Data Arrays]/ S. A. Dichenko // Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy.— 2021.— No 2(46).— P. 97–103. (In Russian)

12. **Dichenko S. A**. Snizhenie vvodimoj izbytochnosti pri obespechenii ustojchivosti informacionno-analiticheskih sistem v usloviyah kompensacii posledstvij destruktivnyh vozdejstvij zloumyshlennika [Reducing the input redundancy while ensuring the stability of information and analytical systems in terms of compensating the consequences of the destructive effects of an intruder] / S. A. Dichenko, O. A. Fin'ko // Avtomatizaciya processov upravleniya.— 2020.— No 4 (62).— P. 38–48. (In Russian)

13. **Sklyar B.** Cifrovaya svyaz'. Teoreticheskie osnovy i prakticheskoe primenenie. [Digital communication. Theoretical foundations and practical application] M.: Vil'yams, 2016.— 1099 p. (In Russian)

14. **Wang X.** How to break MD5 and Other Hash Function. EUROCRYPT. LNCS3494.— Springer-Verlag, 2005.— P. 19–35.

15. **Biham E.** A framework for iterative hash functions. HAIFA. ePrint Archive, Report 2007/278. 2007.— P. 1–20.

16. **Bellare M.** New Proofs for NMAC and HMAC: Security without Collision-Resistance. CRYPTO. ePrint Archive, Report 2006/043. 2006.— P. 1–16.

17. **Dichenko S. A**. Kontrol' i vosstanovlenie celostnosti dannyh v zashchishchennyh informacionno-analiticheskih sistemah [Data integrity control and restoration in secure information and analytical systems] / S. A. Dichenko, O. A. Fin'ko // Trudy Voenno-kosmicheskoj akademii imeni A. F. Mozhajskogo.— 2021.— No 676.— P. 36–49. (In Russian)

18. **Tilborg H.** Fundamentals of cryptology: a professional reference and interactive tutorial. Kluwer Academic Publishers. London, 2014.— 414 p.

19. **Dichenko S. A**. Realizaciya dvoichnyh psevdosluchajnyh posledovatel'nostej linejnymi chislovymi polinomami [Implementation of Binary Pseudo-Random Sequences by Linear Numeric Polynomials] / S. A. Dichenko, A. K. Vishnevskij, O. A. Fin'ko // Izvestiya YUFU. Tekhnicheskie nauki.— 2011.— No 12 (125).— P. 130–140. (In Russian)

20. **Timofeev G. S**. Apparatnaya realizaciya kodirovaniya informacii sistematicheskimi polyarnymi kodami [Hardware implementation of information encoding by systematic polar codes] / G. S. Timofeev // Vestnik Sibirskogo gosudarstvennogo aerokosmicheskogo universiteta im. akademika M. F. Reshetneva.— 2017.— Vol. 18.— No 1.— P. 97–104. (In Russian)

Статья поступила в редакцию 07.02.2022; одобрена после рецензирования 16.02.2022; принята к публикации 22.02.2022.

The article was submitted 07.02.2022; approved after reviewing 16.02.2022; accepted for publication 22.02.2022.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

СТАРИКОВ Тимофей Владимирович — адъюнкт; Краснодарское высшее военное училище имени генерала армии С. М. Штеменко

STARIKOV Timofey V.— adjunct; Krasnodar Higher Military School named after General of the Army S. M. Shtemenko

Email: tvs_7@mail.ru

ORCID: 0000-0001-5895-5385

СОПИН Кирилл Юрьевич — адъюнкт; Краснодарское высшее военное училище имени генерала армии С. М. Штеменко

ДИЧЕНКО Сергей Александрович — к.т.н., докторант; Краснодарское высшее военное училище имени генерала армии С. М. Штеменко

САМОЙЛЕНКО Дмитрий Владимирович — д.т.н.; начальник кафедры, Краснодарское высшее военное училище имени генерала армии С. М. Штеменко

SOPIN Kirill Yu. — adjunct; Krasnodar Higher Military School named after General of the Army S. M. Shtemenko
Email: sopin.kirill2010@yandex.ru
ORCID: 0000-0002-0546-7801

DICHENKO Sergey A. — Ph.D., doctoral student; Krasnodar Higher Military School named after General of the Army S. M. Shtemenko
Email: dichenko.sa@yandex.ru
ORCID: 0000-0002-0644-4353

SAMOILENKO Dmitry V. — Doctor of Technical Sciences; Head of Department, Krasnodar Higher Military School named after General of the Army S. M. Shtemenko
Email: 19sam@mail.ru
ORCID: 0000-0001-9665-2174

Научная статья
DOI 10.48612/jisp/f7z9-ttke-gv81
УДК 004.932.2

Д. Э. Вильховский
Омский государственный университет им. Ф. М. Достоевского
644053, Российская Федерация, г. Омск

МЕТОД ОБНАРУЖЕНИЯ LSB-ВСТАВОК В ЦВЕТНЫХ ФОТОГРАФИЧЕСКИХ ИЗОБРАЖЕНИЯХ С НИЗКИМ ЗАПОЛНЕНИЕМ СТЕГОКОНТЕЙНЕРА

Аннотация. В статье представлен низкоразмерный метод обнаружения LSB-вставок в цветных фотографических изображениях, показывающий высокую эффективность при работе со стего-изображениями, характеризующимися низким уровнем стегонагрузки. Метод основан на анализе признаков попарного сходства между нулевым и первым слоями, решении задачи о наибольшем пустом прямоугольнике, анализе моделей доминирования белых (черных) пикселей и моментов изображения.

Ключевые слова: Стегоанализ, стеганографический анализ, анализ стегоконтейнера, обнаружение LSB-вставки

Research article
DOI 10.48612/jisp/f7z9-ttke-gv81

D. E. Vilkhovsky
Dostoevsky Omsk State University, Russia, Omsk

METHOD OF DETECTING LSB INSERTS IN LOW STEGO-PAYLOAD COLOR PHOTOGRAPHIC IMAGES

Abstract. The paper presents a compact method of detecting LSB inserts in color photographic images that proves high efficiency when dealing with low stego-payload images. The method is based on an analysis of signatures of pairwise similarity the zero and first layers, an algorithm for largest empty rectangles, white (black) pixel dominance pattern and image moments analysis.

Keywords: Steganalysis, steganographic analysis, stegocontainer analysis, LSB-insert detection.

Задача установления факта встраивания в цветных фотографических изображениях, выполненного методом замены наименее значащего бита (LSB-встраивание), является одной из наиболее сложных при проведении стеганографического анализа изображений. При этом популярность анализируемого метода встраивания, а также развитие социальных и иных сетей, через которые в больших объемах передаются именно фотографические изображения, определяет актуальность поставленной задачи. Одним из усложняющих факторов является низкий уровень заполнения стегоконтейнера, когда злоумышленники производят встраивание в область, составляющую не более 40 % от общего объема изображения (т.е. уровень заполнения стегоконтейнера не превышает 40 %).

Некоторые из активно разрабатываемых в последние 5–8 лет алгоритмов стегоанализа с использованием сверхточных нейронных сетей позволяют достаточно успешно выявлять вставки, выполненные методом замены наименее значащего бита (LSB-вставки) [2, 3, 4, 7], однако слишком ресурсоемки для широкого распространения и активного использования.

Поэтому была предпринята попытка разработки метода обнаружения LSB-вставок в фотографических цветных изображениях со стего-нагрузкой на уровне 25–40 %.

В общем виде, разработанный алгоритм обнаружения LSB-вставок в цветных фотографических изображениях с низким заполнением стегоконтейнера состоит из трех этапов:

Этап 1. Определение условно чистых участков изображения, т.е. участков, которые точно не содержат встраивание, посредством нахождения сверхсильных взаимосвязей между нулевым и первым слоями.

Этап 2. Установление максимальных границ области предполагаемого встраивания с использованием задачи о наибольшем пустом прямоугольнике.

Этап 3. Верификация выделенной области на предмет наличия встраивания посредством анализа моделей доминирования белых (черных) пикселей в нулевом и первом слоях.

В качестве анализируемого фотографического изображения, применяемого для визуализации работы алгоритма и получаемых результатов, используем изображение, а также карту пикселей его нулевого и первого слоев, приведенные на рисунке 1.

1. Определение условно чистых участков изображения

Решаемая на данном этапе задача — отсечение тех областей изображения, которые однозначно не содержат встраивание (чистых областей), что позволит сузить область дальнейшего исследования.

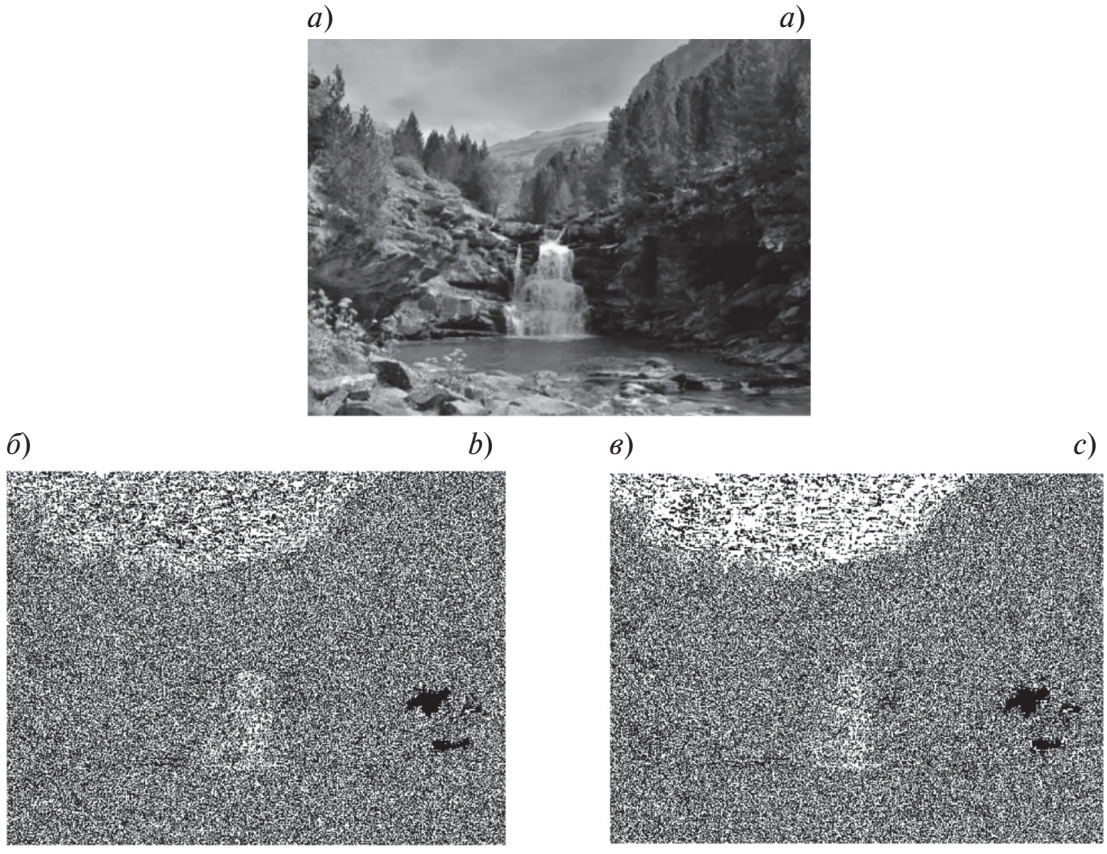


Рис. 1. Анализируемое фотографическое изображение: *а* — анализируемое фотографическое изображение; *б* — карта пикселей нулевого слоя анализируемого фотографического изображения; *в* — карта пикселей первого слоя анализируемого фотографического изображения

Fig. 1. Analyzed photographic image: *a* — analyzed photographic image; *b* — zero layer bit map for the analyzed photographic image; *c* — first layer bit map for the analyzed photographic image

Алгоритм определения чистых областей изображения основывается на гипотезе, что часть областей нулевого и первого слоев фотографического изображения обнаруживает определенное сходство. При этом, чем большее количество раз комбинации нулевого слоя повторяются в первом слое, тем сильнее взаимосвязь этих слоев. Таким образом, необходимо установить области, содержащие сверхсильные взаимосвязи между нулевым и первым слоями.

Алгоритм установления условно чистых участков изображения следующий:

1. Осуществляется разбивка первого и нулевого слоев изображения на блоки размерностью 10. Размерность блока определена эмпирически как показывающая наилучшие результаты для целей анализа

2. В каждом блоке определяется попарное сходство для каждого из пикселей. С этой целью производится последовательное сравнение комбинаций, которые формирует анализируемый пиксель с его правым, нижним, и угловым соседом в нулевом слое с соответствующими комбинациями соответствующих пикселей, обнаруживаемых в первом слое.

3. По каждому блоку рассчитывается средний удельный вес попарного сходства $\overline{\partial_i}$ (1):

$$\overline{\partial_i} = \frac{k_i}{3 \times P_i}, \qquad (1)$$

где k_i — количество попарных совпадений в i-ом блоке, пиксели; P_i — площадь i-ого блока, пиксели.

Полученные данные можно представить в виде точечной диаграммы, представленной на рисунке 2. Для большей наглядности, маркер для блоков, содержащих встраивание, имеет красный цвет.

Анализ множества изображений по признаку попарного сходства позволил сделать вывод о том, что в блоках со встраиванием могут наблюдаться комбинации, повторяющиеся в нулевом и первом слое, однако 97 % таких блоков не показывают удельный вес повторяющихся комбинаций выше 0,2.

4. Из полученной генеральной совокупности берется выборка по признаку $\overline{\partial_i} < 0,2$, и определяется граница областей потенциальных вставок ω по формуле (2):

$$\omega = \overline{X} + S, \qquad (2)$$

где $\overline{X}$ — среднее арифметическое значение по выборке; S — среднеквадратичное отклонение значений по выборке.

Если $\overline{\partial_i} > \omega$ — блок классифицируется как условно чистый и ему присваивается значение 1. Иначе — блок считается условным стего с бинарным значением 0.

2. Установление максимальных границ предполагаемой области встраивания

2.1 Общая постановка задачи

Для определения потенциальной области прямоугольного встраивания отлично подходит задача о наибольшем пустом прямоугольнике, описанная в работах А. Наамад, Д. Ли

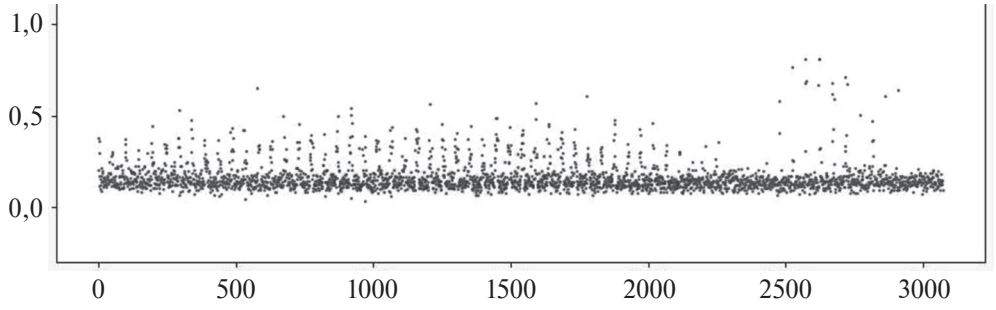


Рис. 2. Удельный вес попарного сходства по блокам для анализируемого изображения

Fig. 2. Specific gravity of pairwise similarity by blocks for the analyzed image

и В. Шу и других авторов [1, 5, 6]. Однако, для успешного решения задачи определения максимально возможной области, потенциально содержащей встраивание необходимо преодолеть следующее ограничение.

Представим результат классификации, проведенной на предыдущем этапе, в виде карты блоков изображения (рис. 3), где условно чистые блоки имеют заливку зеленого цвета, а условно стего-блоки имеют заливку бежевого цвета.

Из рисунка 3 можно сделать вывод о наличии шумов, которые сужают область исследования. Для устранения этих шумов был разработан специальный фильтр.

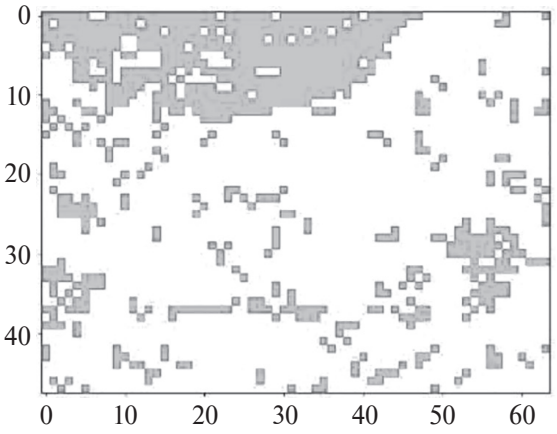


Рис. 3. Карта блоков изображения

Fig. 3. Image block map

2.2 Предварительная обработка карты блоков рекурсивным фильтром

Как видно на рисунке 3, все зеленые блоки можно разделить на 2 типа:

- блоки, формирующие кластеры;
- блоки, не формирующие кластеры

Пример блоков первого и второго типа представлен на рисунке 4.

Можно сделать вывод, что блоки первого типа — это суперсильные комбинации, поскольку послойное сходство наблюдается сразу у нескольких соседствующих блоков. При этом блоки второго типа не обнаруживают подобных комбинаций, а, следовательно, могут являться случайными шумами.

Таким образом, задача фильтра — кластеризация близлежащих блоков, что позволит определить участки с сильной зависимостью и устранить шумы.

Основное условие кластеризации, выдвинутое нами для решения указанных выше задач фильтрации шумов: кластер формируется тогда и только тогда, когда он содержит группу чистых блоков, размер которой составляет не менее 2x2 блока.

Алгоритм рекурсивного фильтра имеет вид:

1. Случайным образом выбирается точка входа — любой *i*-ый блок, классифицированный на предыдущем этапе как условно чистый.

2. Анализируются 8 близлежащих соседей выбранного *i*-го блока:

- если ни один из соседствующих блоков не является условно чистым, анализируемый блок классифицируется как условно-стего;
- если как минимум три соседствующих блока являются условно чистыми и при этом

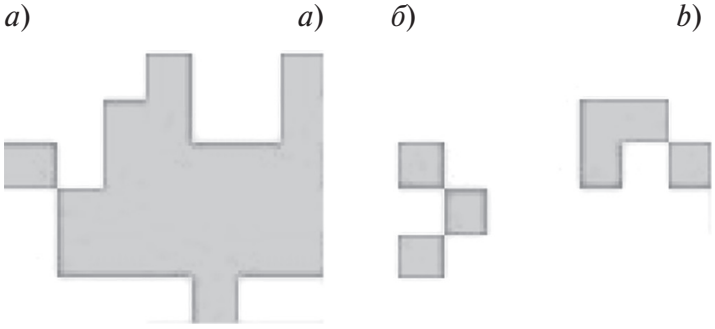


Рис. 4. Пример блоков первого типа (*а*) и второго типа (*б*)

Fig. 4. An example of blocks of the first type (*a*) and the second type (*b*)

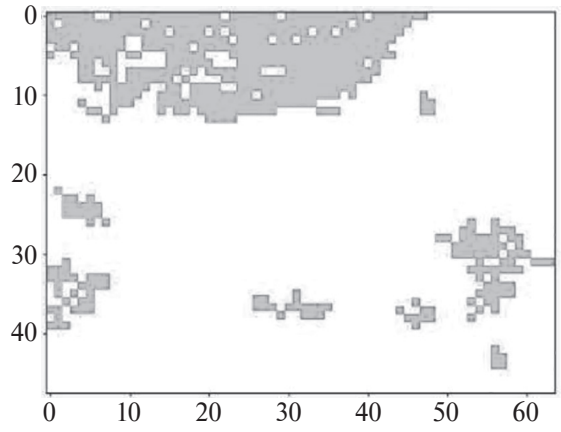


Рис. 5. Карта блоков изображения после обработки фильтром

Fig. 5. Image block map after filter processing

соседствующими между собой, то все четыре блока верифицируются как чистые, и отмечаются как блоки, входящие в новосформированный кластер. При этом поочередно для каждого из соседствующих блоков кластера запускается рекурсивная функция.

На рисунке 5 представлена карта блоков изображения после обработки фильтром.

2.3 Решение задачи о наибольшем пустом прямоугольнике и определение области предполагаемого встраивания

Алгоритм определения области предполагаемого встраивания на основе решения задачи о наибольшем пустом прямоугольнике состоит из двух шагов:

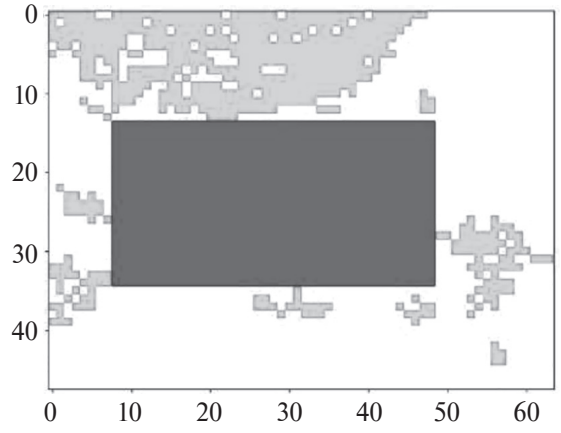


Рис. 6. Предполагаемая область встраивания

Fig. 6. Assumed embedding area

1. Установление границ наибольшего прямоугольника, заключенного с каждого из его сторон между кластерами блоков с суперсильной зависимостью, т.е. кластерами чистых блоков.

2. Верификация площади найденного прямоугольника на предмет соответствия условиям минимального уровня стегонагрузки.

Ранее мы указали, что разрабатываемый алгоритм должен быть жизнеспособен для работы с изображениями, уровень заполнения стегоконтейнера в которых составляет 25 % и более. Таким образом, верификацию площади найденного прямоугольника следует производить путем проверки его на выполнение следующего условия (формула 3):

$$P_1 \geq 0{,}25P_0, \qquad (3)$$

где P_1 — площадь найденного прямоугольника, пиксели; P_0 — площадь всего изображения, пиксели.

Если $P_1 \geq 0{,}25P_0$, то анализируемое изображение классифицируется как условное стего, а найденный прямоугольник считается областью предполагаемого встраивания. В ином случае изображение классифицируется как чистое; анализ изображения завершен.

Для анализируемого изображения решение задачи наибольшего пустого прямоугольника и верификации его площади позволяет классифицировать изображение как стего, с областью возможного встраивания, представленной на рисунке 6.

3. Окончательная верификация выделенной области на предмет наличия встраивания

Для окончательной верификации выделенной области на предмет наличия встраивания проводится анализ выделенной области по признаку доминирования белых (черных) пикселей.

1) По каждому пикселю нулевого и первого слоев проводится последовательное исследование количества белых пикселей с окном 3×3 и центральным анализируемым пикселем (блок вхождения) и формируется массив данных для нулевого и первого слоев.

Графическое представление данных о количестве белых пикселей в блоке вхождения отображено на рисунке 7. На рисунке по оси OX — порядковый номер пикселя, для которого рассматривается блок вхождения с окном 3×3, а по оси OY — количество белых пикселей в данном блоке.

Изучение данных о количестве белых пикселей в нулевом и первом слоях изображения без встраивания, позволяет сделать вывод о наличии определенного сходства по признаку доминирования белых (черных) пикселей. Т.е. в 80 % случаев комбинации, содержащие только белые (количество белых пикселей в блоке = 9) или только черные (количество белых пикселей в блоке = 0) пиксели, а также комбинации, содержащие 8 или 1 пиксель белого цвета, переносятся между слоями с незначительными отклонениями, не превышающими 3 единиц.

2) Для каждого пикселя предполагаемой области встраивания, образующего блок с количеством белых пикселей 0, 1, 8, или 9, производится сравнение моделей доминирования в нулевом (модель m_i^0) и первом слоях (модель m_i^1):

А) если $m_i^1 = m_i^0$, блоки вхождения являются идентичными, а анализируемый пиксель считается неизмененным пикселем.

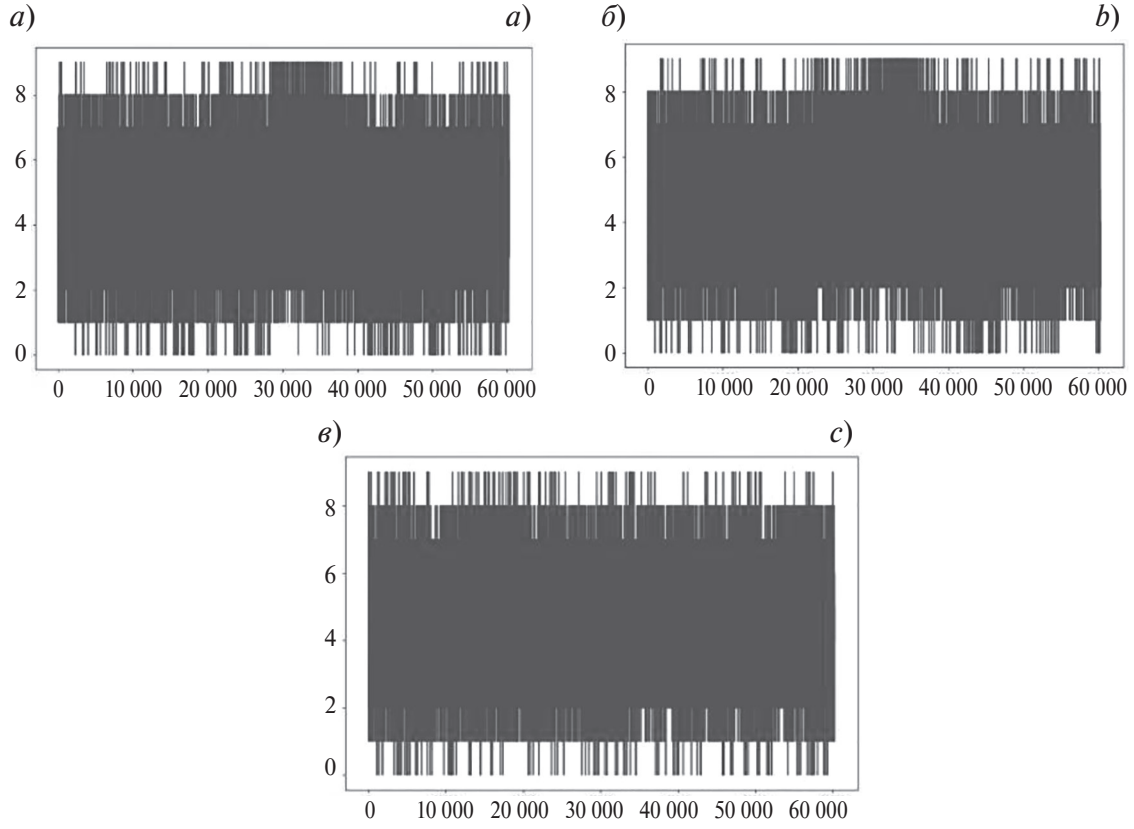


Рис. 7. Диаграммы, построенные с использованием моделей доминирования белых пикселей: *а* — количество белых пикселей в блоке вхождения для нулевого слоя участка изображения без встраивания; *б* — количество белых пикселей в блоке вхождения для первого слоя участка изображения; *в* — количество белых пикселей в блоке вхождения для нулевого слоя участка изображения со встраиванием

Fig. 7. Diagrams using the dominance feature of white (black) pixels: *a* — the number of white pixels in the block of entry for the zero layer of the image area without embedding; *b* — the number of white pixels in the block of occurrence for the first layer of the image area; *c* — the number of white pixels in the block of entry for the zero layer of the image area with embedding

Б) иначе — считаем величину отклонения по выделенному признаку:

• если $\left(m_i^1 = m_i^0\right) < 3$ — изменения считаются несущественными, блоки вхождения определяются как схожие, а анализируемый пиксель классифицируется как неизмененный;

• иначе — изменения считаются существенными, блоки вхождения определяются как несхожие, а анализируемый пиксель классифицируется как измененный.

Карта пикселей с существенными изменениями между слоями в отношении моделей доминирования белых (черных) пикселей для анализируемого изображения, представлена на рисунке 8.

3) Производится расчет моментов изображения *M* для полученной карты пикселей.

Проведенный анализ данных, полученных на множестве фотографических изображений, позволил установить, что пиксели с подобными существенными изменениями присутствуют не только в стего-, но также и в чистых изображениях. Однако, в последних они распределены в достаточной степени равномерно, а моменты изображения *M* имеют координаты, близкие к координатам центра изображения.

При этом в стего-изображениях наблюдается концентрация изменений моделей доминирования в области встраивания, а момент изображения *M* отклоняется от центральных координат изображения. Например, для анализируемого нами примера, момент изображения для участка, содержащего встраивание, представлен на рисунке 9.

Таким образом, окончательная верификация выделенной области на предмет наличия встраивания имеет следующий вид:

• если координаты момента *M* совпадают с координатами центра анализируемой области, область верифицируется как не содержащая встраивание, а анализируемое изображение классифицируется как чистое;

• иначе — область верифицируется как содержащая встраивание, а анализируемое изображение классифицируется как стего.

Заключение

Тестирование разработанного и представленного метода обнаружения LSB-вставок на коллекции из 500 цветных фотографических изображений с уровнем заполнения стегоконтейнера от 25 до 40 % показало высокую эффективность при одновременно высоком быстродействии.

Так, эффективность обнаружения составила 81,7 % при 40 %-м уровне заполнения стегоконтейнера и 73,1 % при 25 %-м уровне заполнения стегоконтейнера. При этом обработка изображения и его анализ

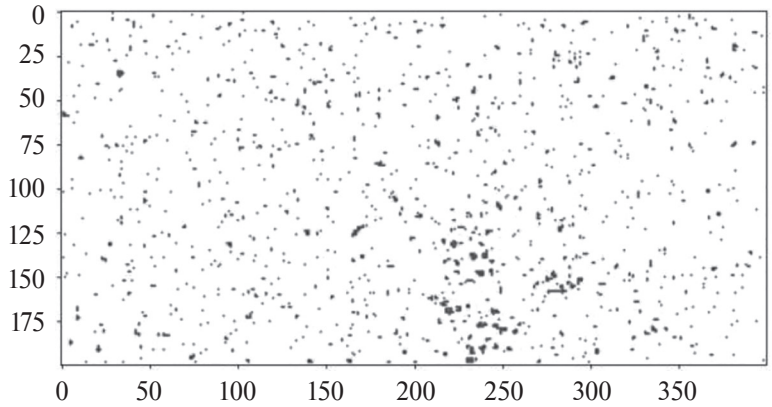


Рис. 8. Карта пикселей с существенными изменениями между слоями в отношении моделей доминирования белых (черных) пикселей

Fig. 8. Map of pixels with significant white (black) pixel dominance pattern changes between the layers

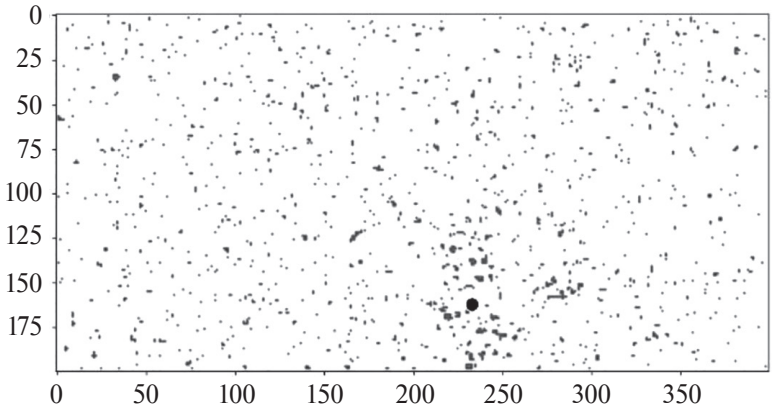


Рис. 9. Момент изображения для участка, содержащего встраивание

Fig. 9. Image moment for an area with embedding

на предмет наличия встраивания занимал не более нескольких секунд, что говорит о возможности обработки большого количества изображений за заданное время. Это подтверждает жизнеспособность метода обнаружения LSB-вставок в цветных фотографических изображениях и возможности его широкого применения даже среди организаций, не располагающих значительными ресурсами вычислительных мощностей.

СПИСОК ИСТОЧНИКОВ

1. **Acharyya A, De M., Subhas C., Pandit S**. Variations of largest rectangle recognition amidst a bichromatic point set. Discrete Applied Mathematics. 2020. Vol. 286. P. 35–50.

2. **Chaumont M.** Deep learning in steganography and steganalysis. In Digital Media Steganography, Academic Press, 2020. P. 321–349.

3. **Chen M, Boroumand M, Fridrich J.** Deep learning regressors for quantitative steganalysis. Electron Imaging 2018, 7. P. 160–161.

4. **Cogranne R., Giboulot Q., Bas P.** The ALASKA steganalysis challenge: A first step towards steganalysis," in Proceedings of the ACM Workshop on Information Hiding and Multimedia Security, 2019. P. 125–137.

5. **Naamad A., Lee D. T., Hsu W.-L.** On the Maximum Empty Rectangle Problem. Discrete Applied Mathematics. 1984. P. 267–277.

6. **Sarkar A., Biswas A., Dutt M., Bhattacharya A.** Finding a largest rectangle inside a digital object and rectangularization. Journal of Computer and System Sciences. 2018, Vol. 95. P. 204–217.

7. **Xu G., Wu H-Z., and Shi YQ.** Ensemble of CNNs for steganalysis: An empirical study. Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security, 2016. P. 103–107.

REFERENCES

1. **Acharyya A, De M., Subhas C., Pandit S**. Variations of largest rectangle recognition amidst a bichromatic point set. Discrete Applied Mathematics. 2020, Vol. 286. P. 35–50.

2. **Chaumont M.** Deep learning in steganography and steganalysis. In Digital Media Steganography, Academic Press, 2020. P. 321–349.

3. **Chen M, Boroumand M, Fridrich J.** Deep learning regressors for quantitative steganalysis. Electron Imaging 2018, 7. P. 160–161.

4. **Cogranne R., Giboulot Q., Bas P.** The ALASKA steganalysis challenge: A first step towards steganalysis," in Proceedings of the ACM Workshop on Information Hiding and Multimedia Security, 2019. P. 125–137.

5. **Naamad A., Lee D. T., Hsu W.-L.** On the Maximum Empty Rectangle Problem. Discrete Applied Mathematics. 1984. P. 267–277.

6. **Sarkar A., Biswas A., Dutt M., Bhattacharya A.** Finding a largest rectangle inside a digital object and rectangularization. Journal of Computer and System Sciences. 2018. Vol. 95. P. 204–217.

7. **Xu G., Wu H-Z., and Shi Y. Q**. Ensemble of CNNs for steganalysis: An empirical study. Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security, 2016. P. 103–107.

Статья поступила в редакцию 14.02.2022; одобрена после рецензирования 21.02.2022; принята к публикации 25.02.2022.

The article was submitted 14.02.2022; approved after reviewing 21.02.2022; accepted for publication 25.02.2022.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

ВИЛЬХОВСКИЙ Данил Эдуардович — ассистент; Омский государственный университет им. Ф. М. Достоевского

VILKHOVSKY Danil E. — Assistant Professor; Dostoevsky Omsk State University
Email: vilkhovskiy@gmail.com
ORCID: 0000–0002–7348–9287

Научная статья
DOI 10.48612/jisp/bt14-g255–73ma
УДК 004.56

А. С. Соколов, А. Ю. Чернов, А. С. Коноплев
Санкт-Петербургский политехнический университет Петра Великого (СПбПУ)
Россия, 195251, Санкт-Петербург, ул. Политехническая, д. 29

ПОДХОД К РАЗРАБОТКЕ КРИПТОСЕРВИСОВ, УСТОЙЧИВЫХ К АТАКАМ НА ОСНОВЕ СПЕКУЛЯТИВНЫХ ВЫЧИСЛЕНИЙ

Аннотация. Появление атак семейства Meltdown/Spectre, эксплуатирующих уязвимости в процессорах Intel посредством злоупотребления спекулятивными вычислениями, подорвало уверенность в защищенности пользовательских конфиденциальных данных, к которым относится ключевая информация. Разработанные меры противодействия атакам Meltdown/Spectre оказались неэффективными с точки зрения нейтрализации новых разновидностей атак из данного семейства. В работе предлагается фундаментальное решение указанной проблемы благодаря использованию процессора Intel ME, архитектурно невосприимчивого к атакам семейства Meltdown/Spectre, в качестве основного вычислительного устройства при работе с криптосервисами. С практической точки зрения решение данной задачи может быть достигнуто посредством использования технологии Intel Dynamic Application Loader (Intel DAL).

Ключевые слова: атаки по сторонним каналам, спекулятивное вычисление, Meltdown, Spectre, Intel ME, Intel DAL, криптосервисы

Research article
DOI 10.48612/jisp/bt14-g255–73ma

A. S. Sokolov, A. Y. Chernov, A. S. Konoplev
Peter the Great St. Petersburg Polytechnic University, Russia, St. Petersburg

SPECULATIVE EXECUTION ATTACK-RESISTANT CRYPTOSERVICES

Abstract. The appearance of Meltdown/Spectre attacks exploiting the vulnerabilities in Intel processors via misuse of speculative executions has destroyed confidence in the security of user's confidential data which includes cryptoservices secret parameters. The developed Meltdown/Spectre countermeasures demonstrated ineffectiveness in neutralizing the newly designed speculative execution attacks. Paper highlights a fundamental solution of specified issue via Intel ME technology usage. The dedicated Intel ME processor is immune to Meltdown/Spectre attacks, which makes it effective to be used as a cryptoprocessor. Implementation of the proposed approach can be achieved through the usage of Intel Dynamic Application Loader (Intel DAL) technology.

Keywords: side-channel attacks, speculative execution, Meltdown, Spectre, Intel ME, Intel DAL, cryptoservice

Сложно представить современные вычислительные устройства без средств криптографии. Криптография позволяет поддерживать защищённые соединения в сети Интернет, обеспечивать целостность документов посредством механизмов электронной цифровой подписи (далее — ЭЦП), хранить пароли и другую конфиденциальную информацию в шифрованном виде и многое другое. Таким образом, любое пользовательское устройство так или иначе реализует или использует набор криптосервисов для обеспечения конфиденциальности и целостности обрабатываемой информации.

В рамках архитектуры Фон Неймана исполняемый код и данные совместно хранятся в памяти ЭВМ, а, следовательно, возникает необходимость изоляции контекста выполняющихся программ друг от друга, что достигается аппаратными механизмами виртуализации адресного пространства. Помимо этого, современная архитектура операционной системы (далее — ОС) подразумевает разделение системного адресного пространства (пространства ядра, Kernel Mode, KM) и пользовательского (User Mode, UM) в рамках модели колец защиты. Большинство системных криптосервисов функционируют в пространстве ядра с уровнем привилегий CPL=0, что позволяет получить как высоко привилегированный доступ к оборудованию и функциональности ОС, так и надёжную изоляцию критически важных участков памяти криптосистемы от пользователя (например, ключевую информацию). С точки зрения нарушителя, существуют следующие основные вектора воздействия на криптосистему:

1. Воздействие на программную реализацию криптосистемы с целью выделения уязвимостей и недостатков.

2. Воздействие на программно-аппаратное окружение, в котором функционирует криптосистема, в частности ОС.

Самым простым способом получения конфиденциальных данных из окружения криптосистемы является поиск артефактов в адресном пространстве криптосистемы при её функционировании или в снимках памяти ОС. По такому принципу работают утилиты mimikatz и volatile. Реализация данного способа подразумевает получение достаточно высоких прав в системе, что может быть реализовано с помощью применения атак класса Priviledge Escalation. Следовательно, безопасность вычислительных процессов криптосистемы целиком и полностью зависит от механизмов защиты ОС, которые в свою очередь базируются на механизмах безопасности аппаратного обеспечения, в частности механизмах изоляции и разграничения доступа к различным участкам памяти. Однако появление атак по сторонним каналам на основе спекулятивных вычислений предоставляют возможности по обходу аппаратных механизмов обеспечения безопасности, что в свою очередь ведёт к компрометации механизмов безопасности ОС.

Спекулятивные вычисления. Механизм спекулятивного и внеочередного вычисления — один из важнейших атрибутов суперскалярных микропроцессоров. Известно, что время выполнения машинных инструкций процессором достаточно сильно зависит от их операндов. Например, если происходит работа с регистровыми операндами, то процессор выполняет данную инструкцию сразу, если происходит обращение к оперативной памяти, то процессор должен дождаться ответа от контроллера памяти, чтобы получить значение операнда инструкции (которое затем помещается в кэш процессора). Внеочередное исполнение позволяет использовать время ожидания процессора для выполнения следующих инструкций, операнды которых готовы для обработки. В случае если главный поток исполнения разветвляется, т.е. доходит до инструкций условного перехода, то процессор может сделать прогноз относительно пути следования и спекулятивно выполнить инструкции согласно выбранному пути.

Особенностью спекулятивных вычислений является то, что в случае неверного предсказания в кэше процессора остаются артефакты обрабатываемой в процессе спекуляций информации. Таким образом, используя различные методики по извлечению артефактов из кэша процессора (например, FLUSH-RELOAD), можно осуществить нелегитимный доступ к различным участкам оперативной памяти, например, к пространству ядра ОС.

Применительно к криптосистемам можно рассмотреть следующий сценарий атак, схематично изображенный на рисунке 1. Допустим, в ОС предустановлен криптосервис, выполняющий с некоторой периодичностью проверку ЭЦП. Процесс проверки ЭЦП включает следующие шаги:

1. Чтение значения ЭЦП из памяти («чтение[1]» на рисунке 1).

2. В случае успеха шага 1, происходит чтение ключевой информации из памяти («чтение[2]» на рисунке 1).

3. В случае успеха шага 2, производится валидация ЭЦП.

Наиболее интересными являются первые 2 шага, так как там происходит работа с памятью (в процессе которой данные попадают в кэш) и условный переход. Благодаря периодичности проверки процессор «запоминает» путь основного потока исполнения («чтение[1]», «чтение[2]» и т.д.), что помогает ему с высокой точностью предсказывать порядок следования инструкций и, тем самым, избегать простоев с помощью использования внеочередного и спекулятивного исполнения команд. В общем случае это означает, что после выполнения операции «чтение[1]» процессор уже будет знать результат операции «чтение[2]», так как она выполнилась спекулятивно.

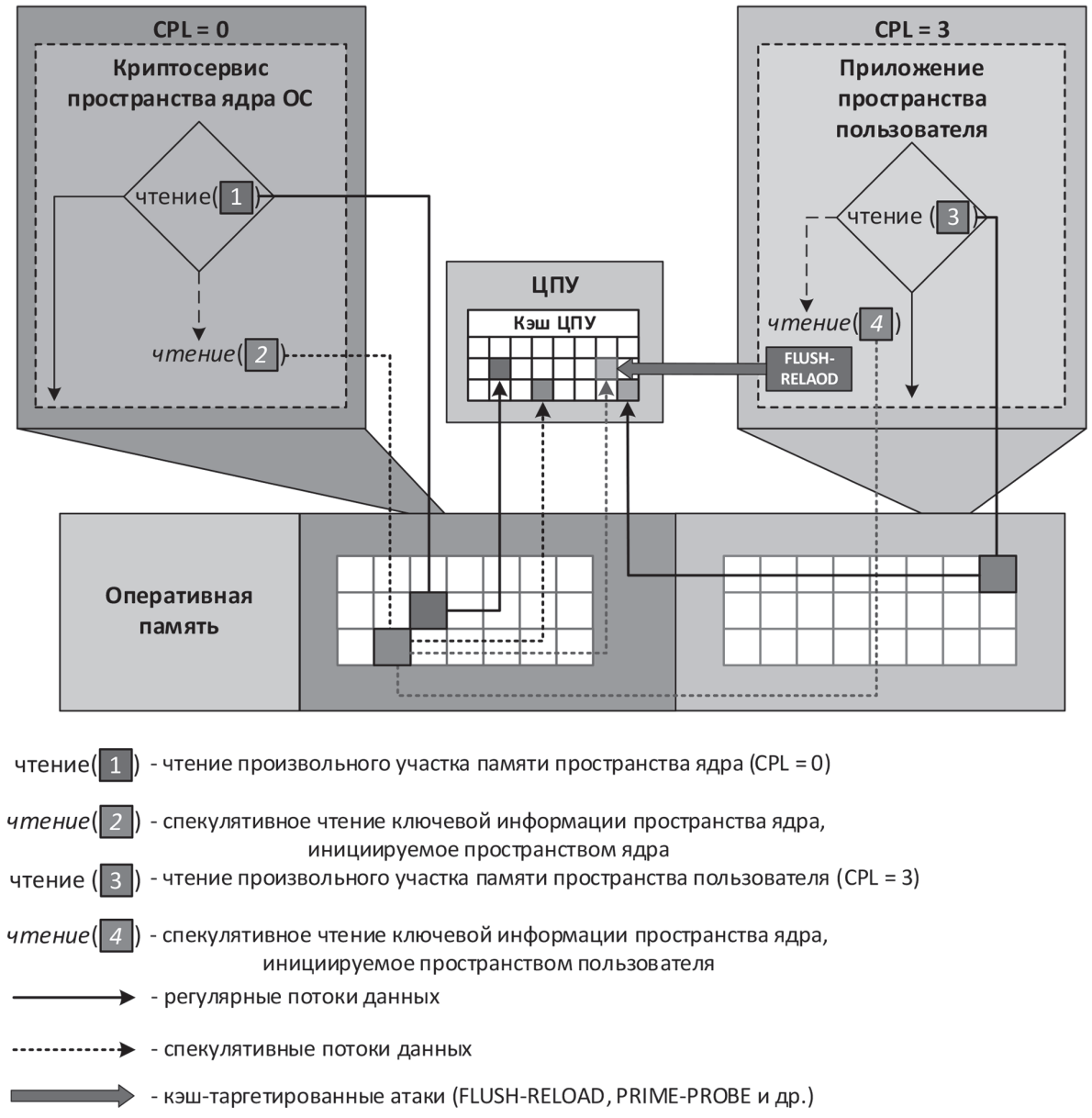


Рис. 1. Извлечение данных криптосистемы с помощью атак по сторонним каналам на основе спекулятивных вычислений

Fig. 1. Cryptosystem data extraction using side-channel attacks based on speculative computing

Особенностью спекулятивного исполнения является то, что в процессе спекуляций процессор не производит проверку прав доступа к обращаемой области памяти и, соответственно, осуществляет её чтение, остаточная информация которого остаётся в кэше. Таким образом, злоумышленник способен разработать собственное приложение и осуществить спекулятивное чтение высоко привилегированных участков памяти. На рисунке 1 данный процесс представлен операциями «чтение[3]» и «чтение[4]». Алгоритм работы следующий:

1. Сначала производится «тренировка» предсказателя процессора на доступном диапазоне адресов памяти. Благодаря этому операция «чтение [4]» будет выполняться спекулятивно.

2. Так как процессор выполняет операцию «чтение [4]» спекулятивно, то в качестве целевого адреса чтения подставляется адрес защищённого участка памяти (например, области криптосервиса).

3. Процессор осуществляет спекулятивное чтение данного участка памяти, в результате чего в кэше остается остаточная информация.

4. С помощью методики FLUSH-RELOAD становится возможным извлечь остаточную информацию.

Таким образом, с помощью данной атаки становится возможным осуществить доступ к защищенным участкам памяти, а, следовательно, извлечь ключевую информацию криптосервиса.

Атаки семейства Spectre/Meltdown. Появление класса атак на основе спекулятивных вычислений (Spectre и Meltdown) дало мощный толчок множеству исследований по изучению функциональных особенностей и недостатков аппаратной архитектуры современных микропроцессоров производства компании Intel, в результате чего, с достаточной частой периодичностью появляются новые варианты атак по сторонним каналам на основе спекулятивного или внеочередного исполнения.

Главной опасностью данных атак является сложный процесс их устранения, который в каждом отдельном случае нужно решать на разном уровне: на уровне приложения, ОС или микрокода процессора. Поэтому для каждой новой разработанной атаки приходится разрабатывать новый метод ее нейтрализации. В таблице 1 представлено сопоставление атак и методов противодействия.

Условные обозначения для Таблицы 1:

– «+» — данный механизм явно нейтрализует атаку;

– «+/-» — данный механизм противодействует атаке, но не нейтрализует ее полностью;

– «-» — данный механизм не противодействует атаке.

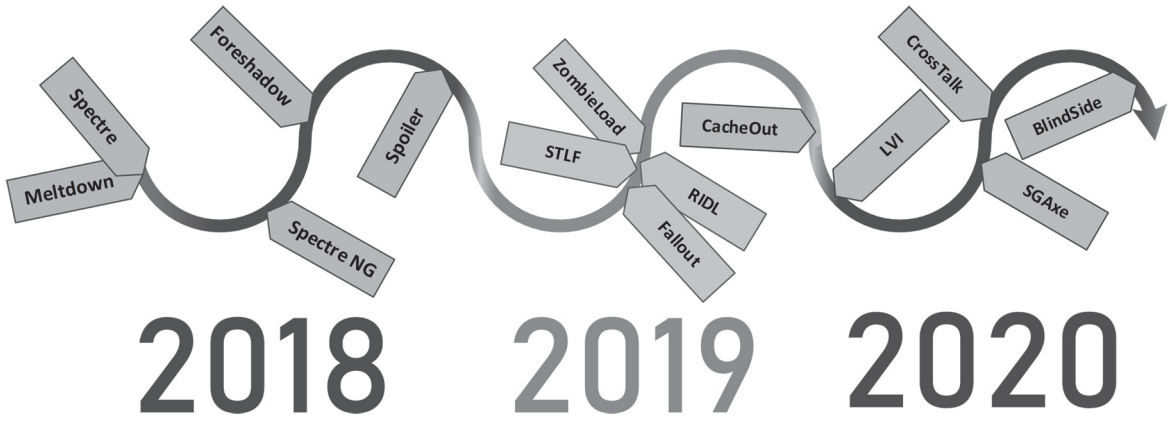


Рис. 2. Дорожная карта атак по сторонним каналам на основе спекулятивного или внеочередного исполнения

Fig. 2. Roadmap of attacks through third-party channels based on speculative or out-of-order execution

Таблица 1. Основные механизмы противодействия спекулятивным атакам

Table 1. Main mechanisms for countering speculative attacks

Механизм защиты / Атака	Отключение спекулятивного исполнения	Отключение HyperThreading	Очистка микроархитектурных буферов (не кэша) или явная очистка кэша	KPTI	Retpoline	IBRS, STIBP, IBPB	SSBD
Meltdown	+	–	–	+	–	–	–
Spectre	+	–	–	–	+	+	+
ForeShadow	+	+	+	–	–	–	–
ForeShadow-NG	+	+	+	–	–	–	–
RIDL	+	–	+	–	+	+	+
ZombieLoad	+	+	+	–	–	–	–
Fallout	+	–	+	–	–	–	–
LVI	+	–	–	–	–	–	–
CacheOut	+	+/–	+/–	–	–	–	–
CrossTalk	+	–	+	–	–	–	–
BlindSide	+	–	–	–	–	–	–

На основании проведенного анализа механизмов защиты от атак на основе спекулятивных вычислений была разработана классификация механизмов защиты по уровню их применения:

1. На уровне микрокода. К ним относятся полное отключение спекулятивного исполнения, отключение гиперпоточности, очистка микроархитектурных буферов при смене контекста выполнения, IBRS, STIBP, SSBD.

2. На уровне компилятора. К ним относятся отключение спекулятивного исполнения с помощью специальных команд процессора, например, LFENCE, очистка микроархитектурных буферов с помощью специальных команд процессора, например, VERW, Retpoline, IBPB.

3. На уровне ОС. К ним относится KPTI/KVAShadow.

4. На уровне антивирусов. К ним относится запрет выполнения специфичных инструкций, используемых в атаках по сторонним каналам на основе, например, clflush.

Также механизмы защиты от атак спекулятивного исполнения можно разделить на следующие классы:

1. Аппаратные. К ним относятся полное отключение спекулятивного исполнения, отключение гиперпоточности, очистка микроархитектурных буферов при смене контекста выполнения, IBRS, STIBP, SSBD.

2. Программные. К ним относятся KPTI/KVAShadow, IBPB, запрет выполнения специфичных инструкций, используемых в атаках по побочным каналам на основе, например, clflush, mprotect.

3. Программно-аппаратные. К ним относятся отключение спекулятивного исполнения с помощью специальных команд процессора, например, LFENCE, очистка микроархитектурных буферов с помощью специальных команд процессора, например, VERW, Retpoline.

Как видно из рисунка 2, появление атак Spectre и Meltdown дало мощный толчок к развитию атак на основе спекулятивных вычислений и по побочным каналам. Более того, таблица 2 показывает, что применяемые механизмы противодействия позволяют защититься от точечных атак, но не от всего семейства атак. Наиболее эффективным механизмом противодействия, исключающим весь класс атак на основе спекулятивных вычислений, является полное его отключение, что невозможно из-за снижения производительности процессоров, которое в свою очередь приведет к неизбежным экономическим затратам.

Предлагаемый подход. С 2008 года платформы на базе процессоров Intel поставляются с интегрированной в них технологией Intel Management Engine (Intel ME). Intel ME представляет собой выделенный сопроцессор низкого энергопотребления, который начинает функционировать при подаче дежурного питания на системную плату. Данная технология является фундаментом для таких технологий как Intel PAVP, Intel AMT, Intel Anti-Theft и т.д. Помимо этого, сам сопроцессор, с момента введения микроархитектуры Skylake, основан на микроконтроллере Minute IA, архитектура которого не поддерживает суперскалярные вычисления, а, следовательно, атаки класса Spectre/Meltdown не могут быть осуществимы. Таким образом, использование Intel ME в качестве основного вычислительного устройства позволяет разработать систему обработки данных (в том числе и криптосервис), устойчивую к атакам на основе спекулятивных вычислений. Решение данной задачи может быть осуществлено использованием технологии Intel Dynamic Application Loader (Intel DAL).

Intel DAL — технология компании Intel, которая позволяет запускать Java-апплеты в контексте вычислительной среды Intel ME. В её основе лежит виртуальная Java-машина, разработанная компанией Intel, которая входит в состав всех современных Intel-based ЭВМ. Применение технологии Intel DAL позволяет избежать влияния известных архитектурных недостатков центрального процессора. При этом, главным достоинством предлагаемого подхода является устойчивость к новым вариациям атак данного семейства, потенциальное использование которых не сможет ни при каких обстоятельствах влиять на процессы криптографических вычислений в целом, так как их функционирование производится в контексте подсистемы Intel ME.

СПИСОК ИСТОЧНИКОВ/REFERENCES

1. **Deepen Desai, Naresh Annangar**. Meltdown and Spectre vulnerabilities: What you need to know. [Электронный ресурс]. URL: https://www.zscaler.com/blogs/research/meltdown-and-spectre-vulnerabilities-what-you-need-know/.— (дата обращения: 25.01.2022).

2. **Moritz Lipp** et al. Meltdown: Reading kernel memory from user space// 27th {USENIX} Security Symposium ({USENIX} Security 18).— 2018.— P. 973–990.

3. **Paul Kocher** et al. Spectre attacks: Exploiting speculative execution.// 2019 IEEE Symposium on Security and Privacy (SP). IEEE.— 2019.— P. 1–19.

4. **Matt Klein**. Meltdown and Spectre, explained. [Электронный ресурс]. URL: https://medium.com/@mattklein123/meltdown-spectre-explained-6bc8634cc0c2/.— (дата обращения: 12.01.2022).

5. Deep Dive: Intel Analysis of Microarchitectural Data Sampling. [Электронный ресурс]. URL: https://software.intel.com/security-software-guidance/insights/ deep-dive-intel-analysis-microarchitectural-data-sampling#MSBDS/.— (дата обращения: 12.01.2022).

6. Microarchitectural Data Sampling / CVE-2018–12126, CVE-2018–12127, CVE-2018–12130, CVE-2019–11091 / INTEL-SA-00233. [Электронный ресурс]. URL: https://software.intel.com/security-software-guidance/software-guidance/ microarchitectural-data-sampling/— (дата обращения: 15.06.2020).

7. **Jo Van Bulck et al.** FORESHADOW: Extracting the Keys to the Intel SGX Kingdom with Transient Out-of-Order Execution//27th {USENIX} Security Symposium ({USENIX} Security 18).— 2018.— P. 991–1008.

8. Deep Dive: Intel® Transactional Synchronization Extensions (Intel® TSX) Asynchronous Abort. [Электронный ресурс]. URL: https://software.intel.com/security-software-guidance/insights/deep-dive-intel-transactional-synchronization-extensions-intel-tsx-asynchronous-abort/— (дата обращения: 14.01.2022).

9. L1D Eviction Sampling / CVE-2020-0549 / INTEL-SA-00329. [Электронный ресурс]. URL: https://software.intel.com/security-software-guidance/software-guidance/l1d-eviction-sampling/.— (дата обращения: 12.01.2022).

10. L1 Terminal Fault / CVE-2018–3615, CVE-2018–3620, CVE-2018–3646 / INTEL-SA-00161. [Электронный ресурс]. URL: https://software.intel.com/ security-software-guidance/software-guidance/l1-terminal-fault/.— (дата обращения: 08.01.2022).

11. **Ofir Weisse** et al. Foreshadow-NG: Breaking the virtual memory abstraction with transient out-of-order execution.— 2018.

12. **Stephan Van Schaik** et al. RIDL: Rogue inflight data load//2019 IEEE Symposium on Security and Privacy (SP). IEEE.— 2019.— P. 88–105.

13. **Michael Schwarz** et al. ZombieLoad: Cross-privilege-boundary data sampling//Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security.— 2019.— P. 753–768.

14. **Claudio Canella** et al. Fallout: Leaking data on meltdown-resistant cpus//Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security.— 2019.— P. 769–784.

15. **Ionut Ilascu**. New RIDL and Fallout Attacks Impact All Modern Intel CPUs. [Электронный ресурс]. URL: https://www.bleepingcomputer.com/news/ security/new-ridl-and-fallout-attacks-impact-all-modern-intel-cpus/.— (дата обращения: 05.01.2022).

16. **Jo Van Bulck** et al. LVI: Hijacking transient execution through microarchitectural load value injection//2020 IEEE Symposium on Security and Privacy (SP). IEEE.— 2020.— P. 54–72.

17. Deep Dive: Load Value Injection. [Электронный ресурс]. URL: https://software.intel.com/security-software-guidance/insights/deep-dive-load-value-injection#os-vmm/.— (дата обращения: 05.01.2022).

18. **Stephan van Schaik** et al. CacheOut: Leaking data on Intel CPUs via cache evictions//arXiv preprint arXiv:2006.13353.— 2020.

19. **Hany Ragab** et al. Crosstalk: Speculative data leaks across cores are real//IEEE Symposium on Security & Privacy.— 2021.

20. **Enes Göktas** et al. Speculative Probing: Hacking Blind in the Spectre Era//Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security.— 2020.— P. 1875–1885.

21. Retpoline: A Branch Target Injection Mitigation. [Электронный ресурс]. URL: https://www.intel.com/content/dam/develop/external/us/en/documents/retpoline-a-branch-target-injection-mitigation.pdf.— (дата обращения: 05.01.2022).

22. Indirect Branch Restricted Speculation. [Электронный ресурс]. URL: https://www.intel.com/content/www/us/en/developer/articles/

technical/software-security-guidance/technical-documentation/indirect-branch-restricted-speculation.html.— (дата обращения: 07.01.2022).

23. Single Thread Indirect Branch Predictors. [Электронный ресурс]. URL: https://www.intel.com/content/www/us/en/developer/articles/technical/software-security-guidance/technical-documentation/single-thread-indirect-branch-predictors.html.— (дата обращения: 07.01.2022).

24. Indirect Branch Predictor Barrier. [Электронный ресурс]. URL: https://www.intel.com/content/www/us/en/developer/articles/technical/software-security-guidance/technical-documentation/indirect-branch-predictor-barrier.html.— (дата обращения: 07.01.2022).

25. Speculative Execution Side Channel Mitigations. URL: https://www.intel.com/content/dam/develop/external/us/en/documents/336996-speculative-execution-side-channel-mitigations.pdf.— (дата обращения: 07.01.2022).

26. KVA Shadow: Mitigating Meltdown on Windows. [Электронный ресурс]. URL: https://www.intel.com/content/www/us/en/developer/articles/technical/software-security-guidance/technical-documentation/single-thread-indirect-branch-predictors.html.— (дата обращения: 07.01.2022).

27. Intel Dynamic Application Loader (Intel DAL) Developer Guide. [Электронный ресурс]. URL: https://www.intel.com/content/www/us/en/develop/documen tation/dal-developer-guide/top.html.— (дата обращения: 24.01.2022).

28. What is Intel® Management Engine? [Электронный ресурс]. URL: https://www.intel.com/content/www/us/en/support/articles/000008927/software/chipset-software.html.— (дата обращения: 24.01.2022).

29. Intel CSME (Converged Security and Management Engine). [Электронный ресурс]. URL: https://www.tadviser.ru/index.php/%D0%9F%D1%80%D0% BE%D0%B4%D1%83%D0%BA%D1%82: Intel_CSME_(Converged_Security_and_Management_Engine).— (дата обращения: 01.02.2022).

30. **Ivanov D. V.** Application of Fractal Methods to Ensure the Cyber-Resilience of Self-Organizing Networks / D. V. Ivanov, D. A. Moskvin // Nonlinear Phenomena in Complex Systems.— 2019.— Vol. 22.— No 4.— P. 336–341.— DOI 10.33581/1561-4085-2019-22-4-336-341.

31. **Dakhnovich A. D.** Approach for Securing Network Communications Modelling Based on Smart Multipath Routing / A. D. Dakhnovich, D. A. Moskvin, D. P. Zegzhda // Nonlinear Phenomena in Complex Systems.— 2020.— Vol. 23.— No 4.— P. 386–396.— DOI 10.33581/1561-4085-2020-23-4-386-396.

Статья поступила в редакцию 09.02.2022; одобрена после рецензирования 16.02.2022; принята к публикации 22.02.2022.

The article was submitted 09.02.2022; approved after reviewing 16.02.2022; accepted for publication 22.02.2022.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

СОКОЛОВ Александр Сергеевич — студент; Санкт-Петербургский Политехнический университет Петра Великого

ЧЕРНОВ Андрей Юрьевич —старший преподаватель; Санкт-Петербургский Политехнический университет Петра Великого

КОНОПЛЕВ Артем Станиславович — к.т.н., доцент; Санкт-Петербургский Политехнический университет Петра Великого

SOKOLOV Alexander S.— student; Peter the Great St. Petersburg Polytechnic University

Email: sokolov2.as@edu.spbstu.ru

CHERNOV Andrey Yu.— Senior Lecturer; Peter the Great St. Petersburg Polytechnic University

Email: chernov@ibks.spbstu.ru

KONOPLEV Artem S.— Ph.D., Associate Professor; Peter the Great St. Petersburg Polytechnic University

Email: a.konoplev@ibks.spbstu.ru

ORCID: 0000-0002-9074-4366

Научная статья
DOI 10.48612/jisp/ktu1–1632-n54t
УДК 519.718

К. Ю. Сопин, С. А. Диченко, Д. В. Самойленко
Краснодарское высшее военное училище имени С. М. Штеменко
Россия, 350063, Краснодар, ул. Красина, д. 4

КРИПТОГРАФИЧЕСКИЙ КОНТРОЛЬ ЦЕЛОСТНОСТИ ДАННЫХ НА ОСНОВЕ ГЕОМЕТРИЧЕСКИХ ФРАКТАЛОВ

Аннотация. Рассматриваются новые сложные задачи, связанные с обеспечением безопасности информации при масштабировании систем хранения данных. Представлен способ криптографического контроля целостности больших массивов данных на основе геометрических фракталов.

Ключевые слова: система хранения данных, защита информации, контроль целостности данных, хэш-функция, треугольник Серпинского.

Research article
DOI 10.48612/jisp/ktu1–1632-n54t

K. Yu. Sopin, S. A. Dichenko, D. V. Samoylenko
Krasnodar Higher Military School named after S. M. Shtemenko, Russia, Krasnodar

CRYPTOGRAPHIC DATA INTEGRITY CONTROL BASED ON GEOMETRIC FRACTALS

Abstract. New complex tasks related to information security when scaling data storage systems are considered. A method of cryptographic integrity control of large data arrays based on geometric fractals is presented.

Keywords: data storage system, information protection, data integrity control, hash function, Sierpinski triangle.

В условиях непрерывного роста объемов информации наиболее совершенным оборудованием для хранения данных являются централизованные системы хранения, использующие специализированную сеть для транспортировки информации к вычислительным комплексам и обратно. При недостаточной емкости систем хранения данных (СХД) или значительном возрастании на них нагрузки, при которой существующей производительности недостаточно, задача решается путем масштабирования без негативного влияния на их работу [1, 2]. При этом совершенствование существующих способов масштабирования СХД ограничивается развитием методов оптимизации хранения данных и не учитывает необходимость развития механизмов их защиты.

Применение вертикального (Scale-up) или горизонтального масштабирования (Scale-out) СХД при росте объемов хранящихся данных приводит к прямо пропорциональному увеличению контрольной информации, требуемой для обеспечения безопасности их хранения, что приводит к преждевременному расходованию ограниченного ресурса СХД и требует нового масштабирования. Одной из мер обеспечения безопасности данных в СХД является обеспечение (контроль и восстановление) их целостности [3].

Издатель: Санкт-Петербургский политехнический университет Петра Великого

Одним из популярных способов контроля целостности данных является применение криптографических методов (функции хэширования) [4–6]. Однако, несмотря на повсеместное применение хэш-функций, информации о них гораздо меньше, чем, к примеру, о блочных шифрах [7]. В сравнении с блочными шифрами хэш-функции крайне мало исследованы, а практические предложения по их применению весьма немногочисленны [8–13].

Известны способы контроля целостности данных [14–17], в которых хэш-функция применяется к защищаемым данным для возможности обнаружения и локализации подблоков данных с признаками нарушения целостности, однако их применение не позволяет решить эту задачу при масштабировании СХД без введения дополнительной контрольной информации, прямо пропорциональной объему увеличиваемых данных.

В классических СХД к данным $\mathbf{M}^{(I)}$, подлежащим защите, интерпретируемым как вектор:

$$\mathbf{M}^{(I)} = \left[\mu_1^{(I)} \quad \mu_2^{(I)} \quad \ldots \quad \mu_\tau^{(I)} \right],$$

где $\mu_g^{(I)} \in \{0,1\}$; $g = 1, 2, \ldots, \tau$, для возможности контроля их целостности применяется хэш-функция h и вычисляется ее значение:

$$\mathbf{H}^{(I)} = h\left(\mathbf{M}^{(I)}\right).$$

При масштабировании СХД для контроля целостности новых данных: $\mathbf{M}^{(II)}$, $\mathbf{M}^{(III)}$, ..., $\mathbf{M}^{(\Psi)}$ аналогичным образом применяется хэш-функция и вычисляются ее значения: $\mathbf{H}^{(II)}$, $\mathbf{H}^{(III)}$, ..., $\mathbf{H}^{(\Psi)}$.

При очередном масштабировании СХД требуется вычисление новых значений хэш-функции — контрольной информации, объем которой возрастает прямо пропорционально объему увеличиваемых данных, для хранения которой требуется новый ресурс СХД, и, как следствие, очередное масштабирование.

Для решения задачи обеспечения безопасности информации при масштабировании СХД предлагается способ на основе применения хэш-функции по правилам построения треугольника Серпинского, который является геометрическим фракталом и обладает свойством самоподобия, что позволяет осуществлять контроль целостности больших массивов данных без введения дополнительной контрольной информации, прямо пропорциональной объему увеличиваемых данных, подлежащих защите.

Способ контроля целостности данных на основе правил построения треугольника Серпинского

В разработанном способе схема применения хэш-функции для контроля целостности исходного блока данных $\mathbf{M}^{(I)}$ будет являться промежуточной. Для новых данных $\mathbf{M}^{(II)}$, $\mathbf{M}^{(III)}$, поступающих на хранение в СХД, разрабатываются схемы применения хэш-функции по аналогичным правилам, как для данных $\mathbf{M}^{(I)}$. Они объединяются в общую схему, которая также является промежуточной и т.д.

Получим схему (рис. 1) контроля целостности данных на основе правил построения треугольника Серпинского [18, 19], которая включает в себя: схемы контроля целостности блоков данных $\mathbf{M}^{(I)}$, $\mathbf{M}^{(II)}$, $\mathbf{M}^{(III)}$, которые являются промежуточными для контроля целостности блоков данных $\mathbf{M}'^{(I)}$, $\mathbf{M}'^{(II)}$, $\mathbf{M}'^{(III)}$, схемы контроля целостности которых, в свою очередь, будут являться промежуточными для контроля целостности блоков данных $\mathbf{M}''^{(I)}$, $\mathbf{M}''^{(II)}$, $\mathbf{M}''^{(III)}$ и т.д.

Таким образом, для контроля целостности блоков данных $\mathbf{M}^{(I)}$, $\mathbf{M}^{(II)}$, $\mathbf{M}^{(III)}$ вычисляется и хранится значение $\mathbf{H}'^{(I)}$ хэш-функции, для блоков данных $\mathbf{M}'^{(I)}$, $\mathbf{M}'^{(II)}$, $\mathbf{M}'^{(III)}$ — значение $\mathbf{H}''^{(I)}$ хэш-функции и т.д.

Если блок данных $\mathbf{M}^{(I)}$, подлежащий защите, обозначить как $\mathbf{M}$, контроль его целостности может быть осуществлен следующим образом.

Блок данных $\mathbf{M}$ фрагментируется на блоки данных фиксированной длины:

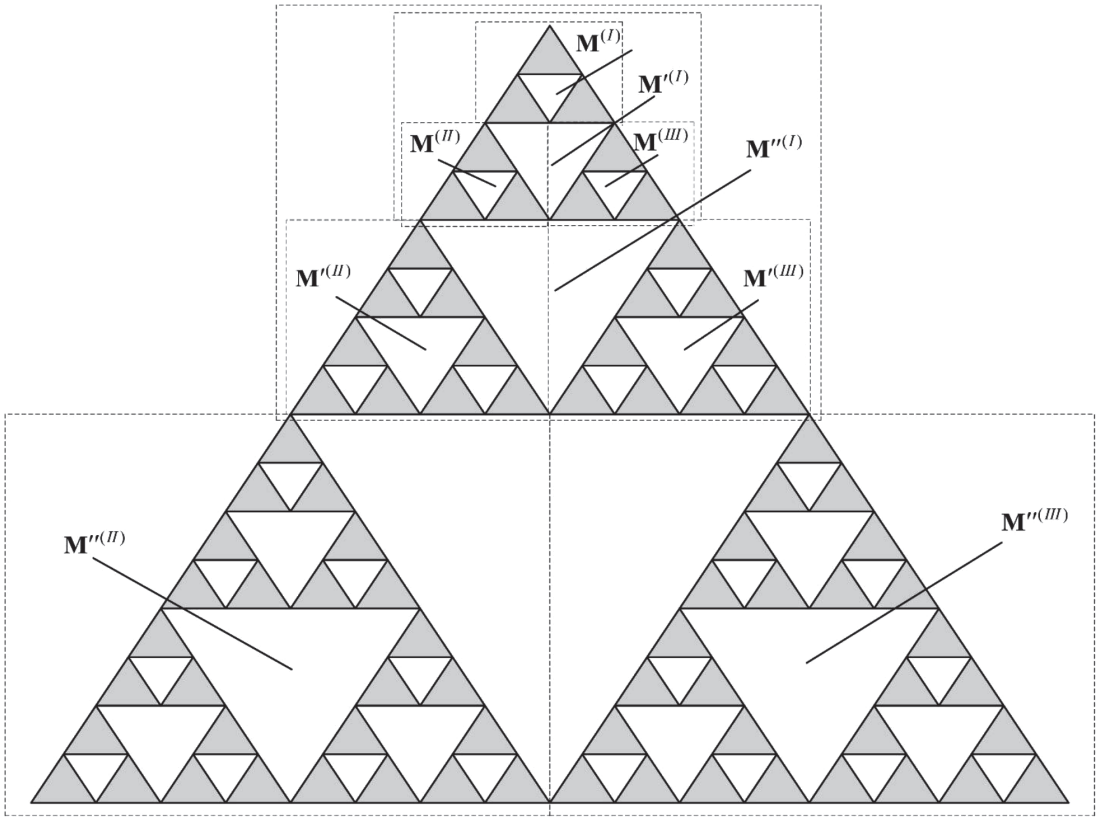


Рис. 1. Схема контроля целостности данных на основе правил построения треугольника Серпинского

Fig. 1. Data integrity control scheme based on Sierpinski triangle construction rules

$$\mathbf{M} = \left\{ \mathbf{M}^{(1)} \| \mathbf{M}^{(2)} \| \mathbf{M}^{(3)} \right\},$$

каждый из которых, в свою очередь, также фрагментируется на блоки данных $\mathbf{M}_{i,j}^{(\varphi)}$, где $\varphi = 1, 2, 3$; $i = 1, 2, ..., n$; $j = 1, 2$, к которым для контроля целостности данных применяется хэш-функция.

Схема применения хэш-функции к полученным блокам данных $\mathbf{M}_{i,j}^{(\varphi)}$ основана на правилах построения треугольника Паскаля [20] и представлена в виде таблицы, имеющей треугольную форму (рис. 2).

В соответствии с [21] по боковым сторонам полученного треугольника последовательно размещены блоки данных $\mathbf{M}_{i,j}^{(\varphi)}$ ($i = 1, 2, ..., n$; $j = 1, 2$) подлежащие защите.

Внутри треугольника — промежуточные результаты преобразований, в частности блоки со значениями $\mathbf{H}_{i,j}^{(\varphi)}$ ($i = 1, 2, ..., n - 1$; $j = 1, 2, ..., n - 1$) хэш-функции h, вычисляемые от:

– блоков данных $\mathbf{M}_{i,j}^{(\varphi)}$ ($i = 1$; $j = 1, 2$):

$$\mathbf{H}_{1,1}^{(\varphi)} = h\left(\mathbf{M}_{1,1}^{(\varphi)} \| \mathbf{M}_{1,2}^{(\varphi)} \right);$$

– блоков данных $\mathbf{M}_{i,j}^{(\varphi)}$ ($i = 1, 2, ..., n - 1$; $j = 1, 2$) и блоков со значениями $\mathbf{H}_{i,j}^{(\varphi)}$ ($i = 1, 2, ..., n - 2$; $j = 1, 2, ..., n - 2$) хэш-функции, к примеру:

$$\mathbf{H}_{t,1}^{(\varphi)} = h\left(\mathbf{M}_{t,1}^{(\varphi)} \| \mathbf{H}_{t-1,1}^{(\varphi)} \right);$$

– блоков со значениями $\mathbf{H}_{i,j}^{(\varphi)}$ ($i = 1, 2, ..., n - 2$; $j = 1, 2, ..., n - 2$) хэш-функции, к примеру:

$$\mathbf{H}_{t,2}^{(\varphi)} = h\left(\mathbf{H}_{t-1,1}^{(\varphi)} \| \mathbf{H}_{t-1,2}^{(\varphi)} \right).$$

На нижней стороне треугольника — блоки со значениями $\mathbf{H}_{i,j}^{(\varphi)}$ ($i = n$; $j = 1, 2, ..., n$) хэш-функции, вычисляемые от блоков данных $\mathbf{M}_{i,j}^{(\varphi)}$ ($i = n$; $j = 1, 2$) и результатов промежуточных преобразований нижнего уровня:

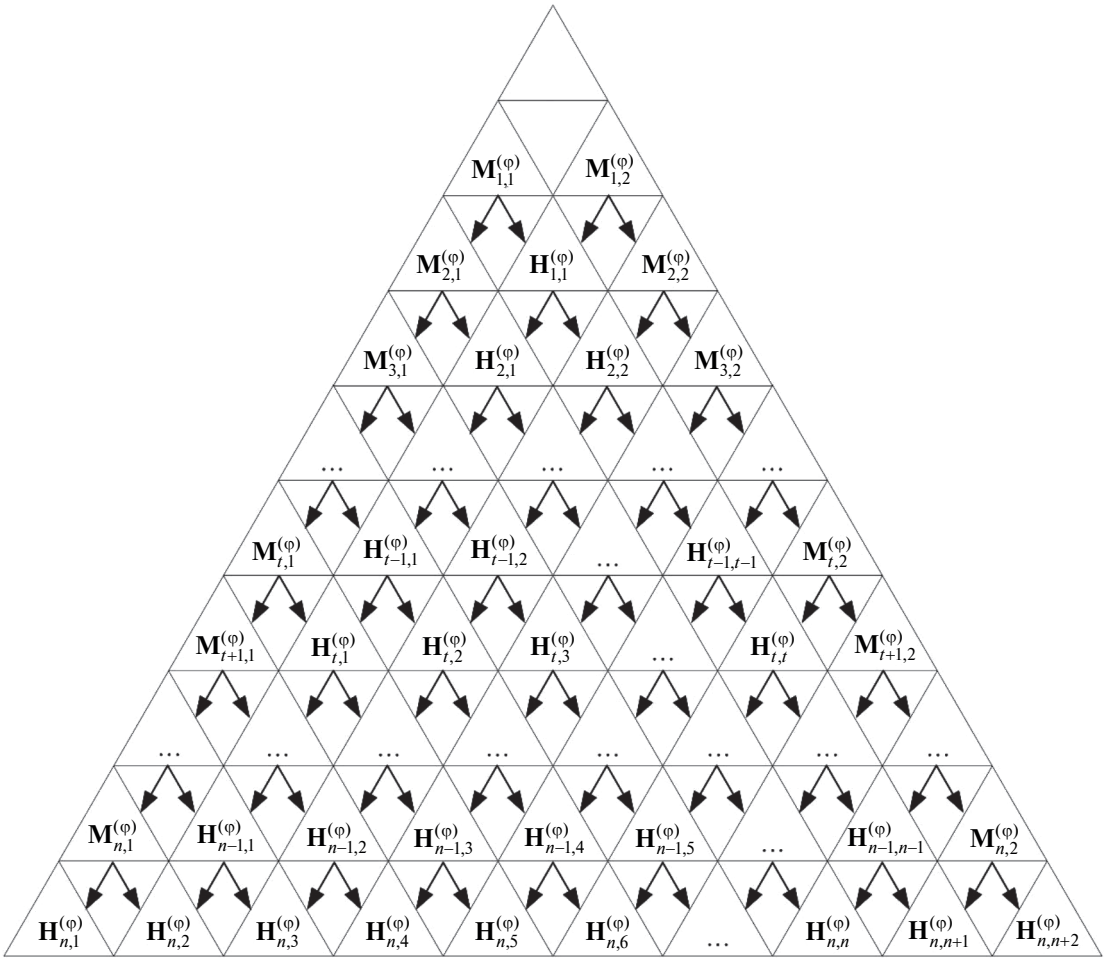


Рис. 2. Схема применения хэш-функции к блоку данных $\mathbf{M}^{(\varphi)}$

Fig. 2. Scheme for applying a hash function to a data block $\mathbf{M}^{(\varphi)}$

$$\mathbf{H}^{(\varphi)}_{n-1,1}, \mathbf{H}^{(\varphi)}_{n-1,2}, \ldots, \mathbf{H}^{(\varphi)}_{n-1,n-1}.$$

Блоки со значениями $\mathbf{H}^{(\varphi)}_{i,j}$ ($i = n; j = 1, 2, \ldots, n, n+1, n+2$) хэш-функции будут использоваться для контроля целостности блоков данных $\mathbf{M}^{(\varphi)}_{i,j}$ ($i = 1, 2, \ldots, n; j = 1, 2$).

Полученные для блоков данных $\mathbf{M}^{(\varphi)}_{i,j}$ схемы применения хэш-функции, имеющие треугольные формы, объединяются в общую схему, в которой схемы применения хэш-функции для блоков данных $\mathbf{M}^{(2)}$ и $\mathbf{M}^{(3)}$ относительно схемы применения хэш-функции для блока данных $\mathbf{M}^{(1)}$ будут повернуты против и по часовой стрелке соответственно, как показано на рисунке 3.

От блоков со значениями $\mathbf{H}^{(\varphi)}_{i,j}$ ($i = n; j = 1, 2, \ldots, n, n+1, n+2$) хэш-функции, размещенных на нижних сторонах треугольников, последовательно в определенном порядке вычисляются k блоков со значениями хэш-функции, среди которых — один или несколько (зависит от значения n) блоков со значениями хэш-функции, применяемой к элементам из каждого блока данных $\mathbf{M}^{(1)}$, $\mathbf{M}^{(2)}$ и $\mathbf{M}^{(3)}$.

Пример 1. Для блоков данных $\mathbf{M}^{(\varphi)}_{i,j}$ построим схемы применения хэш-функции. При $n = 6$ на нижней стороне каждого треугольника получим блоки со значениями $\mathbf{H}^{(\varphi)}_{6,1}, \ldots, \mathbf{H}^{(\varphi)}_{6,8}$ хэш-функции. Последовательно вычислим от них значения хэш-функции:

Шаг 1. $\mathbf{H}^{(1,2)}_{7,1} = h\left(\mathbf{H}^{(2)}_{6,8} \| \mathbf{H}^{(1)}_{6,1}\right)$; ...;

$\mathbf{H}^{(2,3)}_{7,3} = h\left(\mathbf{H}^{(3)}_{6,8} \| \mathbf{H}^{(2)}_{6,1}\right)$.

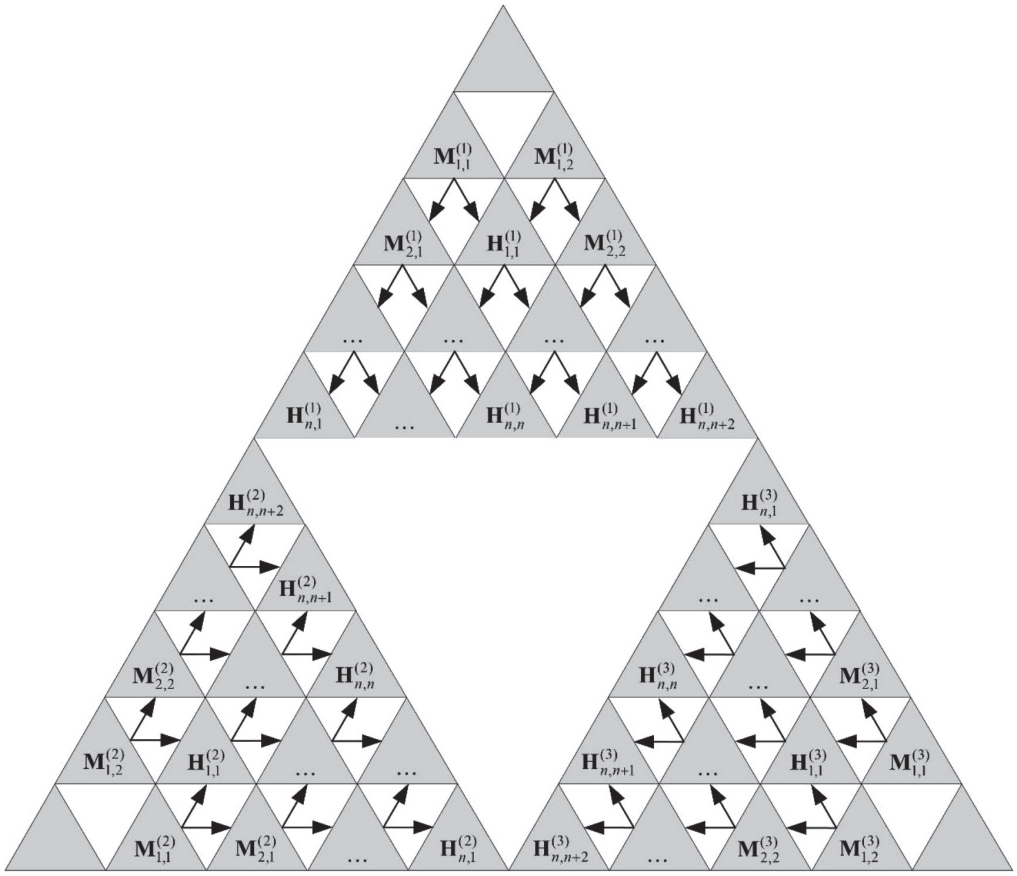


Рис. 3. Схема применения хэш-функции к блокам данных $\mathbf{M}^{(1)}$, $\mathbf{M}^{(2)}$ и $\mathbf{M}^{(3)}$

Fig. 3. Scheme of applying the hash function to data blocks $\mathbf{M}^{(1)}$, $\mathbf{M}^{(2)}$ and $\mathbf{M}^{(3)}$

Шаг 2. $\mathbf{H}^{(1,2)}_{8,1} = h\left(\mathbf{H}^{(2)}_{6,7} \| \mathbf{H}^{(1)}_{7,1}\right)$; ...;

$\mathbf{H}^{(2,3)}_{8,6} = h\left(\mathbf{H}^{(2,3)}_{7,3} \| \mathbf{H}^{(2)}_{6,2}\right)$.

Шаг 3. $\mathbf{H}^{(1,2)}_{9,1} = h\left(\mathbf{H}^{(2)}_{6,5} \| \mathbf{H}^{(2)}_{6,6} \| \mathbf{H}^{(1,2)}_{8,1}\right)$; ...;

$\mathbf{H}^{(2)}_{9,12} = h\left(\mathbf{H}^{(2)}_{6,4} \| \mathbf{H}^{(2)}_{6,5}\right)$.

Шаг4. $\mathbf{H}^{(1,2)}_{10,1} = h\left(\mathbf{H}^{(2)}_{9,12} \| \mathbf{H}^{(1,2)}_{9,1} \| \mathbf{H}^{(1,2)}_{9,2}\right)$; ...;

$\mathbf{H}^{(2,3)}_{10,6} = h\left(\mathbf{H}^{(2,3)}_{9,10} \| \mathbf{H}^{(2,3)}_{9,11} \| \mathbf{H}^{(2)}_{9,12}\right)$.

Шаг 5. $\mathbf{H}^{(1,2,3)}_{11,1} = h\left(\mathbf{H}^{(1,2)}_{10,1} \| \mathbf{H}^{(1,2)}_{10,2} \| \mathbf{H}^{(1,3)}_{10,3} \| \mathbf{H}^{(1,3)}_{10,4} \| \mathbf{H}^{(2,3)}_{10,5} \| \mathbf{H}^{(2,3)}_{10,6}\right)$.

Получим блок со значением $\mathbf{H}^{(1,2,3)}_{11,1}$ хэш-функции, применяемой к элементам из каждого блока данных $\mathbf{M}^{(1)}$, $\mathbf{M}^{(2)}$ и $\mathbf{M}^{(3)}$. Разместим полученные значения хэш-функции, как показано на рисунке 4.

Таким образом, k-й блок со значением $\mathbf{H}^{(1,2,3)}_{11,1}$ хэш-функции, которая применяется к элементам из каждого блока данных $\mathbf{M}^{(\varphi)}$, является общим для всех блоков данных $\mathbf{M}^{(\varphi)}_{i,j}$ ($i = 1, 2, ..., n; j =, 1\ 2$), подлежащим защите. При этом верхний индекс φ вычисляемых блоков со значениями $\mathbf{H}^{(\varphi)}$ хэш-функции определяется верхними индексами блоков со значениями, к которым применяется хэш-функция.

Полученная схема применения хэш-функции для контроля целостности блока данных $\mathbf{M}$ представлена на рисунке 5.

Схема применения хэш-функции для контроля целостности блока данных $\mathbf{M}$, представленная на рисунке 5, является промежуточной, составной частью схемы контроля целостности данных на основе правил построения треугольника Серпинского (рис. 1).

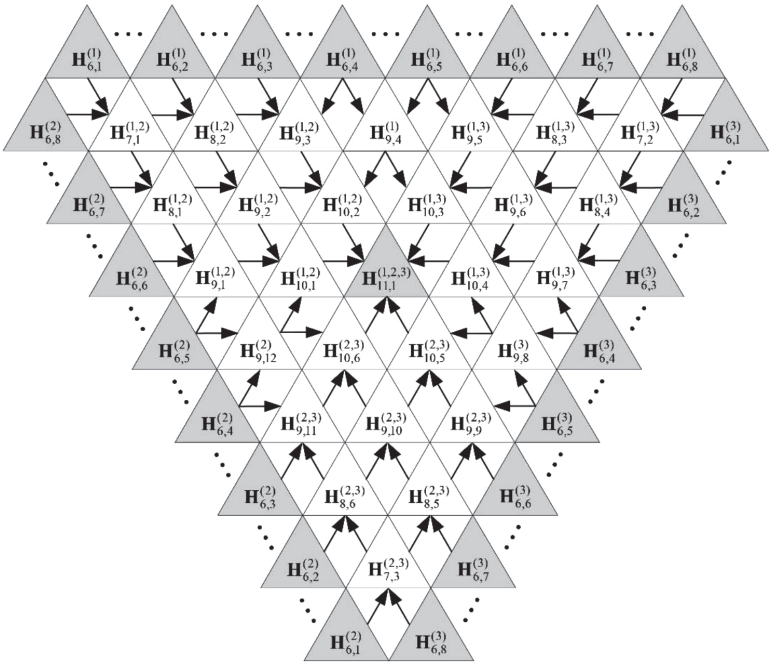


Рис. 4. Схема, иллюстрирующая порядок вычисления значения $\mathbf{H}_{11,1}^{(1,2,3)}$ хэш-функции h

Fig. 4. Scheme illustrating the order of calculation of the value $\mathbf{H}_{11,1}^{(1,2,3)}$ hash functions h

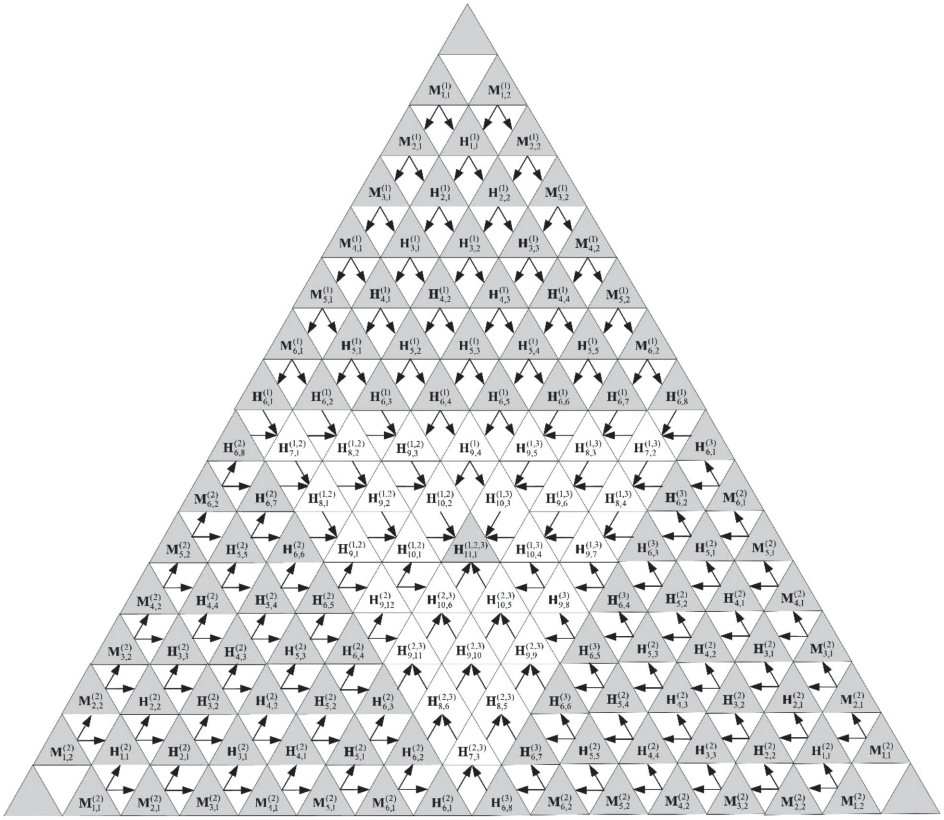


Рис. 5. Схема применения хэш-функции для контроля целостности блока данных $\mathbf{M}$

Fig. 5. Scheme of applying the hash function to control the integrity of the data block $\mathbf{M}$

Один или несколько (зависит от значения n) блоков со значениями $\mathbf{H}^{(1,2,3)}$ хэш-функции являются определяющими для элементов схем применения хэш-функции ко всем блокам данных $\mathbf{M}^{(1)}$, $\mathbf{M}^{(2)}$ и $\mathbf{M}^{(3)}$, подлежащим защите.

Контроль целостности блоков данных $\mathbf{M}_{i,j}^{(\varphi)}$ ($i = 1, 2, ..., n; j = 1, 2$), подлежащих защите, осуществляется путем сравнения значений $\mathbf{H}^{(1,2,3)}$ хэш-функции, вычисленных при запросе на использование защищаемых данных, с их эталонными значениями, вычисленными ранее и хранящимися в надежной среде.

При контроле целостности данных в случае несовпадения вычисляемого и эталонного значения $\mathbf{H}^{(1,2,3)}$ хэш-функции принимается решение о нарушении целостности блока данных $\mathbf{M}$, подлежащего защите.

Пример 2.

Дано: блок данных $\mathbf{M}$ с признаками нарушения целостности.

Требуется: обнаружить и локализовать блок данных $\mathbf{M}^{(\varphi)}(\varphi = 1, 2, 3)$ с признаками нарушения целостности.

Исходные данные взяты из Примера 1.

Для этого согласно следующему алгоритму обнаруживаются и локализуются блоки со значениями $\mathbf{H}_{i,j}^{(\varphi)}$ ($i = n; j = 1, 2, ..., n, n + 1, n + 2$) хэш-функции.

Шаг 1. Проверка блоков со значениями хэш-функции, полученных для вычисления $\mathbf{H}_{11,1}^{(1,2,3)}$.

1.1. Сравниваются вычисленные и эталонные значения $\mathbf{H}_{10,1}^{(1,2)}$, $\mathbf{H}_{10,2}^{(1,2)}$, $\mathbf{H}_{10,3}^{(1,3)}$, $\mathbf{H}_{10,4}^{(1,3)}$, $\mathbf{H}_{10,5}^{(2,3)}$, $\mathbf{H}_{10,6}^{(2,3)}$ к блокам которых применялась хэш-функция для вычисления $\mathbf{H}_{11,1}^{(1,2,3)}$.

1.2. Определяется значение, не соответствующее эталонному (к примеру, это блок со значением $\mathbf{H}_{10,3}^{(1,3)}$).

1.3. Принимается решение, что блок данных с признаками нарушения целостности находится в блоках данных $\mathbf{M}^{(1)}$ или $\mathbf{M}^{(3)}$.

Шаг 2. Проверка блоков со значениями хэш-функции, полученных для вычисления $\mathbf{H}_{10,3}^{(1,3)}$.

2.1. Сравниваются вычисленные и эталонные значения $\mathbf{H}_{9,4}^{(1)}$, $\mathbf{H}_{9,5}^{(1,3)}$, $\mathbf{H}_{9,6}^{(1,3)}$, к блокам которых применялась хэш-функция для вычисления $\mathbf{H}_{10,3}^{(1,3)}$.

2.2. Определяется значение, не соответствующее эталонному (к примеру, это блок со значением $\mathbf{H}_{9,6}^{(1)}$).

Шаг 3. Проверка блоков со значениями хэш-функции, полученных для вычисления $\mathbf{H}_{9,6}^{(1)}$.

3.1. Сравниваются вычисленные и эталонные значения $\mathbf{H}_{8,3}^{(1,3)}$, $\mathbf{H}_{8,4}^{(1,3)}$, к блокам которых применялась хэш-функция для вычисления $\mathbf{H}_{9,6}^{(1)}$.

3.2. Определяется значение, не соответствующее эталонному (к примеру, это блок со значением $\mathbf{H}_{8,3}^{(1,3)}$).

Шаг 4. Проверка блоков со значениями хэш-функции, полученных для вычисления $\mathbf{H}_{8,3}^{(1,3)}$.

4.1. Сравниваются вычисленные и эталонные значения $\mathbf{H}_{6,7}^{(1)}$, $\mathbf{H}_{7,2}^{(1,3)}$, к блокам которых применялась хэш-функция для вычисления $\mathbf{H}_{8,3}^{(1,3)}$.

4.2. Определяются значения, не соответствующие эталонным (к примеру, это блоки со значениями $\mathbf{H}_{6,7}^{(1)}$, $\mathbf{H}_{7,2}^{(1,3)}$).

4.3. Найден первый блок со значением $\mathbf{H}_{6,7}^{(1)}$ хэш-функции, не соответствующим эталонному, расположенный на нижней стороне треугольника, описывающего блок данных $\mathbf{M}^{(1)}$.

Шаг 5. Проверка блоков со значениями хэш-функции, полученных для вычисления $\mathbf{H}_{7,2}^{(1,3)}$.

5.1. Сравниваются вычисленные и эталонные значения $\mathbf{H}_{6,8}^{(1)}$, $\mathbf{H}_{6,1}^{(3)}$, к блокам которых применялась хэш-функция для вычисления $\mathbf{H}_{7,2}^{(1,3)}$.

5.2. Определяется значение, не соответствующее эталонному (к примеру, это блок со значением $\mathbf{H}_{6,8}^{(1)}$).

5.3. Найден второй блок со значением $\mathbf{H}_{6,8}^{(1)}$ хэш-функции, не соответствующим эталонному, расположенный на нижней стороне треугольника, описывающего блок данных $\mathbf{M}^{(1)}$.

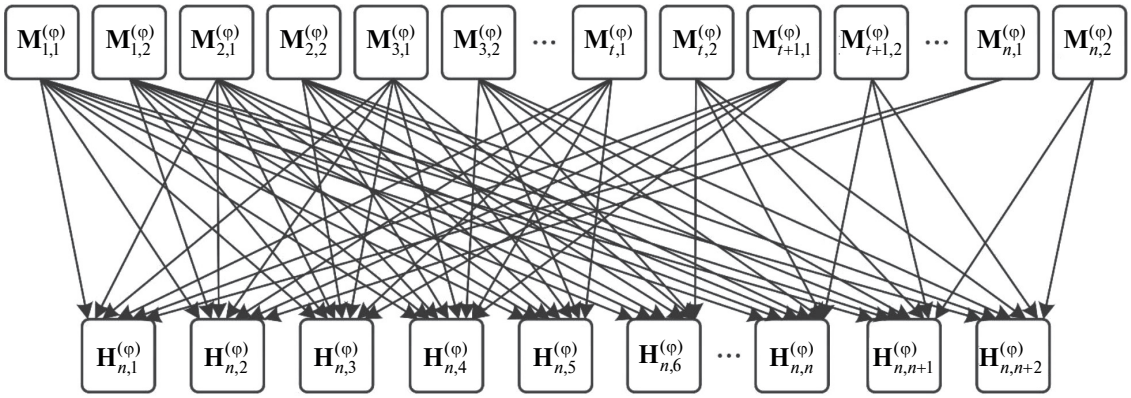


Рис. 6. Общий вид сети хэширования для схемы применения хэш-функции

Fig. 6. General view of the hash network for the hash function application scheme

Таким образом, принимается решение, что блок данных с признаками нарушения целостности находится в блоке данных $\mathbf{M}^{(1)}$, о чем свидетельствуют блоки со значениями $\mathbf{H}^{(1)}_{6,7}$, $\mathbf{H}^{(1)}_{6,8}$, которые не соответствуют эталонным.

Обнаружение и локализация блока данных $\mathbf{M}^{(\varphi)}_{i,j}$ ($i = 1, 2, ..., n; j = 1, 2$) с признаками нарушения целостности, находящегося в локализованном блоке данных $\mathbf{M}^{(\varphi)}$, после обнаружения и локализации блоков со значениями $\mathbf{H}^{(\varphi)}_{i,j}$ ($i = n; j = 1, 2, ..., n, n + 1, n + 2$), которые не соответствуют эталонным, выполняется по правилам, представленным в [21]:

Шаг 1. Строится сеть хэширования.

Сеть хэширования для схемы применения хэш-функции к блокам данных $\mathbf{M}^{(\varphi)}_{i,j}$, основанной на правилах построения треугольника Паскаля (рис. 2), представлена на рисунке 6.

Шаг 2. На основе сети хэширования составляется таблица синдромов для обнаружения и локализации блока данных $\mathbf{M}^{(\varphi)}_{i,j}$ с признаками нарушения целостности.

Пример 3.

Дано: блок данных $\mathbf{M}^{(1)}$ с признаками нарушения целостности. Блоки со значениями $\mathbf{H}^{(1)}_{6,7}$, $\mathbf{H}^{(1)}_{6,8}$ не соответствуют эталонным.

Требуется: обнаружить и локализовать блок данных $\mathbf{M}^{(1)}_{i,j}$ ($i = 1, 2, ..., n; j = 1, 2$) с признаками нарушения целостности.

Исходные данные взяты из Примера 2.

В соответствии с [21] принимается решение, что блоком данных с признаками нарушения целостности является блок данных $\mathbf{M}^{(1)}_{6,2}$, о чем свидетельствуют блоки со значениями $\mathbf{H}^{(1)}_{6,7}$, $\mathbf{H}^{(1)}_{6,8}$, не соответствующие эталонным.

Заключение

Разработан способ контроля целостности данных в СХД на основе правил построения треугольника Серпинского. Представленные механизмы защиты обладают свойством самоподобия и, как следствие, их применение при масштабировании СХД в условиях деструктивных воздействий злоумышленника и возмущений среды функционирования, приводящих к нарушению целостности данных, позволит осуществить контроль их целостности без введения дополнительной контрольной информации, прямо пропорциональной объему увеличиваемых данных.

СПИСОК ИСТОЧНИКОВ

1. **Клеменков П. А**. Большие данные: современные подходы к хранению и обработке / П. А. Клеменков, С. Д. Кузнецов // Труды Института системного программирования РАН.— 2012.— Т. 23.— С. 143–158.

2. **Ямашкин С. А**. Интеграция, хранение и обработка больших массивов пространственно-временной информации в цифровых инфраструктурах пространственных данных / С. А. Ямашкин, А. А. Ямашкин // Современные наукоемкие технологии.— 2021.— № 5.— С. 108–113.

3. **Omondi A.** Residue Number System: Theory and Implementation. Imperial Collegt Press, London, 2007.— 296 p.

4. **Schneier B.** Applied Cryptography Second Edition: protocols, algorithms and source code in C. John Wiley & Sons, Inc, 2016.— 653 p.

5. **Menezes A. J**. Handbook of Applied Cryptography. CRC Press, Inc, 2015.— 770 p.

6. **Knuth D. E**. The Art of Computer Programming: Volume 3: Sorting and Searching. 2nd edn, 2018.— 803 p.

7. **Диченко С. А**. Безопасные генераторы псевдослучайных линейных последовательностей на арифметических полиномах для защищенных систем связи / С. А. Диченко, О. А. Финько // Нелинейный мир.— 2013.— Т. 11.— № 9.— С. 632–645.

8. **Biham E.** A framework for iterative hash functions. HAIFA. ePrint Archive, Report 2007/278, 2007.— pp. 1–20.

9. **Wang X.** How to break MD5 and Other Hash Function. EUROCRYPT. LNCS3494.— Springer-Verlag, 2005.— P. 19–35.

10. **Bellare M.** New Proofs for NMAC and HMAC: Security without Collision-Resistance. CRYPTO. ePrint Archive, Report 2006/043, 2006.— P. 1–16.

11. **Диченко С. А**. Контроль и восстановление целостности многомерных массивов данных посредством криптокодовых конструкций // С. А. Диченко, О. А. Финько // Программирование.— 2021.— № 6.— С. 3–15.

12. **Tchernykh A.** AC-RRNS: Anti-Collusion Secured Data Sharing Scheme for Cloud Storage / A. Tchernykh, M. Babenko, N. Chervyakov // International Journal of Approximate Reasoning. Special Issue on Uncertainty in Cloud Computing: Concepts, Challenges and Current Solutions.— 2018.— Vol. 102.— P. 60–73.

13. **Tilborg H.** Fundamentals of cryptology: a professional reference and interactive tutorial. Kluwer Academic Publishers. London, 2014.— 414 p.

14. **Диченко С. А**. Снижение вводимой избыточности при обеспечении устойчивости информационно-аналитических систем в условиях компенсации последствий деструктивных воздействий злоумышленника / С. А. Диченко, О. А. Финько // Автоматизация процессов управления.— 2020.— № 4 (62).— С. 38–48.

15. **Диченко С. А**. Контроль и восстановление целостности данных в защищенных информационно-аналитических системах / С. А. Диченко, О. А. Финько // Труды Военно-космической академии имени А. Ф. Можайского.— 2021.— № 676.— С. 36–49.

16. **Диченко С. А**. Обобщенный способ применения хэш-функции для контроля целостности данных / С. А. Диченко, О. А. Финько // Наукоемкие технологии в космических исследованиях Земли.— 2020.— Т. 12.— № 6.— С. 48–59.

17. **Dichenko S.** Two-dimensional control and assurance of data integrity in information systems based on residue number system codes and cryptographic hash functions / S. Dichenko, O. Finko // В сборнике: Integrating Research Agendas and Devising Joint Challenges. International Multidisciplinary Symposium ICT Research in Russian Federation and Europe.— 2018.— P. 139–146.

18. **Серпинский В. Ф.** О решении уравнений в целых числах. М.: Государственное издательство физико-математической литературы, 1961.— 88 с.

19. **Пайтген Х. О**. Красота фракталов. М.: Мир, 1993.— 176 с.

20. **Гарднер М.** Неисчерпаемое очарование треугольника Паскаля // Математические новеллы. М.: Мир, 1974.— 456 с.

21. **Диченко С. А**. Модель контроля целостности многомерных массивов данных / С. А. Диченко // Проблемы информационной безопасности. Компьютерные системы.— 2021.— № 2 (46).— С. 97–103.

REFERENCES

1. **Klemenkov P. A**. Bol'shie dannye: sovremennye podhody k hraneniyu i obrabotke [Big data: modern approaches to storage and processing] / P. A. Klemenkov, S. D. Kuznecov // Trudy Instituta sistemnogo programmirovaniya RAN.— 2012.— Vol. 23.— P. 143–158. (In Russian)

2. **Yamashkin S. A**. Integraciya, hranenie i obrabotka bol'shih massivov prostranstvenno-vremennoj informacii v cifrovyh infrastrukturah prostranstvennyh dannyh [Integration, storage and processing of large arrays of spatiotemporal information in digital spatial data infrastructures] / S.A.YAmashkin, A.A.YAmashkin // Sovremennye naukoemkie tekhnologii.— 2021.— No 5.— P. 108–113. (In Russian)

3. **Omondi A.** Residue Number System: Theory and Implementation. Imperial Collegt Press, London, 2007.— 296 p.

4. **Schneier B.** Applied Cryptography Second Edition: protocols, algorithms and source code in C. John Wiley & Sons, Inc, 2016.— 653 p.

5. **Menezes A. J.** Handbook of Applied Cryptography. CRC Press, Inc, 2015.— 770 p.

6. **Knuth D. E.** The Art of Computer Programming: Volume 3: Sorting and Searching. 2nd edn, 2018.— 803 p.

7. **Dichenko S. A**. Bezopasnye generatory pseudosluchajnyh linejnyh posledovatel'nostej na arifmeticheskih polinomah dlya zashchishchennyh sistem svyazi [Secure generators of pseudorandom linear sequences on arithmetic polynomials for secure communication systems] / S. A. Dichenko, O. A. Fin'ko // Nelinejnyj mir.— 2013.— Vol. 11.— No 9.— P. 632–645. (In Russian)

8. **Biham E.** A framework for iterative hash functions. HAIFA. ePrint Archive, Report 2007/278, 2007.— P. 1–20.

9. **Wang X.** How to break MD5 and Other Hash Function. EUROCRYPT. LNCS3494.— Springer-Verlag, 2005.— P. 19–35.

10. **Bellare M.** New Proofs for NMAC and HMAC: Security without Collision-Resistance. CRYPTO. ePrint Archive, Report 2006/043, 2006.— P. 1–16.

11. **Dichenko S. A**. Kontrol' i vosstanovlenie celostnosti mnogomernyh massivov dannyh posredstvom kriptokodovyh konstrukcij [Control and restoration of the integrity of multidimensional data arrays by means of cryptographic constructions] // S. A. Dichenko, O. A. Fin'ko // Programmirovanie.— 2021.— No 6.— P. 3–15. (In Russian)

12. **Tchernykh A.** AC-RRNS: Anti-Collusion Secured Data Sharing Scheme for Cloud Storage / A. Tchernykh, M. Babenko, N. Chervyakov // International Journal of Approximate Reasoning. Special Issue on Uncertainty in Cloud Computing: Concepts, Challenges and Current Solutions.— 2018.— Vol. 102.— P. 60–73.

13. **Tilborg H.** Fundamentals of cryptology: a professional reference and interactive tutorial. Kluwer Academic Publishers. London, 2014.— 414 p.

14. **Dichenko S. A**. Snizhenie vvodimoj izbytochnosti pri obespechenii ustojchivosti informacionno-analiticheskih sistem v usloviyah kompensacii posledstvij destruktivnyh vozdejstvij zloumyshlennika [Reduction of the introduced redundancy while ensuring the stability of information and analytical systems in terms of compensation for the consequences of destructive influences of the attacker] / S. A. Dichenko, O. A. Fin'ko // Avtomatizaciya processov upravleniya.— 2020.— No 4 (62).— P. 38–48. (In Russian)

15. **Dichenko S. A**. Kontrol' i vosstanovlenie celostnosti dannyh v zashchishchennyh informacionno-analiticheskih sistemah [Control and restoration of data integrity in secure information and analytical systems] / S. A. Dichenko, O. A. Fin'ko // Trudy Voenno-kosmicheskoj akademii imeni A. F. Mozhajskogo.— 2021.— No 676.— P. 36–49. (In Russian)

16. **Dichenko S. A**. Obobshchennyj sposob primeneniya hesh-funkcii dlya kontrolya celostnosti dannyh [Generalized way to use a hash function to control data integrity] / S. A. Dichenko, O. A. Fin'ko // Naukoemkie tekhnologii v kosmicheskih issledovaniyah Zemli.— 2020.— Vol. 12.— No 6.— P. 48–59. (In Russian)

17. **Dichenko S. A**. Two-dimensional control and assurance of data integrity in information systems based on residue number system codes and cryptographic hash functions / S. Dichenko, O. Finko // V sbornike: Integrating Research Agendas and Devising Joint Challenges. International Multidisciplinary Symposium ICT Research in Russian Federation and Europe.— 2018.— P. 139–146.

18. **Serpinskij V. F.** O reshenii uravnenij v celyh chislah. [On solving equations in integers] M.: Gosudarstvennoe izdatel'stvo fiziko-matematicheskoj literatury, 1961.— 88 p. (In Russian)

19. **Pajtgen H. O**. Krasota fraktalov. [The beauty of fractals] M.: Mir, 1993.— 176 p. (In Russian)

20. **Gardner M.** Neischerpaemoe ocharovanie treugol'nika Paskalya [The inexhaustible charm of Pascal's triangle] // Matematicheskie novelly. M.: Mir, 1974.— 456 p. (In Russian)

21. **Dichenko S. A**. Model' kontrolya celostnosti mnogomernyh massivov dannyh [A model for controlling the integrity of multidimensional data arrays] / S. A. Dichenko // Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy.— 2021.— No 2 (46).— P. 97–103. (In Russian)

Статья поступила в редакцию 09.02.2022; одобрена после рецензирования 16.02.2022; принята к публикации 22.02.2022.

The article was submitted 09.02.2022; approved after reviewing 16.02.2022; accepted for publication 22.02.2022.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

СОПИН Кирилл Юрьевич — адъюнкт; Краснодарское высшее военное училище имени генерала армии С. М. Штеменко

ДИЧЕНКО Сергей Александрович — к.т.н., докторант; Краснодарское высшее военное училище имени генерала армии С. М. Штеменко

САМОЙЛЕНКО Дмитрий Владимирович — д.т.н.; начальник кафедры, Краснодарское высшее военное училище имени генерала армии С. М. Штеменко

SOPIN Kirill Yu. — adjunct; Krasnodar Higher Military School named after General of the Army S. M. Shtemenko

Email: sopin.kirill2010@yandex.ru

ORCID: 0000-0002-0546-7801

DICHENKO Sergey A. — Ph.D., doctoral student; Krasnodar Higher Military School named after General of the Army S. M. Shtemenko

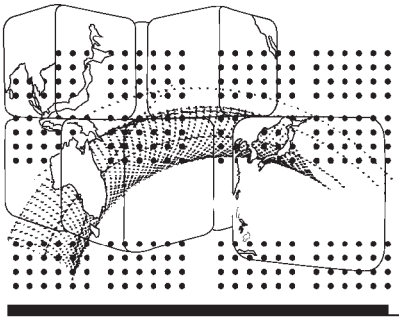
Email: dichenko.sa@yandex.ru

ORCID: 0000-0002-0644-4353

SAMOILENKO Dmitry V. — Doctor of Technical Sciences; Head of Department, Krasnodar Higher Military School named after General of the Army S. M. Shtemenko

Email: 19sam@mail.ru

ORCID: 0000-0001-9665-2174



БЕЗОПАСНОСТЬ КИБЕРФИЗИЧЕСКИХ СИСТЕМ

Научная статья
DOI 10.48612/jisp/5xdh-23hk-dmhm
УДК 004.56

В. Д. Данилов, Т. Д. Овасапян, Д. В. Иванов, А. С. Коноплев
Санкт-Петербургский политехнический университет Петра Великого (СПбПУ)
Россия, 195251, Санкт-Петербург, ул. Политехническая, д. 29

ГЕНЕРИРОВАНИЕ СИНТЕТИЧЕСКИХ ДАННЫХ ДЛЯ HONEYPOT-СИСТЕМ С ИСПОЛЬЗОВАНИЕМ МЕТОДОВ ГЛУБОКОГО ОБУЧЕНИЯ

Аннотация. В статье представлены исследования, направленные на анализ методов генерирования синтетических данных для заполнения honeypot-систем. Для выбора генерируемых типов данных выявляются актуальные целевые объекты в контексте honeypot-систем. Исследуются существующие методы генерирования. Также анализируются методы оценки качества сгенерированных данных в контексте honeypot-систем. В результате разрабатывается макет автоматизированной системы генерирования синтетических данных для honeypot-систем и производится оценка эффективности его работы.

Ключевые слова: honeypot-система, методы глубокого обучения, генерирование синтетических данных, машинное обучения, атаки логического вывода.

Финансирование: Работа выполнена в рамках Государственного задания на проведение фундаментальных исследований (код темы 0784-2020-0026)

Research article
DOI 10.48612/jisp/5xdh-23hk-dmhm

V. D. Danilov, T. D. Ovasapyan, D. V. Ivanov, A. S. Konoplev
Peter the Great St. Petersburg Polytechnic University, Russia, St. Petersburg

SYNTHETIC DATA GENERATION FOR HONEYPOT SYSTEMS USING DEEP LEARNING METHODS

Abstract. This article presents research aimed at analyzing methods for generating synthetic data to populate honeypot systems. To select the generated data types, the relevant target objects in the context of honeypot-systems are identified. Existing generation methods are investigated. Methods for evaluating the quality of generated data in the context of honeypot systems are also analyzed. As a result, a layout of an automated system for generating synthetic data for honeypot-systems is developed and its performance is evaluated.

Keywords: honeypot system, deep learning methods, synthetic data generation, machine learning, inference attacks.

Acknowledgements: The work was carried out within the framework of the State Task for conducting fundamental research (topic code 0784-2020-0026)

Актуальность исследования и создания методов, осуществляющих анализ сетевых атак и противодействие им, трудно переоценить. Наряду с этим наблюдается бурный рост сетевых атак. Так за 1 квартал 2021 года количество сетевых атак увели-

чилось на 17 % по сравнению с аналогичным промежутком времени за 2020 год [1]. Основной мотив атак — получение данных. Наибольшую угрозу представляют сетевые атаки, основанные на неизвестных уязвимостях нулевого дня (0-day). Это свидетельствует о необходимости создания методов для анализа и противодействия сетевым атакам.

Одним из методов анализа действий злоумышленников при эксплуатации неизвестных уязвимостей является использование honeypot-систем, которые позволяют изучить поведение атакующих и в дальнейшем предугадать наиболее вероятные сценарии атак. Несмотря на успешное использование данных систем, существует актуальная проблема, когда злоумышленник при атаке понимает, что система ненастоящая. Поскольку использовать реальные данные для заполнения honeypot-систем небезопасно, в рамках статьи исследуется возможность генерирования синтетических данных с помощью методов глубокого обучения для последующего применения в honeypot-системах.

1. Исследование honeypot-систем и используемых в них данных

Honeypot-система (от англ. «горшочек с медом», honeypot) — это поддельная система, используемая в качестве ловушки для злоумышленников [2, 3]. Задачи данной системы:

– убедить злоумышленника в том, что система настоящая;

– спровоцировать атакующих на вредоносные действия;

– задержать злоумышленников в системе для изучения их поведения.

Важным аспектом реализации honeypot-систем является их заполнение поддельными данными (приманка), получение и изучение которых является основной задачей злоумышленника. Требуется, чтобы эти данные не вызывали у нарушителя сомнений в их подлинности. Поскольку чем больше времени злоумышленник проведет в системе-ловушке и чем больше будет выполнено различных сценариев атак, тем больше полезной информации получит специалист по информационной безопасности. Приманками для honeypot-систем могут быть как поддельные данные и информация, которые вызывают наибольший интерес у потенциальных нарушителей, так и отдельно работающие приложения и программные средства [4].

Анализ изменений, событий внутри системы, а также входящего сетевого трафика при попытках получения доступа к приманкам позволяет:

1. Узнать местонахождение злоумышленника.

2. Изучить методы эксплуатации уязвимостей.

3. Оценить эффективность используемых мер безопасности.

4. Оценить степень угрозы со стороны злоумышленников.

5. Узнать, какие данные и приложения подвергаются атакам чаще других.

По данным экспертного центра кибербезопасности Positive Technologies [1] чаще всего атакам подвергаются госучреждения, промышленные компании и организации в сфере науки и образования. Основным мотивом в атаках как на организации, так и на частных лиц остается получение данных. Главными целями злоумышленников являются персональные и учетные данные, а при атаках на организации к ним добавляется еще и коммерческая тайна. На рисунке 1 представлено соотношение по типам украденных данных для атак на организации, на рисунке 2 — для атак на частных лиц.

Также растет количество атак на промышленные предприятия. Согласно другому исследованию компании Positive Technologies, за 2020 год число атак, направленных на объекты промышленности, выросло на 91 % относительно 2019 года [5]. Атаки на данный сегмент являются наиболее опасными, поскольку влекут за собой последствия на работу объектов критически значимой инфраструктуры [6–8].

В таблице 1 представлены типы информации, подвергаемой наибольшему количеству атак.

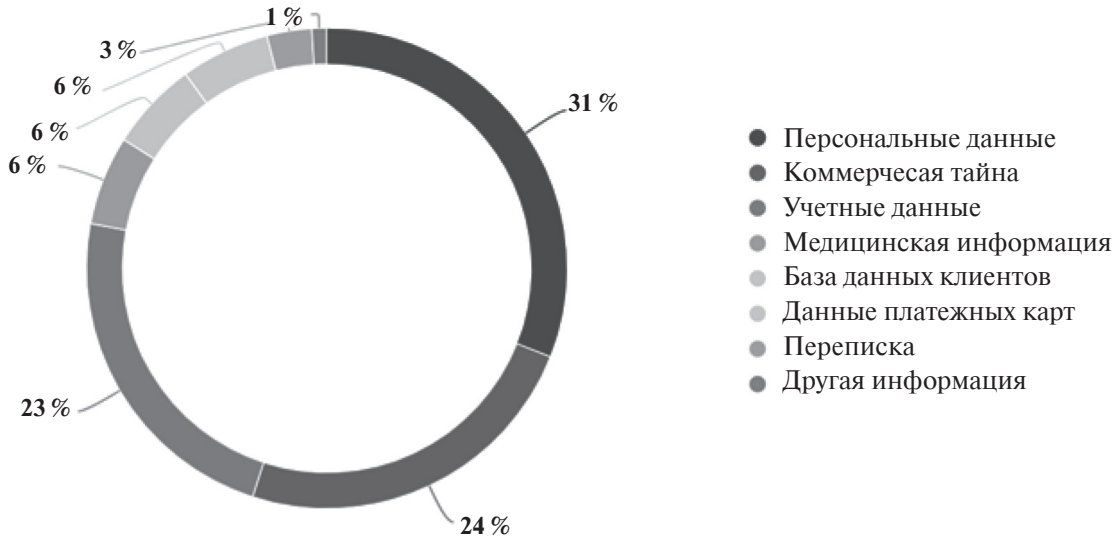


Рис. 1. Соотношение по типам украденных данных для атак на организации

Fig. 1. Attacks on organizations correlation by type of stolen data

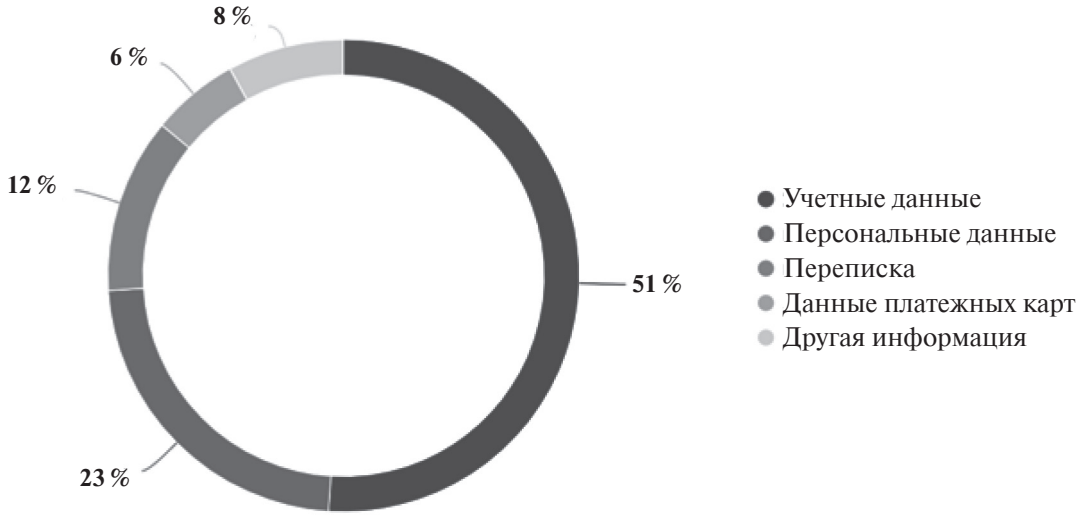


Рис. 2. Соотношение по типам украденных данных для атак на частных лиц

Fig. 2. Attacks on individuals correlation by types of stolen data

Таблица 1. Информация, подвергаемая наибольшему количеству атак

Table 1. Information subjected to the largest number of attacks

Тип данных	Формат хранения данных
Персональные данные	Изображения, текст
Учетные данные	Текст в виде таблицы
Медицинская информация	Текст, числа в виде таблице
Данные платежных карт	Текст
Переписка	Текст
База данных клиентов	Текст, числа в виде таблицы

Таблица 2. Возможность применения методов генерации для каждого из типа данных

Table 2. Possibility of applying generation methods for each of the data types

Тип данных	Статистическая модель	GAN (RNN)	GAN(CNN)
Персональные данные	–	–	+
Медицинская информация	+	+	+
Сетевой трафик АСУ ТП	–	+	+

Генерация типов данных, которые имеют одинаковый формат хранения, будет проходить по одной и той же схеме. Также необходимо учитывать, что для некоторых типов применение методов генерации синтетических данных является непрактичным, поскольку их содержимое является случайным. Поэтому для дальнейшего исследования выбраны типы данных, которые имеют уникальные форматы хранения, а также применимы для методов генерирования синтетических данных:

1. Персональные данные — хранение изображений людей.

2. Медицинская информация — хранение текстовой информации и числовых значений в виде таблиц.

3. Сетевой трафик АСУ ТП — хранение сетевого трафика взаимодействия с промышленным устройством.

2. Выбор методов генерирования синтетических данных

Данные, которые будут генерироваться для honeypot-систем, имеют различные форматы хранения. В связи с этим для каждого типа данных требуется выбрать соответствующий наиболее подходящий метод генерирования.

Статистическая модель наибольшим образом применяется для данных, которые имеют связи и зависимости между значениями различных переменных. Среди используемых данных под эти требования подходит медицинская информация, в которой можно проследить закономерности между некоторыми атрибутами (диагноз, возраст, пол и т.д.).

Сетевой трафик также имеет различные зависимости среди параметров, однако в нем содержатся поля «данные» TCP-пакетов, значения которых не являются конечным дискретным множеством. Решение задачи представления зависимости этого поля по отношению к другим параметрам с помощью статистической модели является неэффективным.

Генеративно-состязательные сети можно применить почти к любым типам данных за счет поиска зависимостей между атрибутами конечного дискретного множества и добавления шумов к атрибутам с непрерывными значениями. Эффективность сгенерированных данных напрямую зависит от вида нейронных сетей, используемых в качестве дискриминатора и генератора [9, 10]. Исходя из своих свойств для работы с изображениями лучше других методов подходят сверточные нейронные сети. Для работы с табличными данными возможно применение как сверточных, так и рекуррентных нейронных сетей.

Таким образом, для каждого из типов, данных, генерация которых осуществляется в рамках статьи, выбраны соответствующие методы (табл. 2).

3. Анализ методов оценки качества сгенерированных данных

Основными задачами сгенерированных данных в контексте honeypot-систем являются привлечение потенциальных нарушителей для совершения нелегитимных действий, а также невозможность раскрытия исходных данных, из которых были получены синтетические. Для этого необходимо проанализировать возможные атаки, направленные на получение реальных данных, а также исследовать метрики, помогающие определить схожесть поддельных и реальных данных по их свойствам [11].

Далее представлено исследование логических атак, направленных на получение реальных данных:

1. Атака установления членства.

Серьезной проблемой конфиденциальности в контексте использования синтетических данных является риск возможности проведения атак логического вывода [12]. Одна из таких атак — это атака установления членства [13]. Атака установления членства направлена на прикрепление личности к предположительно обезличенной записи или на установление факта, что конкретная запись присутствует в конфиденциальном наборе данных. В рамках использования генеративно-состязательных сетей конфиденциальными данными является обучающая выборка.

Предполагается, что единственная информация доступная злоумышленнику для проведения атаки — это синтетические данные, которые являются результатом использования генеративных методов. Злоумышленник не имеет доступа к параметрам целевой модели генерирования данных.

Для оценки возможности проведения атаки считается, что атакующий владеет набором данных $X = \{x_1, ..., x_n\}$ с записями, которые предположительно участвовали в обучении целевой модели (где n-количество записей в наборе данных). Однако злоумышленник не обладает знаниями о том, как обучающий набор был построен из X. Таким образом, он не имеет доступа к реальным записям обучающего набора данных, поэтому не может обучить модель (аналогичную атакуемой) с использованием дискриминационного подхода. Вместо этого злоумышленник обучает GAN с использованием синтетических данных, чтобы добиться воссоздания целевой модели локально.

Прежде всего требуется формализовать атаку установления членства. Для этого атака представляется как задача бинарной классификации, в которой злоумышленник стремится определить — использовался ли образец x для обучения генеративной модели жертвы. Формально атаку можно представить функцией (1).

$$A:(x, M(\theta)) \rightarrow \{0, 1\}, \quad (1)$$

где A — модель атакующего, которая выводит 1, если злоумышленник делает вывод, что образец x содержится в обучающем наборе и 0 — в противном случае. Переменная θ обозначает параметры модели жертвы, а M — представляет тип доступа, доступный злоумышленнику. В рамках статьи рассматривается атака черного ящика, поэтому в данном случае M — это доступ к синтетическим данным.

С байесовской точки зрения злоумышленник стремится вычислить вероятность (2).

$$P\left(x \in D_{train} | x, \right), \quad (2)$$

где D_{train} — обучающий набор данных.

Далее требуется спрогнозировать, что целевая запись содержится в обучающем наборе в случае, если логарифмическая функция правдоподобия неотрицательна, то есть целевая запись скорее будет содержаться в обучающем наборе, чем не будет. Математически модель атакующего можно свести к формуле (3).

$$A\left(x, M\left(\theta\right)\right) = 1_A\left[\log\frac{P\left(x \in D_{train} | x, \right)}{P\left(x \notin D_{train} | x, \right)} \geq 0\right], \quad (3)$$

где 1_A — индикаторная функция, а обучающая выборка пользователя обозначается как D_{train} (конфиденциальная информация).

Также обозначим запрос атакующего (4), который содержит как записи, участвующие в обучающей выборке (5), так и не участвующие в обучении записи (6).

$$S = \left\{\left(x_i, m_i\right)\right\}_{i=1}^{N}, \quad (4)$$

$$x_i \in D_{train}, m_i = 1, \quad (5)$$

$$x_i \notin D_{train}, m_i = 0, \quad (6)$$

где m — переменная установления членства.

Истинно положительные попытки определения членства можно обозначить как (7), истинно отрицательные как (8)

$$E_{x_i}\left[P\left(A\left(x_i, M\left(\theta\right)\right) = 1 | m_i = 1\right)\right], \quad (7)$$

$$E_{x_i}\left[P\left(A\left(x_i, M\left(\theta\right)\right) = 0 | m_i = 0\right)\right]. \quad (8)$$

В общем виде атака установления членства сводится к следующим действиям:

1. Выбор модели дискриминатора для GAN.

2. Обучение дискриминатора синтетическими данными, полученными из honeypot-системы.

3. Выбор записей, которые предположительно содержатся в обучающем наборе данных исходной модели.

4. Вычисление вероятности принадлежности записей из п. 3 с помощью дискриминатора из п. 2.

5. Формирование бинарного ответа об установлении членства записей из п. 3 в обучающей выборке (содержится запись в обучающей выборке или не содержится).

Схема проведения атаки установления членства представлена на рисунке 3.

2. Атака установления атрибутов.

Другая атака логического вывода — это атака установления атрибутов [14]. Атака установления атрибутов подразумевает, что злоумышленник обладает информацией о значениях некоторых атрибутов конкретной записи, участвующей в обучающей выборке. Цель проведения данной атаки — узнать значения неизвестных атрибутов путем анализа синтетических данных. Для дальнейшей оценки возможности проведения данной атаки требуется сформулировать сценарий ее проведения.

Изначально необходимо случайным образом выбрать некоторое количество записей из обучающей выборки. Для каждой записи *r* случайным образом выбираются атрибуты *s*, которые как будто известны злоумышленнику. Далее требуется обучить классификатор на основе синтетических данных, доступных злоумышленнику. После чего с помощью обученного классификатора предсказать значения тех атрибутов, значения которых являются неизвестными для злоумышленника. Затем необходимо рассчитать точность полученных прогнозов, сравнив предсказанные значения с реальными, которые были использованы в обучении исходной модели. В качестве классификаторов будут использованы следующие алгоритмы — наивный байесовский классификатор, метод K-ближайших соседей и случайный лес. Общая схема проведения атаки представлена на рисунке 4 [15].

Второй метод оценки качества сгенерированных данных — это оценка сходства поддельных данных с реальными.

Проведение оценки сходства реальных и поддельных данных возможно с использованием следующих методов:

1. Сравнение статистических показателей. Идея оценки сходства данных на основе сравнения статистических показателей заключается в нахождении частот для каждого уникального значения всех атрибутов. Требуется вычислить в процентном соотношении вероятность появления того или иного значения для всех атрибутов в целевом наборе данных. Данный критерий применим к табличным данным, значения которых образуют конечное дискретное множество.

2. Классификация с помощью алгоритмов машинного обучения — обучение классификатора на синтетических и реальных

Рис. 3. Схема проведения атаки установления членства

Fig. 3. Membership Attack Conducting Scheme

Рис. 4. Схема проведения атаки установления атрибутов

Fig. 4. Attribute Establishment Attack Scheme

данных и последующее прогнозирование реальности записей тестируемого набора данных. Метод применим для всех типов данных, используемых в рамках статьи.

3. Соответствие спецификации протокола — проверка синтетического сетевого трафика на соответствие протоколу с помощью анализаторов сетевого трафика. Данный метод является узконаправленным и будет применен только для сетевого трафика устройств АСУ ТП.

3. Разработка макета автоматизированной системы генерирования синтетических данных для honeypot-систем

Архитектура разработанного макета состоит из следующих модулей:

1. Модуль предобработки входных данных — данный модуль отвечает за преобразование входных «сырых» данных в формат, используемый в алгоритмах машинного обучения и методах генерирования. Для персональных данных — это изменение высоты и ширины изображений. Для медицинской информации и сетевого трафика — конвертация в формат «csv».

2. Модуль генерирования персональных данных — генерирование поддельных изображений с помощью генеративно-состязательных сетей, построенных на CNN. Сначала осуществляется обучение GAN, затем непосредственно генерирование.

3. Модуль генерирования медицинской информации — генерирование поддельной медицинской информации с помощью статистической модели и генеративно-состязательных сетей, построенных на RNN и CNN. Первый шаг — это нахождение статистических зависимостей и обучение GAN, второй шаг — генерирование поддельных данных.

4. Модуль генерирования сетевого трафика — генерирование поддельного сетевого трафика взаимодействия устройств АСУ ТП с помощью генеративно-состязательных сетей, построенных на RNN и CNN. Сначала сети обучаются, после чего генерируют поддельный трафик.

5. Модуль проведения атаки — данный модуль используются для оценки качества сгенерированных данных на возможность восстановления исходных данных из синтетических. В зависимости от типа данных проводится атака установления членства и/или атака установления атрибутов.

6. Модуль оценки сходства — модуль предназначен для оценки качества сгенерированных данных с точки зрения сходства с реальными данными.

7. Модуль проверки соответствия спецификации — функционал модуля заключается в проверке сгенерированного сетевого трафика на соответствие спецификации протокола.

8. Модуль проверки требований качества синтетических данных — полученные в ходе оценки эффективности метрики сравниваются с требованиями по качеству синтетических данных. В случае, когда сгенерированные данные не соответствуют требуемому качеству, осущест-

вляется повторное генерирование данных, то есть возвращение к модулям из п. 2–4. Иначе — данные записываются в базу данных для дальнейшего использования в honeypot-системах.

Общая схема архитектуры разработанного макета представлена на рисунке 5.

Все описанные модули были реализованы с использованием языка программирования Python 3.7. В основе модуля генерации персональных данных была использована библиотека StarGAN. Генерация медицинской информации и сетевого трафика были реализованы с использованием библиотек TGAN и TableGAN. TGAN в качестве генератора и дискриминатора использует рекуррентные нейронные сети, а TableGAN — сверточные нейронные сети соответственно.

Используемые наборы данных для обучения и последующего генерирования синтетических данных:

1. Персональные данные — датасет «CelebA» [16]. Набор изображений лиц известных людей, который содержит более 200 тысяч различных изображений.

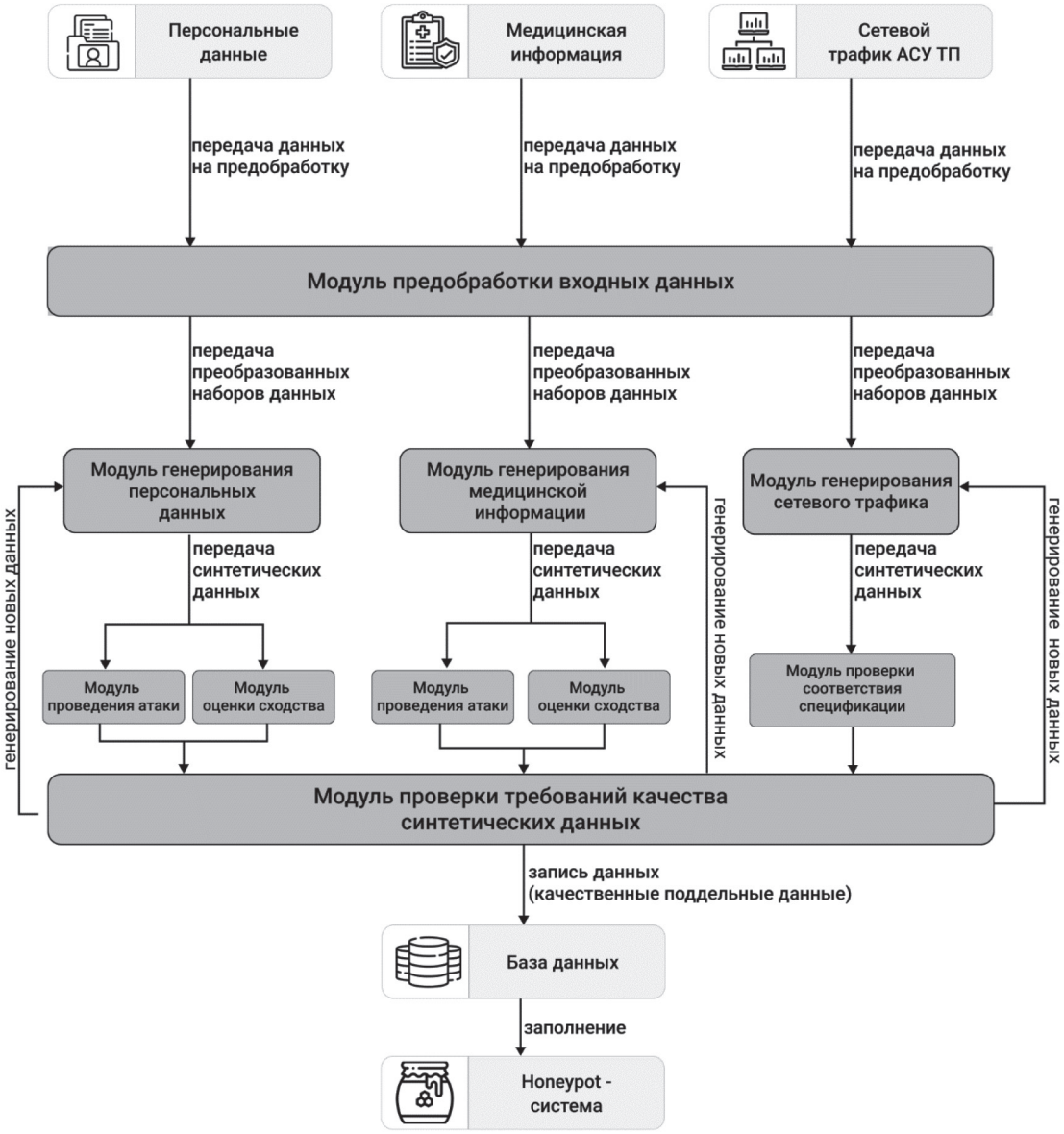


Рис. 5. Общая схема архитектуры разработанного макета

Fig. 5. General scheme of the developed layout architecture

2. Медицинская информация — данные о выписке из больницы из Департамента здравоохранения штата Техас [17]. Набор данных содержит записи о выписках с 2007 по 2015 года. Для анализа были выбраны записи за 2013 год. Набор данных состоит из более, чем 700 тысяч записей, каждая из которых содержит в себе 194 различных атрибута.

3. Сетевой трафик АСУ ТП. В открытом доступе не было обнаружено наборов данных с сетевым трафиком устройств АСУ ТП. Поэтому был сформирован собственный датасет. Для этого с помощью анализатора Wireshark был собран сетевой трафик взаимодействия ПО TIA Portal с панелью оператора Siemens HMI Comfort Panel.

Оценка эффективности разработанного макета генерации поддельных данных для honeypot-системы носит эмпирический характер и была получена на основе тестирования входных данных с различными параметрами макета и использованными методами генерирования. Для каждого выбранного типа данных были сгенерированы поддельные выборки различных размеров, после чего были произведены атаки, а также применены методы оценки сходства с реальными данными.

Для атаки установления членства точность измеряется в отношении количества правильно классифицированных записей о принадлежности обучающему набору к общему числу тестируемых записей из обучающего датасета.

Для атаки установления атрибутов точность проведения атаки измеряется в отношении количества правильно спрогнозированных значений атрибутов к общему числу неизвестных атрибутов.

В статье продемонстрированы наилучшие результаты, которые были получены в ходе тестирования.

На рисунке 6 представлена точность проведения атаки установления членства для сгенерированных персональных данных с помощью GAN с использованием CNN.

На рисунке 7 представлена оценка сходства поддельных персональных данных и реальных. Чем меньше точность классификации поддельных данных, тем больше они похожи на реальные.

В случае генерирования поддельной медицинской информации наилучшие результаты были получены при использовании генеративно-состязательных сетей, построенных на рекуррентных нейронных сетях. На рисунке 8 представлена точность

Рис. 6. Точность проведения атаки установления членства для сгенерированных персональных данных

Fig. 6. Accuracy of a membership attack on generated personal data

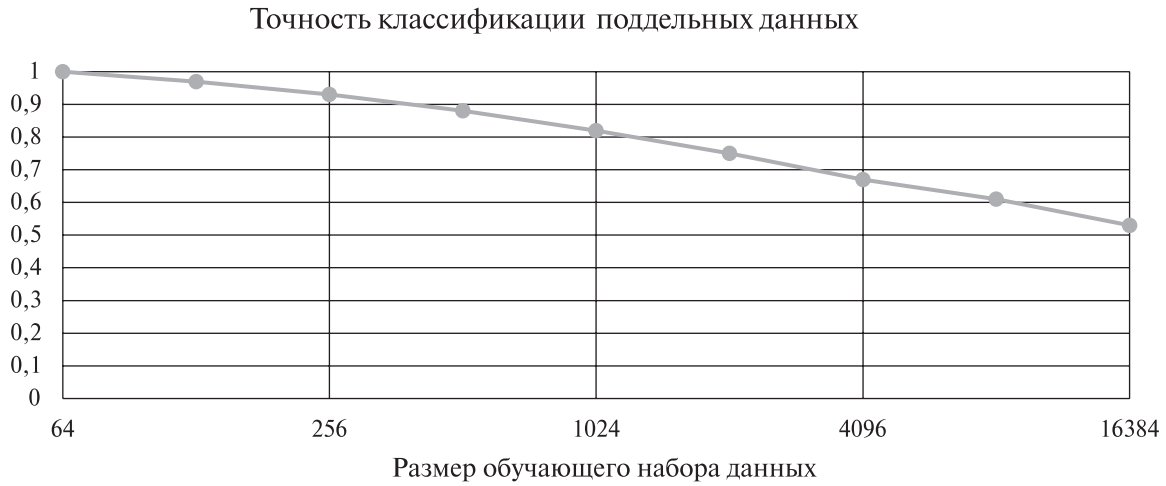


Рис. 7. Оценка сходства реальных и поддельных сгенерированных персональных данных

Fig. 7. Similarity assessment of real and fake generated personal data

Рис. 8. Точность проведения атаки установления членства для сгенерированной медицинской информации

Fig. 8. Accuracy of a Membership Attack on Generated Health Information

проведения атаки установления членства, на рисунке 9 — точность проведения атаки установления атрибутов.

На рисунке 10 представлена точность определения поддельной медицинской информации с помощью трех различных классификаторов.

В таблице 3 представлены выбранные параметры методов генерирования поддельных данных, при использовании которых были получены наилучшие результаты.

Заключение

В статье исследовано генерирование синтетических данных для honeypot-систем с использованием методов глубокого обучения.

Выявление актуальных целевых объектов сетевых атак позволило выделить три типа данных, генерация которых осуществляется в рамках статьи — персональные данные, медицинская информация и сетевой трафик АСУ ТП.

Рис. 9. Точность проведения атаки установления атрибутов для сгенерированной медицинской информации

Fig. 9. Accuracy of an Attribute Attack on Generated Health Information

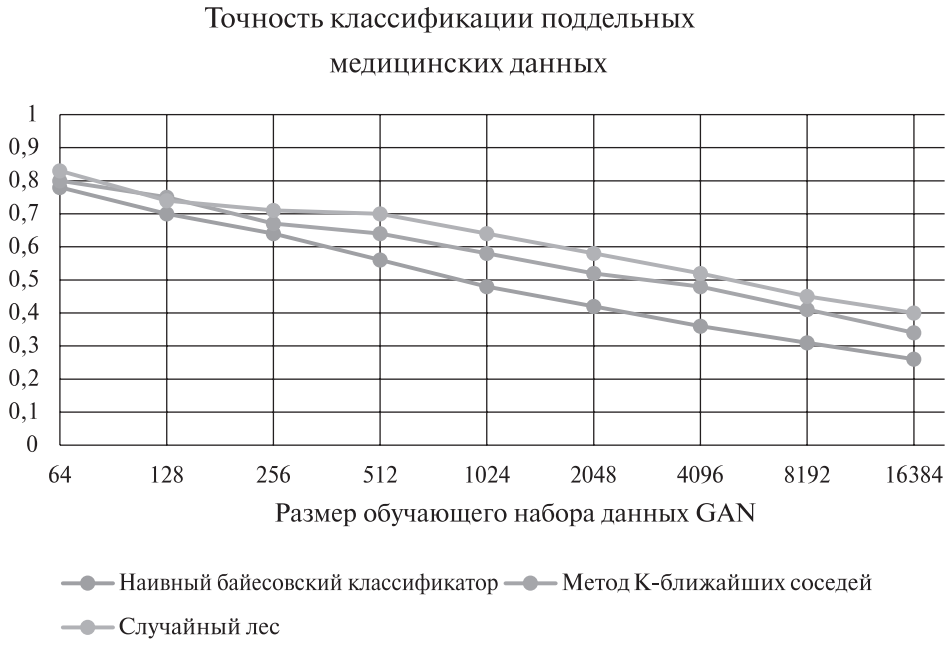


Рис. 10. Оценка сходства реальной и поддельной медицинской информации

Fig. 10. Assessing the similarity between real and fake medical information

Исходя из особенностей существующих методов генерирования синтетических данных для каждого из типов данных, генерирование которых осуществляется в рамках статьи, были выбраны соответствующие методы.

Основными показателями качества синтетических данных являются невозможность восстановления записей из обучающего набора данных, а также сходство сгенерированных данных с реальными. Были проанализированы атаки установ-

Таблица 3. Параметры методов генерирования данных и методов оценки качества поддельных данных, общее время работы

Table 3. Parameters of data generation methods and fake data quality assessment methods, total operating time

Параметры \ Тип данных	Персональные данные	Медицинская информация	Сетевой трафик АСУ ТП
Метод генерации	GAN, построенный на CNN	GAN, построенный на RNN	GAN, построенный на RNN
Количество эпох GAN	10	5	5
Количество шагов оптимизации в эпохе GAN	10000	10000	10000
Количество слоев в дискриминаторе GAN	10	3	3
Количество соседей (алгоритм K-ближайших соседей)	–	14	14
Количество деревьев (алгоритм случайной лес)	–	100	100
Общее время работы	500 часов	20 часов	20 часов

ления членства и атрибутов. Атака установления членства заключается в проверке теории об участии конкретных записей при обучении генеративно-состязательной сети. Атака установления атрибутов направлена на получение значений неизвестных атрибутов конкретной записи путем обучения классификатора на синтетических данных. Проведение оценки сходства реальных и поддельных данных возможно с использованием методов: сравнение статистических показателей, классификация с помощью алгоритмов машинного обучения и соответствие спецификации протокола.

В результате был разработан макет автоматизированной системы генерирования синтетических данных для заполнения honeypot-систем с использованием методов глубокого обучения. Также была осуществлена оценка эффективности работы разработанного макета.

СПИСОК ИСТОЧНИКОВ

1. Positive Technologies — Актуальные киберугрозы: 1 квартал 2021 [Электронный ресурс].— URL: https://www.ptsecurity.com/ru-ru/research/analytics/cybersecurity-threatscape-2021-q1/.— (дата обращения 22.02.2022)

2. **Mairh A.** et al. Honeypot in network security: a survey //Proceedings of the 2011 international conference on communication, computing & security.— 2011.— С. 600–605

3. **Овасапян Т. Д., Никулкин В. А., Москвин Д. А.** Применение технологии honeypot с адаптивным поведением для сетей интернета вещей // Проблемы информационной безопасности. Компьютерные системы.— 2021.— № 2.— С. 135–144.

4. **Bao J., Ji C., Gao M.** Research on network security of defense based on Honeypot //2010 International Conference on Computer Application and System Modeling (ICCASM 2010).— IEEE, 2010.— Vol. 10.— P. V10–299-V10–302

5. Positive Technologies: Атаки на промышленные компании [Электронный ресурс].— URL: https://www.ptsecurity.com/ru-ru/about/news/positive-technologies-chislo-atak-na-promyshlennye-kompanii-vyroslo-na-91-po-sravneniyu-s-2019-godom/.— (дата обращения 10.11.2021)

6. **Krundyshev V., Kalinin M.** The security risk analysis methodology for smart network environments // 2020 International Russian Automation Conference (RusAutoCon).— IEEE, 2020.— P. 437–442.

7. **Ivanov D.** et al. Automatic security management of smart infrastructures using attack graph and risk analysis // 2020 Fourth World Conference

on Smart Trends in Systems, Security and Sustainability (WorldS4).— IEEE, 2020.— P. 295–300.

8. **Зубков Е. А., Жуковский Е. В.** Оценка защищенности гетерогенных компьютерных сетей на основе анализа сетевой топологии // Методы и технические средства обеспечения безопасности информации.— 2021.— № 30.— С. 88–89.

9. **Овасапян Т. Д., Данилов В. Д., Москвин Д. А.** Применение методов генерации синтетических данных в задачах выявления сетевых атак на устройства интернета вещей // Проблемы информационной безопасности. Компьютерные системы.— 2020.— № 4.— С. 26–34.

10. **Belenko V., Krundyshev V., Kalinin M.** Synthetic datasets generation for intrusion detection in VANET // Proceedings of the 11th international conference on security of information and networks.— 2018.— P. 1–6.

11. **Жуковский Е. В., Зегжда Д. П.** Повышение устойчивости к состязательным атакам моделей машинного обучения, используемых в современных средствах защиты // Методы и технические средства обеспечения безопасности информации.— 2020.— № 29.— С. 73–74.

12. **Stadler T., Oprisanu B., Troncoso C.** Synthetic Data–Anonymisation Groundhog Day // arXiv preprint arXiv:2011.07018.— 2020.— P. 1–18.

13. **Shokri R.** et al. Membership inference attacks against machine learning models // 2017 IEEE Symposium on Security and Privacy (SP).— IEEE, 2017.— P. 3–18

14. **Hayes J.** et al. Logan: Membership inference attacks against generative models //Proceedings on Privacy Enhancing Technologies (PoPETs).— De Gruyter, 2019.— Vol. 2019.— No 1.— P. 133–152 — Bellovin S. M., Dutta P. K., Reitinger N. Privacy and synthetic datasets // Stan. Tech. L. Rev.— 2019.— Vol. 22.— P. 1

15. **Peterson L. E.** K-nearest neighbor //Scholarpedia.— 2009.— Vol. 4.— No 2.— P. 1883 — Oshiro T. M., Perez P. S., Baranauskas J. A. How many trees in a random forest? // International workshop on machine learning and data mining in pattern recognition.— Springer, Berlin, Heidelberg, 2012.— P. 154–168

16. Large-scale CelebFaces Attributes (CelebA) Dataset [Электронный ресурс].— URL: https://mmlab.ie.cuhk.edu.hk/projects/CelebA.html/.— (дата обращения 22.02.2022)

17. Hospital Discharge Data Public Use Data File [Электронный ресурс].— URL: https://www.dshs.texas.gov/THCIC/Hospitals/Download.shtm/.— (дата обращения 22.02.2022).

REFERENCES

1. Positive Technologies — Актуальные киберугрозы: 1 квартал 2021 [Электронный ресурс].— URL: https://www.ptsecurity.com/ru-ru/research/analytics/cybersecurity-threatscape-2021-q1/.— (дата обращения 22.02.2022)

2. **Mairh A.** et al. Honeypot in network security: a survey //Proceedings of the 2011 international conference on communication, computing & security.— 2011.— P. 600–605

3. **Ovasapyan T. D., Nikulin V. A., Moskvin D. A.** Application of honeypot technology with adaptive behavior for internet of things networks // Problems of information security. Computer systems.— 2021.— No. 2.— P. 135–144. (In Russian)

4. **Bao J., Ji C., Gao M.** Research on network security of defense based on Honeypot //2010 International Conference on Computer Application and System Modeling (ICCASM 2010).— IEEE, 2010.— Vol. 10.— P. V10–299-V10–302

5. Positive Technologies: Ataki na promyshlennye kompanii [Attacks on industrial companies].— URL: https://www.ptsecurity.com/ru-ru/about/news/positive-technologies-chislo-atak-na-promyshlennye-kompanii-vyroslo-na-91-po-sravneniyu-s-2019-godom/.— (дата обращения 10.11.2021) (In Russian)

6. **Krundyshev V., Kalinin M.** The security risk analysis methodology for smart network environments // 2020 International Russian Automation Conference (RusAutoCon).— IEEE, 2020.— P. 437–442.

7. **Ivanov D.** et al. Automatic security management of smart infrastructures using attack graph and risk analysis // 2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4).— IEEE, 2020.— P. 295–300.

8. **Zubkov E. A., Zhukovskij E. V.** Ocenka zashchishchennosti geterogennyh komp'yuternyh setej na osnove analiza setevoj topologii [security assessment of heterogeneous computer networks based on network topology analysis] // Metody i tekhnicheskie sredstva obespecheniya bezopasnosti informacii.— 2021.— No 30.— P. 88–89. (In Russian)

9. **Ovasapyan T. D., Danilov V. D., Moskvin D. A.** Primenenie metodov generacii sinteticheskih dannyh v zadachah vyyavleniya setevyh

atak na ustrojstva interneta veshchej [Application of synthetic data generation methods in the tasks of detecting network attacks on Internet of Things devices]//Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy.— 2020.— No 4.— P. 26–34. (In Russian)

10. **Belenko V., Krundyshev V., Kalinin M.** Synthetic datasets generation for intrusion detection in VANET // Proceedings of the 11th international conference on security of information and networks.— 2018.— P. 1–6.

11. **Zhukovskij E. V., Zegzhda D. P.** Povyshenie ustojchivosti k sostyazatel'nym atakam modelej mashinnogo obucheniya, ispol'zuemyh v sovremennyh sredstvah zashchity [Increasing the resistance to adversarial attacks of machine learning models used in modern means of protection] // Metody i tekhnicheskie sredstva obespecheniya bezopasnosti informacii.— 2020.— No 29.— P. 73–74. (In Russian)

12. **Stadler T., Oprisanu B., Troncoso C.** Synthetic Data–Anonymisation Groundhog Day // arXiv preprint arXiv:2011.07018.— 2020.— P. 1–18.

13. **Shokri R.** et al. Membership inference attacks against machine learning models //2017 IEEE Symposium on Security and Privacy (SP).— IEEE, 2017.— P. 3–18

14. **Hayes J.** et al. Logan: Membership inference attacks against generative models //Proceedings on Privacy Enhancing Technologies (PoPETs).— De Gruyter, 2019.— Vol. 2019.— No 1.— P. 133–152 — Bellovin S. M., Dutta P. K., Reitinger N. Privacy and synthetic datasets //Stan. Tech. L. Rev.— 2019.— Vol. 22.— P. 1

15. **Peterson L. E.** K-nearest neighbor //Scholarpedia.— 2009.— Vol. 4.— No 2.— P. 1883 — Oshiro T. M., Perez P. S., Baranauskas J. A. How many trees in a random forest? // International workshop on machine learning and data mining in pattern recognition.— Springer, Berlin, Heidelberg, 2012.— P. 154–168.

16. Large-scale CelebFaces Attributes (CelebA) Dataset [Электронный ресурс].— URL: https://mmlab.ie.cuhk.edu.hk/projects/CelebA.html/.— (дата обращения 22.02.2022)

17. Hospital Discharge Data Public Use Data File [Электронный ресурс].— URL: https://www.dshs.texas.gov/THCIC/Hospitals/Download.shtm/.— (дата обращения 22.02.2022).

Статья поступила в редакцию 09.02.2022; одобрена после рецензирования 16.02.2022; принята к публикации 22.02.2022.

The article was submitted 09.02.2022; approved after reviewing 16.02.2022; accepted for publication 22.02.2022.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

ДАНИЛОВ Владислав Дмитриевич — студент; Санкт-Петербургский Политехнический университет Петра Великого

DANILOV Vladislav D. — student; Peter the Great St. Petersburg Polytechnic University
Email: danilov.wrk@gmail.com
ORCID: 0000-0002-6370-123X

ОВАСАПЯН Тигран Джаникович — ассистент; Санкт-Петербургский Политехнический университет Петра Великого

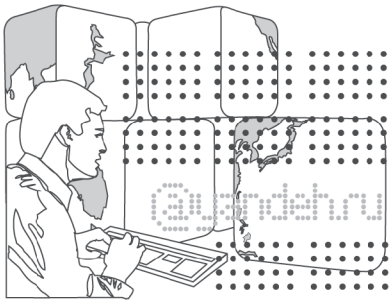
OVASAPYAN Tigran D. — Assistant Professor; Peter the Great St. Petersburg Polytechnic University
Email: otd@ibks.spbstu.ru
ORCID: 0000-0002-2009-5460

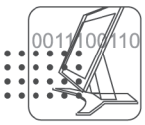
ИВАНОВ Денис Вадимович — к.т.н., доцент; Санкт-Петербургский Политехнический университет Петра Великого

IVANOV Denis V. — Ph.D., Associate Professor; St. Petersburg Polytechnic University of Peter the Great
ORCID: 0000-0001-8206-2915

КОНОПЛЕВ Артем Станиславович — к.т.н., доцент; Санкт-Петербургский Политехнический университет Петра Великого

KONOPLEV Artem S. — Ph.D., Associate Professor; St. Petersburg Polytechnic University of Peter the Great
Email: konoplev@ibks.spbstu.ru
ORCID: 0000-0002-9074-4366





МОДЕЛИРОВАНИЕ ТЕХНОЛОГИЧЕСКИХ СИСТЕМ

Научная статья
DOI 10.48612/jisp/xvx1-7ppr-2prx
УДК 519.718

А. М. Сухов, А. В. Крупенин, В. И. Якунин
Краснодарское высшее военное училище имени С. М. Штеменко
Россия, 350063, Краснодар, ул. Красина, д. 4

МЕТОДЫ ПОСТРОЕНИЯ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ ПОКАЗАТЕЛЕЙ КАЧЕСТВА РЕЗУЛЬТАТОВ ПРОЦЕССА ФУНКЦИОНИРОВАНИЯ СИСТЕМЫ ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Аннотация. Рассматривается новый подход, связанный с построением математических моделей показателей качества для дальнейшего оценивания процесса функционирования системы обеспечения информационной безопасности с учетом предъявления требований к результатам проводимой операции. Обоснован вектор показателей качества результатов процесса функционирования рассматриваемой системы. Представлены модели показателя виртуального и требуемого качества результатов процесса функционирования системы обеспечения информационной безопасности.

Ключевые слова: математическая модель, показатель качества, сценарий деструктивного воздействия, единое информационное пространство, система обеспечения информационной безопасности.

Research article
DOI 10.48612/jisp/xvx1-7ppr-2prx

A. M. Sukhov, A. V. Krupenin, V. I. Yakunin
Krasnodar Higher Military School named after S. M. Shtemenko, Russia, Krasnodar

METHODS OF CONSTRUCTING MATHEMATICAL MODELS OF QUALITY INDICATORS OF THE RESULTS OF THE PROCESS OF FUNCTIONING OF THE INFORMATION SECURITY SYSTEM

Abstract. A new approach is considered related to the construction of mathematical models of quality indicators for further evaluation of the process of functioning of the information security system, taking into account the requirements for the results of the operation. The vector of quality indicators of the results of the process of functioning of the system. Under consideration is substantiated, models of the virtual indicator and the required quality of the results of the process of functioning of the information security system are presented.

Keywords: mathematical model, quality indicator, destructive impact scenario, unified information space, information security system.

Исследованию эффективности в настоящее время посвящено большое число работ [1–5], однако эффективное функционирование системы обеспечения информационной безопасности (СОИБ) в современных условиях, может быть достигнуто только при широком внедрении информационных технологий в повседневную деятельность с существенным повышением уровня информационной поддержки процессов управления [6, 7]. Лавинообразное нарастание объема и разнообразия информации, циркулирующей в едином информационном пространстве (ЕИП) [8, 9], необходимость обеспечения гибкого и оперативного реагирования на изменение обстановки, возрастание роли временного фактора в упреждении сценариев деструктивных воздействий (СДВ) нарушителя приводят к необходимости поиска новых путей повышения качества современного применения СОИБ в ЕИП [10].

Целью процесса функционирования (ПФ) СОИБ является обеспечение требуемого уровня защищенности ЕИП (рис. 1). Для проведения операции обнаружения СДВ направленных на срыв штатного функционирования ЕИП, СОИБ должна функционировать следующим образом. Узлы системы (датчики СОИБ, модули СЗИ) размещенные внутри ЕИП, являются добывающими элементами СОИБ, выявляют вероятностно-временные характеристики (ВВХ) СДВ. На основе полученных значений ВВХ формируются атомарные события информационной безопасности (АСИБ), реализацию которых необходимо спрогнозировать в определенной последовательности и в заданный интервал времени.

Постановка задачи формулируется следующим образом.

По имеющимся априорным данным о эксплуатационно-технических характеристиках (ЭТХ) СОИБ, о параметрах организации процесса ее функционирования, об условиях функционирования СОИБ, о свойствах СДВ и местах их проникновения в ЕИП.

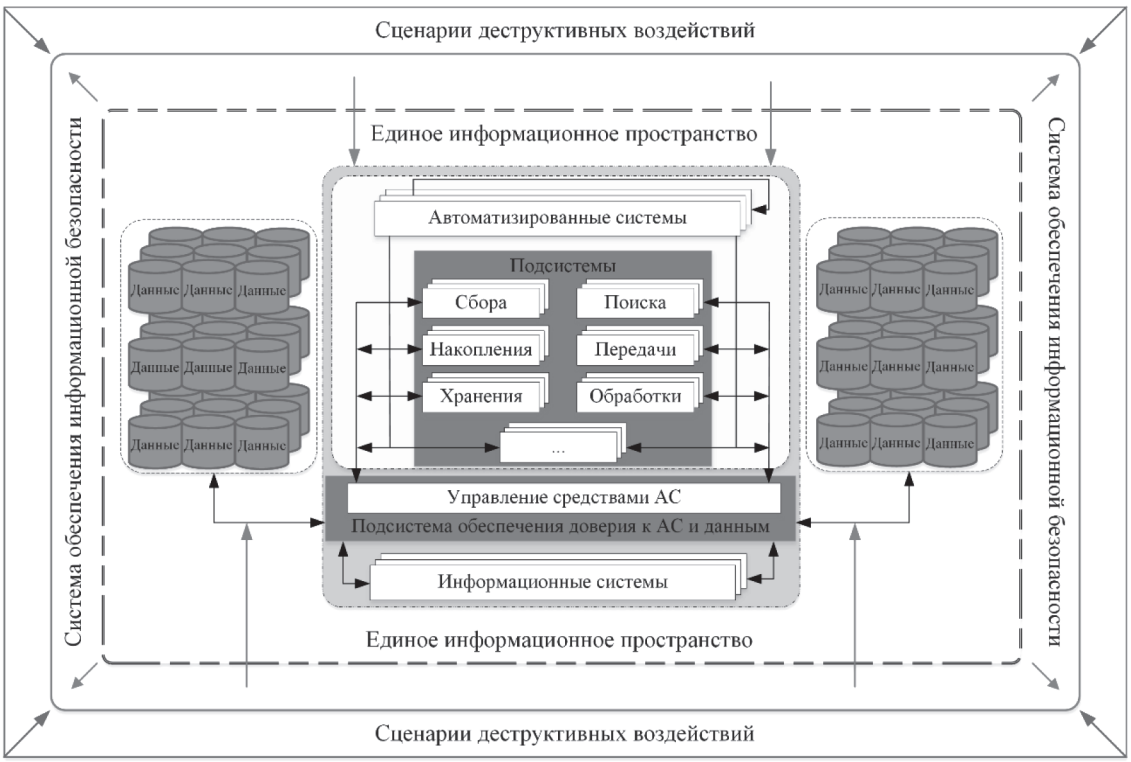


Рис. 1. Схема построения защищенного ЕИП

Fig. 1. Scheme for constructing a secure UIS

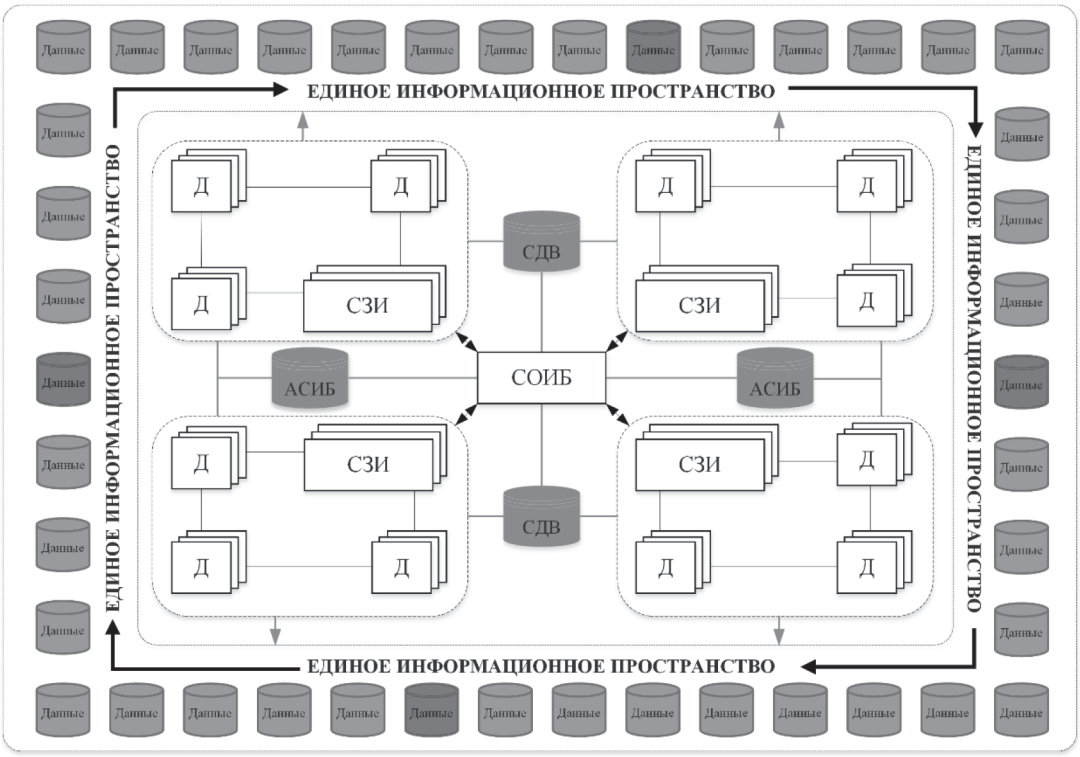


Рис. 2. Схема построения СОИБ ЕИП

Fig. 2. ISS of UIS construction scheme

При ограничениях на объем базы данных СДВ (и как следствие на АСИБ) и их характеристики;

Требуется:

– обосновать показатель качества результатов ПФ СОИБ;

– математическую модель показателя виртуального качества результатов операции;

– математическую модель показателя требуемого качества результатов операции.

Математическая формулировка задачи исследования

На основе содержательной постановки задачи и основных положений теории эффективности целенаправленных процессов [11] представляется возможным формализовать постановку задачи и дать ее математическую формулировку.

По имеющимся априорным данным о ЭТХ $\hat{\mathbf{A}}'_{\langle k' \rangle}$ СОИБ, о параметрах $\hat{\mathbf{A}}''_{\langle k'' \rangle}$ организации процесса ее функционирования, о характеристиках $\hat{\mathbf{B}}'_{\langle l' \rangle}$ условий функционирования СОИБ и априорных данных о характеристиках $\hat{\mathbf{B}}''_{\langle l'' \rangle}$ СДВ (характеристиках условий применения) требуется построить математические модели показателя качества результатов ПФ СОИБ, а именно математическую модель показателя виртуального качества результатов операции и математическую модель показателя требуемого качества результатов операции.

Допустим, что СОИБ M (рис. 2), состоит из совокупности различных средств защиты информации (СЗИ), которые получают различного рода информацию от датчиков (Д) выявления СДВ, проводит операцию по защите ЕИП D от J СДВ, тогда математическая постановка выше сформулированной задачи имеет вид:

Дано:

– характеристики возможных СДВ:

1. $C^{j_i} = \left\{ c_1^{j_i}, c_2^{j_i}, \ldots, c_k^{j_i} \right\}$ — перечень демаскирующих признаков j_i-го АСИБ;

2. $T_J = \left\{\left\langle t'_1, t''_1\right\rangle, \left\langle t'_2, t''_2\right\rangle, ..., \left\langle t'_g, t''_g\right\rangle\right\}$ — множество нормативных $\left(t'_i\right)$ и директивных $\left(t''_i\right)$ значений затрат времени на упреждение АСИБ СОИБ;

– параметры СОИБ:

3. $D_{\text{ПД}} = \{d_1, d_2, ..., d_h\}$ — множество возможных средств обнаружения, анализа и реагирования на СДВ;

4. $M_{\text{ПД}} = \{m_1, m_2, ..., m_l\}$ — множество возможных методов противодействия реализации СДВ;

– параметры нарушителя:

5. $\mathbf{U}_{\langle r\rangle} = \left\langle u_1, u_2, ..., u_r\right\rangle$ — вектор параметров системы нарушителя и характеристик её ПФ.

Ограничения:

1. $\mathbf{U}_{\langle r\rangle} \in \left\{\mathbf{U}^{\partial}_{\langle r\rangle}\right\}$ — вектор параметров системы принадлежит ОДЗ;

2. $\hat{\tau}'_{\text{РСДВ}} \le \tau_{\partial}$ — затраты времени $\hat{\tau}'_{\text{РСДВ}}$ на реализацию ДВ не превышают максимально допустимых (директивных) τ_{∂}.

Допущения:

1. $J = \{j_1, j_2, ..., j_g\}$ — множество СДВ генерируются в ходе исследования.

Требуется:

1. Обосновать вектор показателей качества результатов ПФ СОИБ.

2. Построить модель показателя виртуального качества результатов ПФ СОИБ.

3. Построить модель показателя требуемого качества результатов ПФ СОИБ.

Обоснование вектора показателей качества результатов ПФ СОИБ

Целью операции обнаружения признаков реализации СДВ СОИБ является своевременное выявление АСИБ, входящих в реализуемый нарушителем СДВ, с качеством (полнотой) не хуже требуемого для их дальнейшего анализа и определение ВВХ АСИБ составляющих СДВ. Предположим, что в ЕИП реализуется J_g СДВ, состоящий из C_k АСИБ на интервале $\hat{\tau}'_{\text{РСДВ}}$ времени реализации (рис. 3).

Исходя из цели рассматриваемого ПФ СОИБ, показатель качества результатов операции с учетом эффекта поглощения [10, 11] по операционным ресурсам должен включать две компоненты характеризующих соответственно целевой эффект: относительный объем (полноту) обнаруженных признаков СДВ и временные затраты на получение целевого эффекта.

В качестве характеристики целевого эффекта $\hat{y}_1$ (где ^ — символ случайного объекта) процесса обнаружения СОИБ признаков СДВ в ЕИП, целесообразно принять долю $\hat{\upsilon}$ обнаруженных с качеством не хуже требуемого АСИБ от общего числа переданных нарушителем (входящих в состав СДВ).

Операционное время $\hat{\tau}$ определяется интервалом времени между моментом $\hat{t}_H$ обнаружения C_1 АСИБ в ЕИП, обусловленного моментом начала внедрения СДВ, и моментом $\hat{t}_{\text{УПР}}$ упреждения реализуемого СДВ СОИБ в целом.

В рамках построения модели показателя качества результатов операции введем следующие допущения:

1. Все сообщения (пакеты) передаваемые по каналу связи имеют одинаковую длину l, равную модальному (наиболее вероятному) значению.

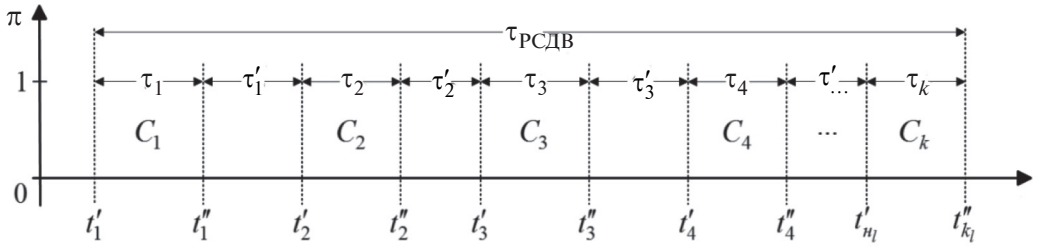


Рис. 3. Процесс реализации J_g СДВ

Fig. 3. The implementation process of the J_g destructive impact scenario

2. Скорость обработки пакетов равна скорости их передачи по используемой сети передачи данных.

3. АСИБ принадлежит одному из возможных СДВ.

Ввиду вышеизложенного, операционное время $\hat{\tau}$ можно представить в виде:

$$\hat{\tau} = \hat{\tau}_{\text{УПР}} + \hat{\tau}_{\xi}$$

где $\hat{\tau}_{\text{УПР}}$ — интервал времени упреждения СДВ, $\hat{\tau}_{\text{УПР}} \in \left(\hat{t}_1, \hat{t}_{\text{УПР}}\right)$, $\hat{\tau}_{\xi}$ — задержки возникающие в процессе реализации сценария, $\hat{\tau}_{\xi} \in \left[t_0, \hat{t}_{\text{УПР}}\right)$.

Так как в ходе проведения операции обнаружения АСИБ расходуются все имеющиеся в наличии СОИБ ресурсы (технические, технологические, людские), то имеет место эффект поглощения (т.е. операционные затраты ресурсов $\hat{r}$ равны предельным $r_{\text{П}}$). Следовательно, показатель $\hat{r}$ расхода ресурсов явно не присутствует в показателе качества результатов процесса обнаружения СДВ — векторе $\langle \hat{\upsilon}, \hat{\tau} \rangle$, но фигурирует опосредствованно через его ведущие компоненты [11], т.е. $\langle\, \upsilon(\hat{r}), \tau(\hat{r}) \,\rangle = \langle\, \hat{\upsilon}, \hat{\tau} \,\rangle$.

Таким образом, качество результатов процесса обнаружения СДВ характеризуется случайным вектором $\langle \hat{\upsilon}, \hat{\tau} \rangle$.

Модель показателя виртуального качества результатов ПФ СОИБ

Поскольку процесс ведения мониторинга ЕИП СОИБ непрерывный и круглосуточный, то предлагается применить дифференциальный подход к моделированию и качество результатов рассматривать относительно одного АСИБ, являющегося полномочным представителем всей их совокупности. Тогда случайную величину $\hat{\upsilon}$ следует трактовать как долю АСИБ обнаруженных с требуемым качеством, а операционное время $\hat{\tau}$ — временем прохождения АСИБ всех технологических этапов процесса упреждения возможного СДВ (приема, обработки, анализа, оценивания и устранения различных видов задержек и т.п.).

Для определения закона распределения $\Phi_{\langle \hat{\upsilon}, \hat{\tau} \rangle}(\upsilon, \tau)$ показателя $\langle \hat{\upsilon}, \hat{\tau} \rangle$ виртуального качества результатов операции необходимо знать операционный функционал $\upsilon = S(\tau)$ (соотношение, связывающее между собой компоненты вектора $\hat{Y}_{\langle 2 \rangle}$) и закон распределения одной из компонент вектора $\hat{Y}_{\langle 2 \rangle} = \langle \hat{\upsilon}, \hat{\tau} \rangle$.

Для рассматриваемой модели процесса обнаружения признаков СДВ вид операционного функционала $\upsilon = S(\tau)$ обусловлен «эффектом насыщения».

Поскольку датчики СОИБ размещаются в местах наиболее благоприятного приема, существует некоторая минимальная доля υ_T АСИБ заведомо выявленных с качеством не хуже требуемого. Ввиду использования нарушителем различных методов скрытого внедрения СДВ, доля $(1 - \upsilon_T)$ АСИБ принимается с маскирующими признаками, причем доля $(\upsilon_{\max} - \upsilon_{\min})$ может быть доведена до требуемого уровня качества на этапе обработки АСИБ СОИБ. Доля же $(1 - \upsilon_{\max})$ АСИБ маскирована настолько, что не может быть приведена к требуемому качеству в приемлемые сроки и поэтому необходим более тщательный анализ, требующий больших временных затрат.

Таким образом, операционный функционал $\upsilon = S(\tau)$ связывающий целевой эффект υ с временем $\tau_{\text{ВП}}$ выявления признака СДВ и операционным временем τ будет иметь вид описываемый выражениями:

$$\left.\begin{array}{l} \upsilon = \\ = \left(\left(\upsilon_{\max} - \upsilon_{\min}\right)\left(1 - e^{\frac{1}{\lambda}\left(\tau_{\text{ВП}} - \tau'_{\text{ВП}}\right)}\right) + \upsilon_{\text{m}} \right) \times \\ \times \Delta\left(\tau_{\text{ВП}} - \tau'_{\text{ВП}}\right) \\ \tau = \tau_{\text{ВП}} + \tau_{\xi} \end{array}\right\}, (1)$$

где λ — интенсивность обработки АСИБ СОИБ, $\tau'_{\text{ВП}}$ — минимально необходимое технологическое время ликвидации последствий СДВ, $\upsilon_{\max}$ — максимально возможная доля АСИБ выявленных с качеством не хуже требуемого, $\upsilon_{\min}$ — минимально возможная доля АСИБ выявленных с качеством не хуже требуемого, $\Delta\left(\tau - \tau'_{\text{ВП}}\right)$ — селектор луча $\left[\tau'_{\text{ВП}}, \infty\right)$.

Исходя из сказанного и учитывая, что результаты обнаружения каждого СДВ сле-

дует представлять к некоторому сроку τ^{∂}, т.е. время $\hat{\tau}_{\text{ВП}}$ обнаружения СДВ является ограниченной на интервале $\left[\tau'_{\text{ВП}}, \tau''_{\text{ВП}}\right]$ случайной величиной, то закон распределения случайной величины $\hat{\tau}'_{\text{ВП}}$ будет иметь вид:

$$F_{\hat{\tau}_{\text{ВП}}}(\tau) = \left[P_{\min} + (1 - P_{\min}) \frac{\tau - \tau'_{\text{ВП}}}{\tau''_{\text{ВП}} - \tau'_{\text{ВП}}} \right] \times \times \Pi\left(\tau; \tau'_{\text{ВП}}, \tau''_{\text{ВП}}\right) + \Delta\left(\tau - \tau''_{\text{ВП}}\right). \quad (2)$$

Из формулы (2) видно, что случайная величина $\hat{\tau}_{\text{ВП}}$ является случайной величиной смешанного типа, причем $P_{\min}$ есть вероятность того, что на обнаружение СДВ тратится минимально необходимое технологическое время $\tau'_{\text{ВП}}$, т.е. $P_{\min} = P\left(\hat{\tau} = \tau'_{\text{ВП}}\right)$.

Очевидно, что наихудший для процесса обнаружения СДВ случай, дающий гарантированный по надежности результат исследования эффективности ее функционирования соответствует $P_{\min} = 0$, и поскольку для технологического времени $\hat{\tau}_{\text{ВП}}$ обнаружения СДВ на этапе внешнего проектирования можно указать только диапазон принимаемых им значений, то в силу принципа неопределенности Лапласа целесообразно закон распределения случайной величины $\hat{\tau}_{\text{ВП}}$ принять равномерным на интервале $\left[\tau'_{\text{ВП}}, \tau''_{\text{ВП}}\right]$. Таким образом, плотность распределения технологического времени $\hat{\tau}_{\text{ВП}}$ определяется выражением:

$$\varphi_{\hat{\tau}_{\text{ВП}}}(\tau) = \varphi_{\hat{\tau}_{\text{ВП}}}\left(\tau; \tau'_{\text{ВП}}, \tau''_{\text{ВП}}\right) = = \frac{\Pi\left(\tau; \tau'_{\text{ВП}}, \tau_{\text{ВП}}\right)}{\tau''_{\text{ВП}} - \tau'_{\text{ВП}}}, \quad (3)$$

где $\tau''_{\text{ВП}}$ — максимально допустимое значение времени $\hat{\tau}_{\text{ВП}}$ обнаружения СДВ.

При хорошей организации ПФ СОИБ целесообразно предположить, что, чем временная $\hat{\tau}_{\xi}$ задержка больше, тем вероятность ее возникновения меньше, поэтому закон распределения случайной величины $\hat{\tau}_{\xi}$ следует принять смещенным экспоненциальным с параметрами μ и τ'_{ξ}:

$$\varphi_{\hat{\tau}_{\xi}}(\tau) = \mu e^{-\mu\left(\tau - \tau'_{\xi}\right)} \Delta\left(\tau - \tau'_{\xi}\right), \quad (4)$$

где τ'_{ξ} — смещение, характеристика позволяющая учитывать время $\tau_{\text{ПРМ}}$ приема АСИБ, μ — интенсивность возникновения временных задержек по различным причинам.

Найдем плотность распределения $\varphi_{\langle\hat{\upsilon},\hat{\tau}\rangle}(\upsilon, \tau)$ случайного вектора $\langle\hat{\upsilon}, \hat{\tau}\rangle$ — показателя виртуального качества процесса обнаружения СДВ — по формуле:

$$\varphi_{\langle\hat{\upsilon},\hat{\tau}\rangle}(\upsilon, \tau) = \varphi_{\hat{\upsilon}}(\upsilon) \varphi_{\hat{\tau}/\hat{\upsilon}}(\tau; \upsilon) \quad (5)$$

С учетом формулы (3) имеем:

$$\varphi_{\hat{\upsilon}}(\upsilon) = \varphi_{\hat{\tau}_{\text{ВП}}}\left[\tau_{\text{ВП}}(\upsilon)\right]\left[\tau'_{\text{ВП}}(\upsilon)\right] = = \frac{\Pi\left(\upsilon; \upsilon', \upsilon''\right)}{\lambda\left(\tau''_{\text{ВП}} - \tau'_{\text{ВП}}\right)\left(\upsilon_{\max} - \upsilon\right)} = = \frac{\Pi\left(\upsilon; \upsilon_{\min}, \left(\upsilon_{\max} - \upsilon_{\min}\right)\right) \times}{\lambda\left(\tau''_{\text{ВПТ}} - \tau''_{\text{ВПТ}}\right)\left(\upsilon_{\max} - \upsilon\right)} \times \times \left(\left(1 - e^{\frac{1}{\lambda}\left(\tau''_{\text{ВП}} - \tau'_{\text{ВП}}\right)}\right) + \upsilon_{\min} \right) \times \frac{}{\lambda\left(\tau''_{\text{ВПТ}} - \tau''_{\text{ВПТ}}\right)\left(\upsilon_{\max} - \upsilon\right)}. \quad (6)$$

Зафиксируем υ и выразим $\tau_{\text{ВП}}$ через него. Из формулы (1) будем иметь:

$$\tau_{\text{ВП}} = \tau'_{\text{ВП}} - \frac{1}{\lambda} \ln\left(\frac{\upsilon_{\max} - \upsilon}{\upsilon_{\max} - \upsilon_{\min}}\right) \quad (7)$$

и, следовательно,

$$\tau_{\text{ВП}} = \tau'_{\text{ВП}} - \frac{1}{\lambda} \ln\left(\frac{\upsilon_{\max} - \upsilon}{\upsilon_{\max} - \upsilon_{\min}}\right) + \tau_{\xi}. \quad (8)$$

Поскольку случайные величины $\hat{\tau}_{\text{ВП}}$ и $\hat{\tau}_{\xi}$ в рамках модели взаимно независимы и величина $\hat{\upsilon}$ не зависит от временной задержки $\hat{\tau}_{\xi}$, то

$$\varphi_{\hat{\tau}/\hat{\upsilon}}(\tau; \upsilon) = \varphi_{\hat{\tau}_{\text{ВП}}/\hat{\upsilon}}(\tau; \upsilon) * \varphi_{\hat{\tau}_{\xi}}(\tau) = = \mu e^{\frac{1}{\mu}\left[\tau - \tau'_{\xi} - \tau'_{\text{ВП}} + \frac{1}{\lambda}\ln\left(\frac{\upsilon_{\max} - \upsilon}{\upsilon_{\max} - \upsilon_{\min}}\right)\right]} \times \times \Delta\left[\tau - \tau'_{\xi} - \tau'_{\xi} + \frac{1}{\lambda}\ln\left(\frac{\upsilon_{\max} - \upsilon}{\upsilon_{\max} - \upsilon_{\min}}\right)\right]. \quad (9)$$

где * — символ композиции законов распределения.

Таким образом, плотность распределения $\varphi_{\langle\hat{\upsilon},\hat{\tau}\rangle}(\upsilon, \tau)$ случайного вектора $\langle\hat{\upsilon}, \hat{\tau}\rangle$ имеет вид:

$$\varphi_{\langle\hat{\upsilon},\hat{\tau}\rangle}(\upsilon,\tau)=\frac{\Pi\left(\upsilon;\upsilon_T,(\upsilon_{\max}-\upsilon_{\min})\left(1-e^{-\lambda(\tau''_{\text{ВП}}-\tau'_{\text{ВП}})}\right)+\upsilon_{\min}\right)}{\lambda(\tau''_{\text{ВП}}-\tau'_{\text{ВП}})(\upsilon_{\max}-\upsilon)}\times$$
$$\times\mu e^{-\mu\left[\tau-\tau'_{\xi}-\tau'_{\text{ВП}}+\frac{1}{\lambda}\ln\left(\frac{\upsilon_{\max}-\upsilon}{\upsilon_{\max}-\upsilon_{\min}}\right)\right]}\times\Delta\left[\tau-\tau'_{\xi}-\tau'_{\text{ВП}}+\frac{1}{\lambda}\ln\left(\frac{\upsilon_{\max}-\upsilon}{\upsilon_{\max}-\upsilon_{\min}}\right)\right]+ \quad (10)$$
$$+\frac{\Delta(-\upsilon)}{\lambda(\tau''_{\text{ВП}}-\tau'_{\text{ВП}})(\upsilon_{\max}-\upsilon_{\min})}\mu e^{\frac{1}{\mu}\left[\tau-\tau'_{\xi}-\tau'_{\text{ВП}}\right]}\Delta(\tau-\tau'_{\xi}-\tau'_{\text{ВП}}).$$

Функция распределения $\Phi_{\langle\hat{\upsilon},\hat{\tau}\rangle}(\upsilon,\tau)$ случайного вектора $\langle\hat{\upsilon},\hat{\tau}\rangle$ определяется по формуле:

$$\Phi_{\langle\hat{\upsilon},\hat{\tau}\rangle}(\upsilon,\tau)=P\left[(\hat{\upsilon}\ge\upsilon)\bigcap(\hat{\tau}\le\tau)\right]=$$
$$=\int_{\upsilon}^{\infty}\int_{-\infty}^{\tau}\phi_{\langle\hat{\upsilon},\hat{\tau}\rangle}(\upsilon,\tau)d\upsilon d\tau=$$
$$=\int_{\upsilon}^{1}\phi_{\hat{\upsilon}}(\upsilon)\left[\int_{0}^{\tau}\phi_{\hat{\tau}/\hat{\upsilon}}(\tau;\upsilon)d\tau\right]d\upsilon. \quad (11)$$

Выражение (11) представляет собой математическую модель показателя $\langle\hat{\upsilon},\hat{\tau}\rangle$ виртуального качества результатов ПФ СОИБ. Метод построения модели показателя виртуального качества результатов ПФ СОИБ представлен на рис. 4.

Модель показателя требуемого качества результатов ПФ СОИБ

На этапе внешнего проектирования СОИБ и организации процесса ее функционирования с целью большей наглядности и лучшей интерпретируемости результатов исследования обычно предъявляются детерминированные требования к показателю качества результатов целенаправленного процесса функционирования системы (ЦнПФС). Поэтому закон распределения вектора $\hat{\mathbf{Z}}_{\langle 2\rangle}\overset{d}{=}\langle\upsilon_T,\tau_{\partial}\rangle$ — показателя требуемого качества результатов операции обнаружения СДВ — будет вырожденным, т.е.:

$$F_{\hat{\mathbf{Z}}_{\langle 2\rangle}}\left(\mathbf{Z}_{\langle 2\rangle}\right)=F_{\langle\hat{\upsilon}_{\mathcal{A}},\hat{\tau}\ \rangle}(\upsilon_T,\tau_{\partial})=$$
$$=\Delta(\upsilon-\upsilon_T)\Delta(\tau-\tau_{\partial}). \quad (12)$$

где υ_T — требуемое (минимально допустимое) значение доли $\hat{\upsilon}$ принятых АСИБ с требуемым качеством, τ_{∂} — предельно допустимые затраты операционного времени $\hat{\tau}$.

При определении значений величин υ_T и τ_{∂} следует руководствоваться: с одной стороны — априорными соображениями о скорости реализации СДВ и степени защищенности ЕИП, а с другой — соображениями о уровне развития и возможностях современных информационных технологий, реализуемых при проектировании системы и организации ее ПФ. То есть выбор требуемых значений показателей результатов осуществляется путем неформальной процедуры экспертного компромисса.

При обосновании требований υ_T к значению доли $\hat{\upsilon}$ принятых с требуемым качеством АСИБ необходимо учитывать потребности суперсистемы в интересах которой проводится операция обнаружения СДВ. Среди возможных значений компонент вектора $\mathbf{Z}_{\langle 2\rangle}=\langle\upsilon_T,\tau_{\partial}\rangle$ требуемых результатов особый интерес представляют три случая:

1. К результатам ЦнПФС предъявлены наиболее «жесткие» (пессимистические) требования — $\mathbf{Z}^{\Pi}_{\langle 2\rangle}=\langle\upsilon^{\Pi}_T,\tau^{\Pi}_{\partial}\rangle$.

2. К результатам ЦнПФС предъявлены наиболее «мягкие» (оптимистические) требования — $\mathbf{Z}^{O}_{\langle 2\rangle}=\langle\upsilon^{O}_T,\tau^{O}_{\partial}\rangle$.

3. К результатам ЦнПФС предъявлены наиболее вероятные (модальные) требования — $\mathbf{Z}^{M}_{\langle 2\rangle}=\langle\upsilon^{M}_T,\tau^{M}_{\partial}\rangle$.

Предъявление оптимистических требований $\mathbf{Z}^{O}_{\langle 2\rangle}=\langle\upsilon^{O}_T,\tau^{O}_{\partial}\rangle$ к результатам ЦнПФС

Рис. 4. Метод построения модели показателя виртуального качества результатов ПФ СОИБ

Fig. 4. Model constructing method of the virtual quality indicator of the results of the ISS functioning process

позволяет определить потенциальные возможности исследуемой СОИБ, путем рассмотрения ее ПФ в идеальных условиях (например, при отсутствии противоборствующей стороны). Но при проектировании такого рода систем и организации процессов их функционирования рассчитывать на ситуацию, в которой достаточно будет удовлетворить этим требованиям, очевидно не следует.

На этапе внешнего проектирования СОИБ наиболее обоснованным является предъявление пессимистических требований $\mathbf{Z}^{\text{П}}_{\langle 2 \rangle} = \left\langle \upsilon^{\text{П}}_T, \tau^{\text{П}}_{\partial} \right\rangle$, если они возникают достаточно часто, и не выходят за пределы допустимых значений требований по ресурсоемкости и оперативности.

Однако, довольно часто имеет место ситуация, когда виртуальные результаты ПФ СОИБ заведомо не удовлетворяют пессимистическим требованиям, а вероятность возникновения такой операционной ситуации, в которой действительно необходимо удовлетворить таким требованиям достаточно мала. В этом случае целесообразно ориентироваться на модальные (наиболее вероятные) требования $\mathbf{Z}^{M}_{\langle 2 \rangle} = \left\langle \upsilon^{M}_T, \tau^{M}_{\partial} \right\rangle$ к результатам операции.

Выражение (12) представляет собой математическую модель показателя $\langle \upsilon_T, \tau_{\partial} \rangle$ требуемого качества результатов ПФ СОИБ. Метод построения модели показателя требуемого качества результатов ПФ СОИБ представлен на рис. 5.

Рис. 5. Метод построения модели показателя требуемого качества результатов ПФ СОИБ

Fig. 5. Model constructing method of the required quality indicator of the results of the ISS functioning process

Таким образом, сформулированные в постановке задачи неизвестные найдены путем применения вышеперечисленных методов.

Заключение

Для комплексного исследования эффективности ПФ СОИБ вектор показателей качества его результатов должен включать три группы компонент, характеризующих соответственно целевые эффекты, затраты ресурсов и времени. При описании процесса реализации СДВ следует учитывать взаимосвязь всех компонент вектора показателей качества результатов операции. Для построения аналитической модели операционной системы необходимо и достаточно определить операционный функционал и законы распределения генеральных аргументов. Для учета системных свойств ПФ СОИБ, упрощения процедуры построения модели ПФ и интерпретации результатов исследования на этапе внешнего проектирования предпочтительно использование агрегированных параметров модели.

СПИСОК ИСТОЧНИКОВ

1. **Юсупов Р. М**. Особенности оценивания эффективности информационных систем и технологий / Р. М. Юсупов, А. А. Мусаев // Труды СПИИРАН. 2017. Вып. 1(51). С. 5–34.

2. **Гейда А. С**. Основные концепты и принципы исследования операционных свойств использования информационных технологий / А. С. Гейда, И. В. Лысенко, Р. М. Юсупов // Труды СПИИРАН.— 2015.— Вып. 5(42).— С. 5–36.

3. **Гейда А. С.** Задачи исследования операционных и обменных свойств систем / А. С. Гейда, З. Ф. Исмаилова, И. В. Клитный, И. В. Лысенко // Труды СПИИРАН.— 2014.— Вып. 4(35).— С. 136–160.

4. **Schilke O.** Quo vadis, dynamic capabilities? A content-analytic review of the current state of knowledge and recommendations for future research / O. Schilke, S Hu, C. Helfat // Academy of Management Annals. 2018. Vol. 12. No. 1. P. 390–439.

5. **McMahon P.** 15 Fundamentals for Higher Performance in Software Development: Includes discussions on CMMI, Lean Six Sigma, Agile and SEMAT's Essence Framework // Pem Systems Publ. 2014. 336 p.

6. **Сухов А. М**. Подход к моделированию целенаправленных компьютерных атак на основе построения оперативного Т-процесса реализации возможного сценария деструктивного воздействия / А. М. Сухов, И. Е. Горбачев, В. И. Якунин // Материалы 26 научно-технической конференции «Методы и технические средства обеспечения безопасности информации»,— СПб.: Изд-во Политехн. ун-та.— 2017.— С. 44–46.

7. **Сухов А. М**. Математическая модель процесса функционирования подсистемы реагирования системы обнаружения, предупреждения и ликвидации последствий компьютерных атак / А. М. Сухов, С. Ю. Герасимов, М. А. Еремеев, В. И. Якунин // Проблемы информационной безопасности. Компьютерные системы.— 2019.— С. 56–64

8. **Dichenko S.** Two-dimensional control and assurance of data integrity in information systems based on residue number system codes and cryptographic hash functions / S. Dichenko, O. Finko // В сборнике: Integrating Research Agendas and Devising Joint Challenges. International Multidisciplinary Symposium ICT Research in Russian Federation and Europe. 2018. P. 139–146.

9. **Диченко С. А**. Контроль и восстановление целостности многомерных массивов данных посредством криптокодовых конструкций / С. А. Диченко, О. А. Финько // Программирование.— 2021.— № 6.— С. 3–15.

10. **Сухов А. М**. Модель проактивного обнаружения компьютерных атак / А. М. Сухов, Д. Д. Ступин, А. Г. Ломако // В сборнике: Проблемы управления и моделирования в сложных системах. Труды XX Международной конференции. Под редакцией Е. А. Федосова, Н. А. Кузнецова, С. Ю. Боровика.— 2018.— С. 509–512.

11. **Петухов Г. Б., Якунин В. И**. Методологические основы внешнего проектирования целенаправленных процессов и целеустремленных систем.— М.: АСТ, 2006.— 503 с.

REFERENCES

1. **Yusupov R. M**. Osobennosti ocenivaniya effektivnosti informacionnyh sistem i tekhnologij [Features of evaluating the effectiveness of information systems and technologies] / R. M. Yusupov, A. A. Musaev // Trudy SPIIRAN. 2017. Vyp. 1(51). P. 5–34. (In Russian)

2. **Gejda A. S**. Osnovnye koncepty i principy issledovaniya operacionnyh svojstv ispol'zovaniya informacionnyh tekhnologij [Basic concepts and principles of research of operational properties of the use of information technologies] / A. S. Gejda, I. V. Lysenko, R.M.YUsupov // Trudy SPIIRAN. 2015. Vyp. 5(42). P. 5–36. (In Russian)

3. **Gejda A. S**. Zadachi issledovaniya operacionnyh i obmennyh svojstv sistem [Research tasks of operational and exchange properties of systems] / A. S. Gejda, Z. F. Ismailova, I. V. Klitnyj, I. V. Lysenko // Trudy SPIIRAN. 2014. Vyp. 4(35). P. 136–160. (In Russian)

4. **Schilke O.** Quo vadis, dynamic capabilities? A content-analytic review of the current state of knowledge and recommendations for future research / O. Schilke, S Hu, C. Helfat // Academy of Management Annals. 2018. Vol. 12. No. 1. P. 390–439.

5. **McMahon P.** 15 Fundamentals for Higher Performance in Software Development: Includes discussions on CMMI, Lean Six Sigma, Agile and SEMAT's Essence Framework // Pem Systems Publ. 2014. 336 p.

6. **Suhov A. M**. Podhod k modelirovaniyu celenapravlennyh komp'yuternyh atak na osnove

postroeniya operativnogo T-processa realizacii vozmozhnogo scenariya destruktivnogo vozdejstviya [An approach to modeling targeted computer attacks based on the construction of an operational T-process for the implementation of a possible destructive impact scenario] / A. M. Suhov, I. E. Gorbachev, V.I.YAkunin // Materialy 26 nauchno-tekhnicheskoj konferencii «Metody i tekhnicheskie sredstva obespecheniya bezopasnosti informacii»,— S-Pb.: Izd-vo Politekhn. un-ta.— 2017.— P. 44–46. (In Russian)

7. **Suhov A. M**. Matematicheskaya model' processa funkcionirovaniya podsistemy reagirovaniya sistemy obnaruzheniya, preduprezhdeniya i likvidacii posledstvij komp'yuternyh atak [Mathematical model of the functioning process of the response subsystem of the system of detection, prevention and elimination of consequences of computer attacks] / A. M. Suhov, S. Yu. Gerasimov, M. A. Eremeev, V. I. Yakunin // Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy. 2019. P. 56–64 (In Russian)

8. **Dichenko S.** Two-dimensional control and assurance of data integrity in information systems based on residue number system codes and cryptographic hash functions / S. Dichenko, O. Finko // V sbornike: Integrating Research Agendas and Devising Joint Challenges. International Multidisciplinary Symposium ICT Research in Russian Federation and Europe. 2018. P. 139–146.

9. **Dichenko S. A**. Kontrol' i vosstanovlenie celostnosti mnogomernyh massivov dannyh posredstvom kriptokodovyh konstrukcij [Control and restoration of the integrity of multidimensional data arrays by means of cryptographic constructions] / S. A. Dichenko, O. A. Fin'ko // Programmirovanie. 2021. No 6. P. 3–15. (In Russian)

10. **Suhov A. M**. Model' proaktivnogo obnaruzheniya komp'yuternyh atak [A model of proactive detection of computer attacks] / A. M. Suhov, D. D. Stupin, A. G. Lomako // V sbornike: Problemy upravleniya i modelirovaniya v slozhnyh sistemah. Trudy XX Mezhdunarodnoj konferencii. Pod redakciej E. A. Fedosova, N. A. Kuznecova, S.YU. Borovika. 2018. P. 509–512. (In Russian)

11. **Petuhov G. B., Yakunin V. I**. Metodologicheskie osnovy vneshnego proektirovaniya celenapravlennyh processov i celeustremlennyh system [Methodological foundations of external design of purposeful processes and purposeful systems].— M.: AST, 2006.— 503 p. (In Russian)

Статья поступила в редакцию 09.02.2022; одобрена после рецензирования 16.02.2022; принята к публикации 22.02.2022.

The article was submitted 09.02.2022; approved after reviewing 16.02.2022; accepted for publication 22.02.2022.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

СУХОВ Александр Максимович —к.т.н.; докторант, Краснодарское высшее военное училище имени генерала армии С. М. Штеменко.

КРУПЕНИН Александр Владимирович —д.т.н., профессор; заместитель начальника училища по учебной и научной работе, Краснодарское высшее военное училище имени генерала армии С. М. Штеменко.

ЯКУНИН Владимир Иванович — к.т.н., доцент; Военно-космическая академия имени А. Ф. Можайского.

SUKHOV Alexander M.— Ph.D.; doctoral student, Krasnodar Higher Military School named after Army General S. M. Shtemenko.

Email: 19am87@mail.ru

ORCID: 0000-0003-2233-811X

KRUPENIN Alexander V.— Doctor of Technical Sciences, Professor; Deputy head of the School for Academic and Scientific Work, Krasnodar Higher Military School named after Army General S. M. Shtemenko.

Email: 19am87@mail.ru

ORCID: 0000-0003-1404-6125

YAKUNIN Vladimir I.— Ph.D., Associate Professor; Military Space Academy named after A. F. Mozhaisky.

Email: Yavi1957@mil.ru

ORCID: 0000-0002-6071-8087

Научная статья
DOI 10.48612/jisp/5tuk-48fz-zmx5
УДК 65.011.56

А. Ю. Гарькушев[1], А. Ф. Супрун[2], С. Ю. Сысуев[3]

[1]Санкт-Петербургский государственный морской технический университет
[2]Санкт-Петербургский политехнический университет Петра Великого
[3]Михайловская военная артиллерийская академия

МЕТОДИКА ИНТЕГРАЦИИ МОДУЛЕЙ ЗАЩИТЫ ИНФОРМАЦИИ В ОТЕЧЕСТВЕННЫЕ СИСТЕМЫ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ В КОРАБЛЕСТРОЕНИИ

Аннотация. Статья посвящена разработке модели и алгоритма решения задачи обеспечения информационной безопасности в перспективных отечественных системах автоматизированного проектирования на основе модифицированного метода ветвей и границ с применением двойственности решений задач линейного программирования. В результате получен алгоритм, позволяющий разработчику обеспечить информационную безопасность проектов кораблестроения на достаточном уровне.

Ключевые слова: информационная безопасность, проектирование судов, математическая модель.

Research article
DOI 10.48612/jisp/5tuk-48fz-zmx5

A. Yu. Garkushev[1], A. F. Suprun[2], S. Yu. Sysuev[3]

[1]St. Petersburg State Marine Technical University
[2]Peter the Great St. Petersburg Polytechnic University
[3]Mikhailovskaya Military Artillery Academy

PROCEDURE FOR INTEGRATION OF INFORMATION PROTECTION MODULES INTO DOMESTIC AUTOMATED DESIGN SYSTEMS IN SHIPBUILDING

The article is devoted to the development of a model and algorithm for solving the problem of ensuring information security in promising domestic computer-aided design systems based on a modified method of branches and boundaries using the duality of solutions to linear programming problems. As a result, an algorithm was obtained that allows the developer to ensure the information security of shipbuilding projects at a sufficient level.

Keywords: information security, ship design, mathematical model.

Высокая сложность и разнообразие работ при проектировании объектов водного транспорта и организации производственных процессов на судостроительной верфи требует большого набора прикладного программного обеспечения, включая системы автоматизированного проектирования (САПР) для автоматизации выполнения этих работ. Конфигурирование и визуализация проектных решений в виде трехмерной электронной модели судна от разработчика заводу-изготовителю позволяет создать непрерывную IT-цепочку по постройке судна по схеме: технический проект — рабочий проект — производственные подразделения предприятия. Эта же последовательность может быть использована технологической службой предприятия-строителя для проектирования оснастки, для написания технологии сборки и т.п. [1–3]. Кроме того, учитывая подготовку кадрового обеспечения для

работы с САПР, необходимо отметить, что в рамках большинства программ обучения конструкторов для такой малосерийной области проектирования как кораблестроение полноценное освоение хотя бы одной комплексной САПР является нетривиальной задачей. Поэтому, как правило, обучают работе с 3–4 основными модулями такой системы, изучая которые будущие специалисты могли бы получить целостное представление об использовании САПР в постройке и проектировании судов, при этом вопросы обеспечения информационной безопасности остаются за скобками учебных планов.

Рынок САПР на сегодняшний день предлагает разноуровневый выбор решений, позволяющих решать проектные и производственные задачи с той или иной эффективностью.

Активная разработка и внедрение отечественного программного обеспечения в важнейшую промышленную область кораблестроения сегодня показывает первые результаты. Проекты, выполненные с помощью отечественных САПР, незначительно уступают зарубежным по функционалу и существенно уступают по наполнению специальных библиотек цифровых проектов, однако за период 5–6 лет, вероятнее всего, достигнут результатов сравнимых с проектированием на программных продуктах ведущих производителей подобного софта. При этом вопросы информационной безопасности рассматриваются разработчиками во вторую очередь или реализуются традиционными для аналогичных систем способами. Так назревает противоречие между усиливающимся «интересом» конкурентов и спецслужб к замыслам и инженерным решениям отечественных корабелов, и защищённостью информации проектного контента [4–6]. Руководство конструкторской организации или кораблестроительного предприятия в такой ситуации должно понимать, как совместить необходимость применения САПР для интенсификации создания отечественного флота и обеспечением информационной безопасности.

В противоречивых условиях текущего развития судостроительной отрасли особую важность приобретает выбор эффективных средств автоматизации технической подготовки производства, обеспеченных стабильностью компании-разработчика и поддержкой квалифицированных специалистов по внедрению [7–9]. Преимущества, которые дает применение для разработки закрытых проектов именно отечественный софт, не вызывают сомнений. На первый взгляд он отсекает программные закладки, способные в час «Х» вывести из строя САПР или передавать скрытно сведения при наличии подключения к интернету. Однако, до сих пор не разработан эффективный механизм предотвращающий кражу, подмену и уничтожение проектной информации. Не всегда понятен разработчику общепринятый научно-методический аппарат, позволяющий получать обоснованные количественные оценки влияния мер по обеспечению ИБ на необходимость внедрения в архитектуру САПР соответствующих программных модулей, достаточно надежно реагирующих на угрозы и не препятствующих выполнению основных функций [10–12]. Основой своевременного принятия обоснованного решения о интеграции модулей безопасности в САПР должна стать адекватная модель процесса обеспечения ИБ с учетом целей и ограничений, характеризующих функционал проекта. Такая модель при определенной адаптации может рассматривать вопросы интеграции не только защитных модулей, но и других составляющих, обеспечивающих удобный интерфейс и комфортную эксплуатацию модели.

1. Постановка задачи

Создание отечественной САПР реализуется в интересах удовлетворения некоторого спектра характерных потребностей проектировщиков кораблей и судов.

В кораблестроении к ним можно отнести:

– создание поверхностей по заданным граничным условиям или по множеству точек;

– построение поверхностей на основе сегментов форм и оболочек;

– определение площади, центра тяжести и других параметров поверхности;

– контроль кривизны и касательных;

– пересечение, объединение, разрезание, зеркальное отражение, выравнивание и сглаживание поверхностей;

– обнаружение и устранение зазоров (разрывов) и наложений;

– импортирование и обработка поверхностей из других CAD-систем (через стандартные форматы обмена графическими данными IGES, VDAFS и др.).

На основе опыта эксплуатации решений аналогичных PDM (CATIA Data Manager CDM или ENOVIA VPM/PM) и с использованием стандартной реляционной СУБД создается единая многопользовательская база данных, содержащая всю текстовую и графическую информацию проекта и предоставляющая реальные возможности для одновременного согласованного (параллельного) проектирования силами различных рабочих групп с контролем доступа в базу отдельных групп и пользователей.

Для обмена электронными данными с партнерами и поставщиками, работающими в иных CAD/CAM-системах, используются интерфейсы DXF, IGES, STL, STEP (AP203, AP207, AP214) и других форматов.

Поскольку все они, как правило, не могут быть полностью обеспечены каким-либо одним вариантом технологии защиты, рациональная интеграция соответствующих модулей в защиту оболочки и баз данных происходит путем формирования соответствующих параметрических рядов (наборов) программных продуктов [13–17].

Независимо от специфики конкретных видов угроз, информационная безопасность САПР должна сохранять целостность, конфиденциальность, доступность. Угрозы нарушения целостности, конфиденциальности и доступности являются первичными. Нарушение целостности включает в себя любое умышленное изменение информации, хранящейся в компьютерной системе или передаваемой из одной системы в другую. Нарушение конфиденциальности может привести к ситуации, когда информация становится известной тому, кто не располагает полномочия доступа к ней. Угроза недоступности информации возникает всякий раз, когда в результате преднамеренных действий других пользователей или злоумышленников блокируется доступ к некоторому ресурсу компьютерной системы.

К основным угрозам при работе в частности с PLM-продуктами относят утечку конфиденциальной информации и нарушение работоспособности системы. Причем последняя угроза может быть реализована как с помощью атак, рассчитанных на отказ в обслуживании и выводящих из строя отдельные элементы CAD-систем, так и с помощью вирусов и червей, заражающих САПР.

Сложность современных САПР и разнообразие предъявляемых к ним требований приводит к тому, что проблемы формирования набора средств защиты часто усложняют и без того непростые варианты построения систем информационной безопасности (СИБ). Попытки внедрения СИБ на основе предыдущего опыта и интуиции часто приводит к потере информационного контента, что в условиях высокой стоимости и большой длительности разработки САПР недопустимо. Поэтому, решение указанных проблем должно быть не только интуитивным, но и опираться на объективные оценки. Инструментом для таких оценок могут быть соответствующие математические модели, которые предложены виде модели и алгоритма решения задачи оптимизации многомерных параметрических рядов в интересах рациональной интеграции программных модулей в проект отечественной кораблестроительной САПР.

Для парирования угроз ИБ имеется опыт применения различных технологий. Возможно описание множества $I = \{1, 2, ...i, ..., M\}$ технологий, которые совместно достигнут приемлемого уровня ИБ. Каждой i-й технологии также соответствует шлейф параметров, характеризующих эксплуатационные, программные, эргономические и другие свойства.

С формальной точки зрения существует множество $J = \{1, 2, ..j, ..., N\}$ — видов угроз ИБ. Каждый вид угроз определяется совокупностью многих характеристик (показателей, параметров). Например, угроза уничтожения информации, угроза раскрытия информации, угроза подмены контента и так далее.

Необходимо из множества I возможных технологий защиты выбрать такое подмножество $I^* \subset I$, которые полностью купирует негативные последствия от множества J видов угроз с необходимой эффективностью.

В основу модельного представления этой задачи положена гипотеза о многофакторности парирования информационных угроз. В частности, разрушительными свойствами угроз ИБ, выбором критерия оптимизации ряда, учетом дополнительных ограничений, неопределенностью среды возникновения угрозы и т.п. [18–21].

Рациональным с точки зрения принципа максимума неопределенности (то есть «хуже уже не будет») представляется рассмотрение задачи в следующем виде.

Необходимо

$$C = \min\left(\sum_{i=1}^{M} c_i^o y_i + \sum_{i=1}^{M}\sum_{j=1}^{N} c_{ij} x_{ij}\right) \qquad (1)$$

при наличии ряда ограничений

$$\sum_{i=1}^{M} \alpha_{ij} x_{ij} = 1,\ j = 1, 2, ..., N; \qquad (2)$$

$$x_{ij} \le y_i,\ i = 1, 2, ..., M, j = 1, 2, ..., N; \qquad (3)$$

$$x_{ij} \in \{0, 1\},\ y_i \in \{0, 1\},\ i = 1, 2, ..., M, j = 1, 2, ..., N; \qquad (4)$$

$$\sum_{i=1}^{M} y_i \le M_0,\ M_0 \le M, \qquad (5)$$

где c_i^0 — расходы на технологию i-го типа (начальные затраты); $c_{ij} = \alpha_{ij} c_i b_j$ — расходы, связанные с купированием угроз j-того вида технологиями i-го типа; α_{ij} — число модулей i -го типа, необходимых для парирования угрозы ИБ j-го вида; c_{ij} — расход ресурса для технологии защиты i-того типа; b_{ij} — условная значимость угрозы j-того вида; M_0 — число технологий, которые могут быть включены в оптимальный ряд I^*.

Взаимное влияние элементов множеств I и J друг на друга задается с помощью матрицы влияния $||\alpha_{ij}||_{MxN,}$ в которой $\alpha_{ij} = 1$, если технология i-го типа может парировать j-ю угрозу, и $\alpha_{ij} = 0$ — в противном случае. Причем в начальный момент считается, что $c_{ij} = \infty$, если $\alpha_{ij} = 0$.

Управляющие переменные принимают значения: $y_i = 1$, если i-ую технологию защиты включаем в набор ИБ, 0 — в противном случае; $x_{ij} = 1$, если i-тый тип защиты может купировать j-тую угрозу, 0 — в противном случае.

Выражения (1)-(5) позволяют рассматривать процесс парирования угроз ИБ в САПР в виде модели дискретной многомерной задачи стандартизации при определенном спросе [22, 23].

2. Вариант решения комбинирования многопараметрических рядов модифицированным методом ветвей и границ

Комбинирование, направленный перебор и сравнение наборов технических средств технологий, описываемых шлейфом параметров является актуальной задачей для многих отраслей, где принимаются решения в условиях неопределенности, в том числе и в кораблестроении. Распространенным подходом к решению подобных задач комбинирования многопараметрических рядов [24] является алгоритм, представляющий собой процесс направленного перебора с возвращениями. В основу этого процесса положена принципиальная схема метода ветвей и границ. Модификация известных методов [25, 26] заключается в том, что для ветвления используется всего два варианта включения технологии защиты в состав программного обеспечения на каждом шаге решения. Для оценки нижней границы решения в процессе ветвления используется задача линейного программирования двойственная по отношению к задаче (1)-(5), в которой условие целочисленности (4) заменяется условием [27]

$$0 \le y_i \le 1,\ i = 1, 2, ..., M. \qquad (6)$$

Алгоритмизация потребовала введения таких понятий, как:

$I_0 = \{i | y_i = 0\}$ множество индексов управляющих переменных y_i, вошедших в ветвь дерева вариантов со значением, равным нулю (обозначаются как $\overline{y_i}$);

$I_1 = \{i | y_i = 1\}$ — набор индексов управляющих переменных y_i, вошедших в ветвь дерева вариантов со значением, равным единице;

$I_2 = I_1 \cup I_0$ — набор индексов управляющих переменных y_i, вошедших в ветвь дерева вариантов;

$I_3 = I / I_2$ набор индексов управляющих переменных y_i, из которых выбирается индекс переменной для включения в множество I_2;

$I_Д$ ($I_Д = I_1$ при условии $|I_2| = M$, или $I_3 = \varnothing$) — набор индексов управляющих переменных y_i, соответствующих допустимому решению задачи (1)-(5);

i^+ — индекс управляющей переменной y_i, включение которого в множество I_1 предпочтительнее по сравнению с другими индексами из множества I_3;

i — индекс управляющей переменной $\overline{y_i}$, который целесообразно включить в множество I_0.

Проверка допустимого решения задачи (1)-(5) на оптимальность осуществляется решением оценочной двойственной задачи. Двойственную задачу линейного программирования по отношению к модифицированной задаче (1)-(3), (6) можно записать следующим образом:

$$Z = \max \sum_{j \in J} V_j \qquad (7)$$

при ограничениях

$$\sum_{j \in J} W_{ij} \le c_i^0, \; i = 1, 2, ..., M; \qquad (8)$$

$$V_j - W_{ij} \le c_{ij}, j = 1, 2, ..., N, \; i = 1, 2, ..., M; \qquad (9)$$

$$W_{ij} \ge 0, j = 1, 2, ..., N, i = 1, 2, ..., M, \qquad (10)$$

где V_j, W_{ij} — переменные двойственной задачи.

Интерпретацию задачи в виде (7)–(10), введя дополнительные переменные γ_{ij}, δ_i, $\left(i = \overline{1, M}, j = \overline{1, N}\right)$, можно представить в виде основной задачи линейного программирования, начальное базисное решение которой имеет вид

$$Z = \sum_{j=1}^{N} V_j;$$

$$\gamma_{ij} = c_{ij} + W_{ij} - V_j, \quad i = 1, 2, ..., M, \; j = 1, 2, ...N;$$

$$\delta_i = c_i^0 - \sum_{j=1}^{N} W_{ij}; \quad i = 1, 2, ..., M.$$

Известно, что задачи оптимизации многомерных параметрических рядов требуют решения задач линейного программирования большой размерности. Действительно, количество переменных k_1 и число уравнений ограничений k_2, задачи линейного программирования связано с размерностью исходной задачи соотношениями $k_1 = M + 2MN + N$, $k_2 = M(N + 1)$. Для решения задачи линейного программирования такой размерности требуется обычно от $M(N + 1)$ до $3M(N + 1)$ итераций.

Очевидным является способ расчета, основанный на следующих соотношениях

$$W_{ij} = \frac{c_i^0}{J_i}; V_j = \min_{i \in I_j} \left(c_{ij} + W_{ij}\right); \quad Z = \sum_{j=1}^{N} V_j;$$

где $J_i = \left\{j \,|\, \alpha_{ij} = I\right\}$; $I_j = \left\{i \,|\, \alpha_{ij} = I\right\}$.

Постановка задачи в виде (7)–(10) позволяет получить приближенное решение, совпадающее с оптимальным, или достаточно близкое к нему. Компьютерные эксперименты показали, что особенностью задачи (1)–(5) является довольно частое совпадение оптимального целочисленного решения с непрерывным. Поэтому зачастую решают двойственную задачу, а от нее переходят к соответствующему решению непрерывной задачи (1)–(3), (6). При совпадении целочисленного и непрерывного решений полученный результат является оптимальным.

Для реализации направленного способа решения двойственной задачи (7)–(10) упростим ее структуру. Для этого исключим переменные W_{ij}, входящие лишь в ограничения. С учетом этого двойственную задачу (7)–(10) запишем в виде

$$Z = \max \sum_{j=1}^{N} V_j \qquad (11)$$

при ограничениях

$$\sum_{j=1}^{N} \max\,(V_j - c_{ij}; 0) \le c_i^0, \quad i = 1,2,\ldots,M \quad (12)$$

Идея направленного решения двойственной задачи (11)-(12) заключается в том, что при учете ее более простой структуры и возможности использования приближенного решения для оценки нижней границы на различных этапах ветвления, предлагается способ формирования значений переменных V_j, при котором они обеспечивают целевой функции (11) по возможности большее значение. Процесс начинается с допустимого решения

$$V_j = \min_{i \in I_1 \in I_3} c_{ij}$$

задачи (11)-(12) и состоит в многократном последовательном просмотре j-х столбцов (j = 1, 2, ..., N) матрицы $\|c_{ij}\|$ с целью проверки возможности увеличения V_j до ближайшей большей величины c_{ij}. При этом в каждом цикле просмотра проверяется возможность увеличения V_j для тех столбцов, в которых находится наименьшее количество элементов c_{ij} удовлетворяющих условию $c_{ij} \le V_j$. Увеличение значений переменных V_j происходит с учетом (12) и ограничено величинами

$$V_j \le \min\left[\min_{i \in I_1} c_{ij};\quad \min_{i \in I}\left(c_{ij} + c_i^0\right)\right]. \quad (13)$$

Значения, получаемые с помощью данного способа, определяются элементами множеств I_1, I_3 и, когда дальнейшее увеличение переменных V становится невозможным, вычисляются по формуле

$$Z(I_1) = z_x = \sum_{j=1}^{N} V_j + \sum_{i \in I_1} c_i^0. \qquad (14)$$

Эффективность применения предложенного аппарата и точность полученного с его помощью результата в значительной степени определяются точностью используемого (одновременно проверяемого на оптимальность) целочисленного решения. Если для оценки нижней границы целевой функции путем направленного решения двойственной задачи (11)-(12) используются ограничения (13), то при проверке допустимого решения на оптимальность вместо (13) требуется выполнение неравенств

$$c_{rj} \le V_j \le \min\left[c_{lj};\quad \min_{i \in I}\left(c_{ij} + c_i^0\right)\right];$$
$$j = 1,2,\ldots,N, \qquad (15)$$

где $c_{ri} = \min_{i \in I_{\ddot{A}}} c_{ij}$; $\quad c_{lj} = \min_{\substack{i \in I \\ i \ne 1}} c_{ij}$.

Процесс проверки допустимого решения задачи (1)–(5) на оптимальность начинается с допустимого решения $V_j = \min_{i \in I} c_{ij}$ задачи (11)–(12).

Необходимость неравенств (15) следует из соотношений

$$y_i^*\left[c_i^0 - \sum_{j=1}^{N} \max\left(V_j^* - c_{ij}; 0\right)\right] = 0;$$
$$\left(y_i^* - x_{lj}^*\right)\max\left(V_j^* - c_{ij}; 0\right) = 0,$$

определяющих оптимальность полученных решений для непрерывной и целочисленной задач. Знак «*» указывает на оптимальность решений соответствующих задач. При выполнении ограничения (15), совпадение величины

$$Z(I_{\ddot{a}}) = \sum_{j=1}^{N} V_j\,, \qquad (16)$$

полученной в результате решения двойственной задачи (11)-(12), со значением целевой функции задачи (1)-(5), соответствующей проверяемому целочисленному решению

$$C_{\partial} = \sum_{i \in I_{\text{Д}}} c_i^0 + \sum_{j=1}^{N} \min_{i \in I_{\text{д}}} c_{ij}\,,\ I_{\text{Д}} = I, \quad (17)$$

указывает на оптимальность найденного целочисленного решения, т.е.

$$I^* = I_{\text{Д}},\ c^* = c_{\text{Д}}.$$

В случае несовпадения значений целевых функций; поиск оптимального решения задачи (1)-(5) продолжается методом ветвей и границ.

Поиск первого допустимого решения и дальнейшее ветвление осуществляется с помощью показателей вида

$$\rho_i = \sum_{j=1}^{N} \max\left(V_j - c_{ij}; 0\right) - c_i^0, \quad i \in I_3, \quad (18)$$

где $V_j = \min\left[\min_{i \in I_1} c_{ij}; \quad \min_{i \in I}\left(c_i^0 + c_{ij}\right)\right]$.

Таким образом, общая схема алгоритма решения задачи (1)-(5) включает следующие основные этапы:

1. Обоснование допустимого варианта $I_д$, $C_д$ исходной задачи.

2. Исследование этого варианта на оптимальность путем решения двойственной задачи.

3. Уточнение неоптимального варианта методом ветвей и границ и итерационная проверка его на оптимальность.

Выбор варианта заканчивается если доказана его оптимальность с помощью решения двойственной задачи или просмотрены все перспективные ветви дерева вариантов.

На каждом шаге исследований определяются показатели ρ_i, $i \in I_3$, характеризующие целесообразность включения переменных y_i, $i \in I_3$ в формируемое допустимое решение задачи (1)-(5). Если среди показателей (18) имеются $\rho_I < 0$, то принимаем соответствующие им переменные y_i, равными нулю, так как включение их в допустимое решение не приведет к его улучшению. Индексы $\{i|\rho_i, < 0\}$ выводим из множества I_3 и включаем в множество I_0. Кроме того, из множества I_3 исключается и вводится в множество I_1 индекс i^+ для которого

$$\rho_{i^+} = \max_{\{i|\rho_i \geq 0\}} \rho_i, \quad (19)$$

так как переменная y_i^+ может привести к улучшению решения. Последним шагом алгоритма, заканчивающим формирование допустимого решения, является шаг, на котором $I_3 = 0$. Значение целевой функции, соответствующее найденному допустимому целочисленному решению, определяется согласно (17). Если проверка с помощью решения двойственной задачи не подтверждает оптимальности допустимого решения, то его уточняем методом ветвей и границ. Схема ветвления односторонняя, использующая мнемоправило «последний вошел — первый выходит», т.е. для ветвления выбирается свободная вершина, образованная последней доступной. В основу дерева вариантов принимается ветвь, полученная на первом этапе, в дальнейшее ветвление производится с помощью показателей ρ_i. как и при поиске первого допустимого решения.

Отсечение неисследуемых ветвей осуществляется сравнением оценки (14), полученной в результате решения двойственной задачи (11)–(12) с учетом ограничения (13), со значением целевой функции (17), соответствующей наилучшему из уже найденных допустимых решений.

Выводы

1. В разрабатываемых по программам импортозамещения кораблестроительных САПР недостаточно внимания уделяется вопросам информационной безопасности. Остаются возможности утраты части информационного массива в результате преднамеренных и непреднамеренных действий нарушителя требований информационной безопасности.

2. Технологии обеспечения информационной безопасности для программных оболочек могут быть внедрены в САПР, но их разнообразие и режимы работы могут существенно замедлить выполнение основного функционала системы и повлиять как на сроки разработки проекта в целом так и на его экономическую эффективность.

3. Для рациональной интеграции обоснованного состава набора технологий защиты в САПР необходимо провести предварительные исследования с использованием предложенного математического аппарата, позволяющего подобрать и адаптировать такой параметрический ряд защитных средств, который не оказывает существенного влияния на функционал САПР и в то же время обеспечивает требуемый уровень безопасности.

Отметим, что предложенный подход к решению задачи формирования параметрических рядов и алгоритм выбора рационального ряда позволяют обоснованно предложить вариант интеграции защитных модулей в разрабатываемые САПР и могут найти широкое применение в проектном IT- менеджменте. Этот подход к формированию технического облика перспективных технических средств был апробирован [28, 29] и показал высокую эффективность.

СПИСОК ИСТОЧНИКОВ

1. **Анцигин А. В., Борисов А. М., Кежаев В. А., Свертилов Н. И**. Методы и модели оптимизации в управлении развитием сложных технических систем.— СПб., 2004.— 279 с.

2. **Ведерников Ю. В**. Модели и алгоритмы интеллектуализации автоматизированного управления диверсификацией деятельности промышленного предприятия / Ю. В. Ведерников [и др.] // Вопросы оборонной техники. Серия 16: Технические средства противодействия терроризму. 2014. № 5–6 (71–72). С. 61–72.

3. **Anisimov, V.G., Anisimov, E.G., Saurenko, T.N., Sonkin, M. A.** The model and the planning method of volume and variety assessment of innovative products in an industrial enterprise // Journal of Physics: Conference Series, 2017, 803(1), 012006. DOI: 10.1088/1742–6596/803/1/012006.

4. **Сауренко Т. Н**. Прогнозирование инцидентов информационной безопасности / Т. Н. Сауренко [и др.] // Проблемы информационной безопасности. Компьютерные системы. — 2019. — № 3. — С. 24–28.

5. **Anisimov V. G., Anisimov E. G., Saurenko T. N., Zotova E. A**. Models of forecasting destructive influence risks for information processes in management systems // Information and Control Systems. 2019. No 5 (102). P. 18–23.

6. **Супрун А. Ф**. Проблема инновационного развития систем обеспечения информационной безопасности в сфере транспорта // Проблемы информационной безопасности. Компьютерные системы. — 2017. — № 4. — С. 27–32.

7. **Анисимов Е. Г.** Экономическая политика в системе национальной безопасности Российской Федерации / Е. Г. Анисимов [и др.] // Вестник академии военных наук. — 2017. — № 1 (58). — С. 137–144.

8. **Тебекин А. В**. Методический подход к моделированию процессов формирования планов инновационного развития предприятий / А. В. Тебекин [и др.] // Журнал исследований по управлению. — 2019. — Т. 5. — № 1. — С. 65–72.

9. **Зегжда П. Д**. Модель формирования программы развития системы обеспечения информационной безопасности организации / П. Д. Зегжда, Д. П. Зегжда [и др.] // Проблемы информационной безопасности. Компьютерные системы. — 2021. — № 2 (46). — С. 109–117.

10. **Анисимов А. В**. Проблема сравнения и выбора варианта построения системы безопасности / А. В. Анисимов [и др.] // Актуальные проблемы защиты и безопасности: Труды Четвертой Всероссийской научно-практической конференции.— Санкт-Петербург: Научно-производственное объединение специальных материалов. — 2001.— С. 348–351.

11. **Anisimov V. G.** A risk-oriented approach to the control arrangement of security protection subsystems of information systems / V. G. Anisimov [и др.] // Automatic Control and Computer Sciences, 2016, 50(8). С. 717–721. DOI: 10.3103/S0146411616080289.

12. **Зегжда П. Д**. Подход к оцениванию эффективности защиты информации в управляющих системах / П. Д. Зегжда [и др.] // Проблемы информационной безопасности. Компьютерные системы. — 2020. — № 1 (41). — С. 9–16.

13. **Тебекин А. В**. Нелинейная модель оптимизации параметрических рядов в системах управления // Вестник Российской таможенной академии. — 2015. — № 3. — С. 115–122.

14. **Зегжда П. Д**. Модель оптимального комплексирования мероприятий обеспечения информационной безопасности / П. Д. Зегжда [и др.] // Проблемы информационной безопасности. Компьютерные системы. — 2020. — № 2. С. 9–15.

15. **Зегжда П. Д.** Модели и метод поддержки принятия решений по обеспечению информационной безопасности информационно-управляющих систем / П. Д. Зегжда [и др.] // Проблемы информационной безопасности. Компьютерные системы. — 2018. — № 1. — С. 43–47.

16. **Анисимов В. Г., Анисимов Е. Г**. Формальная структура задач стандартизации и унификации при управлении развитием сложных технических систем // Защита и безопасность. — 2004. № 4 (31). С. 26–31.

17. **Крикун В. М., Васильковский С. А.** Многоуровневая задача стандартизации технических комплексов // Стандарты и качество. 1992. № 1. С. 30–32.

18. **Anisimov, V. G**. Indices of the effectiveness of information protection in an information interaction system for controlling complex distributed organizational objects / V.G Anisimov, E. G. Anisimov [и др.] // Automatic Control and Computer Sciences, 2017, 51(8), pp. 824–828. https://doi.org/10.3103/S0146411617080053.

19. **Зегжда П. Д**. Методический подход к построению моделей прогнозирования показателей свойств систем информационной безопасности / П. Д. Зегжда [и др.] // Проблемы информационной безопасности. Компьютерные системы. 2019. № 4. С. 45–49.

20. **Anisimov V. G**. The problem of innovative development of information security systems in the transport sector / V. G. Anisimov [и др.] // Automatic Control and Computer Sciences. 2018. Vol. 52. No 8. P. 1105–1110.

21. **Зегжда П. Д**. Эффективность функционирования компьютерной сети в условиях вредоносных информационных воздействий / П. Д. Зегжда [и др.] // Проблемы информационной безопасности. Компьютерные системы. — 2021. — № 1 (45). — С. 96–101.

22. **Анисимов В. Г., Анисимов Е. Г**. Математические модели и методы в управлении развитием сложных технических систем.— СПб., 2004.— 280 с.

23. **Кежаев В. А., Свертилов Н. И., Шатохин Д. В**. Методы и модели стандартизации и унификации в управлении развитием военно-технических систем.— Москва: Военная академия Генерального штаба Вооруженных Сил Российской Федерации; 2004.— 279 с.

24. **Гарькушев А. Ю., Проценко Д. С**. Модель и метод оптимизации плана подготовки космических систем. // Известия Российской академии ракетных и артиллерийских наук. — 2015. — № 4 (89). — С. 34–39.

25. **Пеннер Я. А., Гарькушев А. Ю**. Метод распределения неоднородных ресурсов при управлении организационно-техническими системами // Вопросы оборонной техники. Серия 16: Технические средства противодействия терроризму. — 2016. — № 3–4 (93–94). — С. 20–26.

26. **Анисимов В. Г., Анисимов Е. Г**. Модификация метода решения одного класса задач целочисленного программирования // Журнал вычислительной математики и математической физики. — 1997. — Т. 37, № 2. — С. 179–183.

27. **Алексеев, А. О**. Применение двойственности для повышения эффективности метода ветвей и границ при решении задачи о ранце / А. О. Алексеев, О. Г. Алексеев [и др.] // Журнал вычислительной математики и математической физики. — 1985. — Т. 25, № 11. — С. 1666–1673.

28. **Ведерников Ю. В., Гарькушев А. Ю., Карпова И. Л., Супрун А. Ф**. Формализация задачи выбора варианта структурного построения информационного комплекса управления многоуровневой иерархической системой по критерию информационной безопасности // Проблемы информационной безопасности. Компьютерные системы. — 2018. — № 3. — С. 78–82.

29. **Гарькушев А. Ю., Анцев И. Г., Смирнов Ю. Д., Кремчеев Ю. А**. Перспективы применения систем экологического мониторинга в разведывательных беспилотных летательных аппаратах // Вопросы оборонной техники. Серия 16: Технические средства противодействия терроризму. — 2013. — № 1–2 (55–56). — С. 98.

30. **Зегжда Д. П.** Автоматизация управления безопасностью интеллектуальных систем с использованием графа атак и анализа рисков / Д. П. Зегжда, Д. А. Москвин, Е. М. Орел // Защита информации. Инсайд.— 2022.— № 2(104).— С. 22–28.

REFERENCES

1. **Ancigin A. V., Borisov A. M., Kezhaev V. A., Svertilov N. I**. Metody i modeli optimizacii v upravlenii razvitiem slozhnyh tekhnicheskih sistem. [Methods and models of optimization in managing the development of complex technical systems] Sankt-Peterburg, 2004.— 279 p. (In Russian)

2. **Vedernikov Yu. V**. Modeli i algoritmy intellektualizacii avtomatizirovannogo upravleniya diversifikaciej deyatel'nosti promyshlennogo predpriyatiya [Models and algorithms for the intellectualization of automated control of the diversification of the activities of an industrial enterprise] / Yu. V. Vedernikov [et al.] // Voprosy oboronnoj tekhniki. Seriya 16: Tekhnicheskie sredstva protivodejstviya terrorizmu. 2014. No 5–6 (71–72). P. 61–72. (In Russian)

3. **Anisimov V.G., Anisimov, E.G., Saurenko, T.N., Sonkin, M.A**. The model and the planning method of volume and variety assessment of innovative products in an industrial enterprise // Journal of Physics: Conference Series, 2017, 803(1), 012006. DOI: 10.1088/1742–6596/803/1/012006. (In Russian)

4. **Saurenko T. N**. Prognozirovanie incidentov informacionnoj bezopasnosti [Predicting Information Security Incidents] / T. N. Saurenko [et al.] // Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy. 2019. No 3. P. 24–28. (In Russian)

5. **Anisimov V. G., Anisimov E. G., Saurenko T. N., Zotova E. A**. Models of forecasting destructive influence risks for information processes in management systems // Information and Control Systems. 2019. No 5 (102). P. 18–23.

6. **Suprun A. F**. Problema innovacionnogo razvitiya sistem obespecheniya informacionnoj bezopasnosti v sfere transporta [The problem of innovative development of information security systems in the field of transport] // Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy. 2017. No 4. P. 27–32. (In Russian)

7. **Anisimov E. G**. Ekonomicheskaya politika v sisteme nacional'noj bezopasnosti Rossijskoj Federacii [Economic policy in the national security system of the Russian Federation] / E. G. Anisimov [et al.] // Vestnik akademii voennyh nauk. 2017. No 1 (58). P. 137–144. (In Russian)

8. **Tebekin A. V**. Metodicheskij podhod k modelirovaniyu processov formirovaniya planov innovacionnogo razvitiya predpriyatij [Methodical approach to modeling the processes of formation of plans for innovative development of enterprises] / A. V. Tebekin [et al.] // ZHurnal issledovanij po upravleniyu. 2019. T. 5. № 1. S. 65–72. (In Russian)

9. **Zegzhda P. D**. Model' formirovaniya programmy razvitiya sistemy obespecheniya informacionnoj bezopasnosti organizacii [Model of formation of the program for the development of the information security system of the organization] / P. D. Zegzhda, D. P. Zegzhda [et al.] // Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy. 2021. No 2 (46). P. 109–117. (In Russian)

10. **Anisimov A. V**. Problema sravneniya i vybora varianta postroeniya sistemy bezopasnosti [The problem of comparing and choosing a variant of building a security system] / A. V. Anisimov [et al.] // Aktual'nye problemy zashchity i bezopasnosti: Trudy CHetvertoj Vserossijskoj nauchno-prakticheskoj konferencii.— Sankt-Peterburg: Nauchno-proizvodstvennoe ob"edinenie special'nyh materialov 2001.— P. 348–351. (In Russian)

11. **Anisimov V. G.** A risk-oriented approach to the control arrangement of security protection subsystems of information systems / V. G. Anisimov [et al.] // Automatic Control and Computer Sciences, 2016, 50(8). P. 717–721. DOI: 10.3103/S0146411616080289.

12. **Zegzhda P. D**. Podhod k ocenivaniyu effektivnosti zashchity informacii v upravlyayushchih sistemah [Approach to evaluating the effectiveness of information protection in control systems] / P. D. Zegzhda [et al.] // Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy. 2020. No 1 (41). P. 9–16. (In Russian)

13. **Tebekin A. V**. Nelinejnaya model' optimizacii parametricheskih ryadov v sistemah upravleniya [Nonlinear optimization model of parametric series in control systems] // Vestnik Rossijskoj tamozhennoj akademii. 2015. No 3. P. 115–122. (In Russian)

14. **Zegzhda P. D**. Model' optimal'nogo kompleksirovaniya meropriyatij obespecheniya informacionnoj bezopasnosti [Model of optimal integration of measures to ensure information security] / P. D. Zegzhda [et al.] // Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy. 2020. No 2. P. 9–15. (In Russian)

15. **Zegzhda P. D**. Modeli i metod podderzhki prinyatiya reshenij po obespecheniyu informacionnoj bezopasnosti informacionno-upravlyayushchih system [Models and method of decision support for ensuring information security of information and control systems] / P. D. Zegzhda [et al.] // Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy. 2018. No 1. P. 43–47. (In Russian)

16. **Anisimov V. G., Anisimov E. G**. Formal'naya struktura zadach standartizacii i unifikacii pri upravlenii razvitiem slozhnyh tekhnicheskih system [Formal structure of the tasks of standardization and unification in managing the development of complex technical systems] // Zashchita i bezopasnost'. 2004. No 4 (31). P. 26–31. (In Russian)

17. **Krikun V. M., Vasil'kovskij S. A**. Mnogourovnevaya zadacha standartizacii tekhnicheskih kompleksov [Multilevel task of standardization of technical complexes] // Standarty i kachestvo. 1992. No 1. P. 30–32. (In Russian)

18. **Anisimov, V. G**. Indices of the effectiveness of information protection in an information interaction system for controlling complex distributed organizational objects / V.G Anisimov, E. G. Anisimov [i dr.] // Automatic Control and Computer Sciences, 2017, 51(8), pp. 824–828. https://doi.org/10.3103/S0146411617080053.

19. **Zegzhda P. D**. Metodicheskij podhod k postroeniyu modelej prognozirovaniya pokazatelej svojstv sistem informacionnoj bezopasnosti [Methodical approach to building models for predicting indicators of properties of information security systems] / P. D. Zegzhda [et

al.] // Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy. 2019. No 4. P. 45–49. (In Russian)

20. **Anisimov V. G**. The problem of innovative development of information security systems in the transport sector / V. G. Anisimov [et al.] // Automatic Control and Computer Sciences. 2018. Vol. 52. No 8. P. 1105–1110.

21. **Zegzhda P. D**. Effektivnost' funkcionirovaniya komp'yuternoj seti v usloviyah vredonosnyh informacionnyh vozdejstvij [Efficiency of computer network functioning under conditions of malicious information impacts] / P. D. Zegzhda [et al.] // Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy. 2021. No 1 (45). P. 96–101. (In Russian)

22. **Anisimov V. G., Anisimov E. G.** Matematicheskie modeli i metody v upravlenii razvitiem slozhnyh tekhnicheskih system [Mathematical models and methods in managing the development of complex technical systems]. — Sankt-Peterburg, 2004. — 280 p. (In Russian)

23. **Kezhaev V. A., Svertilov N. I., Shatohin D. V**. Metody i modeli standartizacii i unifikacii v upravlenii razvitiem voenno-tekhnicheskih system [Methods and models of standardization and unification in managing the development of military-technical systems] Moskva: Voennaya akademiya General'nogo shtaba Vooruzhennyh Sil Rossijskoj Federacii; 2004. — 279 p. (In Russian)

24. **Gar'kushev A. Yu., Procenko D. S**. Model' i metod optimizacii plana podgotovki kosmicheskih system [Model and method for optimizing the space systems preparation plan]. // Izvestiya Rossijskoj akademii raketnyh i artillerijskih nauk. 2015. No 4 (89). P. 34–39. (In Russian)

25. **Penner Ya.A., Gar'kushev A. Y**u. Metod raspredeleniya neodnorodnyh resursov pri upravlenii organizacionno-tekhnicheskimi sistemami [The method of distribution of heterogeneous resources in the management of organizational and technical systems] // Voprosy oboronnoj tekhniki. Seriya 16: Tekhnicheskie sredstva protivodejstviya terrorizmu. 2016. No 3–4 (93–94). P. 20–26. (In Russian)

26. **Anisimov V. G., Anisimov E. G**. Modifikaciya metoda resheniya odnogo klassa zadach celochislennogo programmirovaniya [Modification of a Method for Solving a Class of Integer Programming Problems] // ZHurnal vychislitel'noj matematiki i matematicheskoj fiziki. 1997. Vol. 37. No 2. P. 179–183. (In Russian)

27. **Alekseev, A. O**. Primenenie dvojstvennosti dlya povysheniya effektivnosti metoda vetvej i granic pri reshenii zadachi o rance [Application of Duality to Improve the Efficiency of the Branch and Bound Method in Solving the Knapsack Problem] / A. O. Alekseev, O. G. Alekseev [et al.] // Zhurnal vychislitel'noj matematiki i matematicheskoj fiziki. 1985. Vol. 25. No 11. P. 1666–1673. (In Russian)

28. **Vedernikov Yu.V., Gar'kushev A. Yu., Karpova I. L., Suprun A. F**. Formalizaciya zadachi vybora varianta strukturnogo postroeniya informacionnogo kompleksa upravleniya mnogourovnevoj ierarhicheskoj sistemoj po kriteriyu informacionnoj bezopasnosti [Formalization of the problem of choosing a variant of the structural construction of an information complex for managing a multilevel hierarchical system according to the criterion of information security] // Problemy informacionnoj bezopasnosti. Komp'yuternye sistemy. 2018. No 3. P. 78–82. (In Russian)

29. **Gar'kushev A. Yu., Ancev I. G., Smirnov Yu.D., Kremcheev Yu. A**. Perspektivy primeneniya sistem ekologicheskogo monitoringa v razvedyvatel'nyh bespilotnyh letatel'nyh apparatah [Prospects for the use of environmental monitoring systems in reconnaissance unmanned aerial vehicles] // Voprosy oboronnoj tekhniki. Seriya 16: Tekhnicheskie sredstva protivodejstviya terrorizmu. 2013. No 1–2 (55–56). P. 98. (In Russian)

30. **Zegzhda D. P.** Avtomatizaciya upravleniya bezopasnost'yu intellektual'nyh sistem s ispol'zovaniem grafa atak i analiza riskov [Automation of security management of intelligent systems using attack graph and risk analysis] / D. P. Zegzhda, D. A. Moskvin, E. M. Orel // Zashchita informacii. Insajd. — 2022. — No 2(104). — P. 22–28. (In Russian)

Статья поступила в редакцию 17.02.2022; одобрена после рецензирования 24.02.2022; принята к публикации 28.02.2022.

The article was submitted 17.02.2022; approved after reviewing 24.02.2022; accepted for publication 28.02.2022.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

ГАРЬКУШЕВ Александр Юрьевич — к.т.н., доцент; Санкт-Петербургский государственный морской технический университет

СУПРУН Александр Федорович — к.т.н., доцент; Санкт-Петербургский политехнический университет Петра Великого

СЫСУЕВ Сергей Юрьевич — к.воен.н., доцент; «Михайловская военная артиллерийская академия» МО РФ

GARKUSHEV Alexander Yu.— Ph.D., Associate Professor; St. Petersburg State Marine Technical University

SUPRUN Alexander F.— Ph.D., Associate Professor; St. Petersburg Polytechnic University of Peter the Great

Email: suprun_af@spbstu.ru

SYSUEV Sergey Yu.— Ph.D., Associate Professor; "Mikhailovskaya Military Artillery Academy" of the Ministry of Defense of the Russian Federation

Email: sysuev1971@mail.ru

ПРОБЛЕМЫ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ
КОМПЬЮТЕРНЫЕ СИСТЕМЫ

№ 1, 2022

Редактор *Д. А. Клубничкина*
Корректор *Д. А. Клубничкина*
Компьютерная верстка *Н. В. Стасеевой*
Дизайн обложки *Т. М. Ивановой*

Налоговая льгота — Общероссийский классификатор продукции
ОК 005-93, т. 2; 95 3004 — научная и производственная литература

Подписано в печать 31.03.2022. Формат 60×84/8.
Усл. печ. л. 16,5. Тираж 250. Заказ 1430.

Отпечатано в Издательско-полиграфическом центре
Политехнического университета.
195251, Санкт-Петербург, Политехническая ул., 29.
Тел.: (812) 552-77-17; 550-40-14.