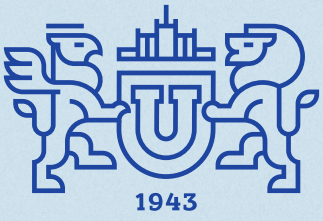


Южно-Уральский государственный университет

БЕЗОПАСНОСТЬ ИНФОРМАЦИОННОГО ПРОСТРАНСТВА

Сборник трудов
XXII Всероссийской научно-практической конференции
студентов, аспирантов и молодых учёных

Министерство науки и высшего образования Российской Федерации
Южно-Уральский государственный университет
Высшая школа электроники и компьютерных наук
Кафедра «Защита информации»

004
Б40

БЕЗОПАСНОСТЬ ИНФОРМАЦИОННОГО ПРОСТРАНСТВА

Сборник трудов XXII Всероссийской научно-практической конференции студентов, аспирантов и молодых учёных

Челябинск
Издательский центр ЮУрГУ
2024

УДК 004.056(063)
Б40

Одобрено
Советом факультета
Высшей школы электроники и компьютерных наук
Рецензенты:
А.Н. Ручай, канд. физ.-мат. наук (ЧелГУ),
Е.Ю. Мищенко, канд. техн. наук (ООО «Стратегия безопасности»)

Безопасность информационного пространства: сборник трудов XXII Всероссийской научно-практической конференции студентов, аспирантов и молодых учёных / сост. А.Н. Соколов. – Челябинск: Издательский центр ЮУрГУ, 2024. – 310 с.

Б40

ISBN 978-5-696-05435-3

В сборник трудов вошли 53 работы участников XXII Всероссийской научно-практической конференции студентов, аспирантов и молодых учёных «Безопасность информационного пространства», проходившей 30 ноября 2023 г. на базе Южно-Уральского государственного университета (г. Челябинск). Представлены работы из нескольких вузов России: Магнитогорского государственного технического университета имени Г.И. Носова, Тюменского государственного университета, Уральского государственного университета путей сообщения (г. Екатеринбург), Уральского федерального университета имени Б.Н. Ельцина (г. Екатеринбург), Уральского государственного экономического университета (г. Екатеринбург), Челябинского государственного университета, Южно-Уральского государственного университета (г. Челябинск).

УДК 004.056(063)

ISBN 978-5-696-05435-3

СЕКЦИЯ «ОРГАНИЗАЦИОННЫЕ, ПРАВОВЫЕ, ГУМАНИТАРНЫЕ И СОЦИАЛЬНЫЕ АСПЕКТЫ ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ»

УДК 004.056

ОБУЧЕНИЕ СТУДЕНТОВ ВУЗОВ ОСНОВАМ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ С ПОМОЩЬЮ ИГРОВЫХ МЕТОДОВ

И.А. Хабаров, Т.Ю. Зырянова
Научный руководитель: канд. техн. наук, доц. Т.Ю. Зырянова
Уральский государственный университет путей сообщения, г. Екатеринбург

Проблема сложности усваивания материалов по информационной безопасности приводит к быстрой потере интереса у учащихся к изучению, на их взгляд, нетривиальной сферы – информационной безопасности. В связи с этим было принято решение об анализе современного состояния тематики обучения студентов, а также проведении исследования с целью разработки собственного метода обучения. Статья посвящена созданию структуры обучения студентов вузов основам информационной безопасности с помощью игровых методов. Игровые методы уже зарекомендовали себя как наиболее соответствующий инструмент, отвечающий, как и требованиям к образовательным процессам, так и запросам учащихся. Также данный инструмент является настолько гибким, что позволяет легко адаптировать его и в другие сферы обучения. И, помимо этого, он обладает необходимым функционалом для его настройки под конкретную категорию учащихся. В данной статье игровые методы реализованы в виде компьютерной игры, и она была совмещена с разработанной структурой обучения. Данная методика была внедрена в процесс обучения будущих специалистов в области информационной безопасности, а конкретно у студентов первых курсов информационной безопасности в УрГУПСе. Результаты анализа демонстрируют общую положительную тенденцию развития этой тематики, а проведенное исследование доказывает положительное влияние разработанной методики на качество обучения студентов основам информационной безопасности.

Ключевые слова: анализ, игра, игровые методы, информационная безопасность, исследование, обучение, структура обучения, студенты.

В современном мире уровень развития технологий позволяет злоумышленникам регулярно создавать новые методы реализации угроз информаци-

онной безопасности (ИБ). И особенность информационной безопасности в том, что она не ограничивается в стенах рабочего места сотрудника, и не заканчивается, когда сотрудник уходит домой. Она продолжает существовать в его телефоне, на его личном компьютере и так далее.

Примером может быть следующая ситуация: на рабочем компьютере человек соблюдает все правила ИБ, и этот компьютер надёжно защищён паролем. Но тот же самый пароль установлен на домашнем компьютере сотрудника, который никак не защищен, так и во время его эксплуатации не соблюдается элементарная интернет-гигиена. Вследствие пароль становится известен злоумышленникам. И в случае несанкционированного доступа (физического или виртуального) к компьютеру сотрудника, злоумышленник сможет получить доступ к защищаемой информации, так как знает пароль.

В любой, даже самой защищенной системе, самым уязвимым звеном является человек. И если будущие сотрудники компании не владеют основами ИБ, то у фирмы могут возникнуть множественные проблемы: материальные, финансовые, организационные, репутационные, политические. В связи с этим вопрос, связанный с обучением студентов основам ИБ, является крайне актуальным.

Основной проблемой является то, что стандартные формы обучения предполагают в основном традиционную подачу материала [1]. Это приводит к его неполноценному усвоению и даже к потере интереса у студентов, что в свою очередь может привести к различным по степени проблемам в профессиональной сфере студента. Поэтому, необходимо оптимизировать процесс обучения таким образом, чтобы он был интересным и разнообразным.

В связи с этим возникла цель провести анализ текущего состояния выбранной тематики, а также провести исследование по разработке собственной методики обучения студентов.

Для достижения цели анализа были поставлены следующие задачи:

- изучить имеющиеся материалы по тематике обучения студентов;
- изучить существующие игровые методы в обучении, а также их эффективность.

Для достижения цели исследования разрабатываемая методика обучения должна соответствовать следующим критериям [2]:

- сформировать мотивацию у студентов на обучение;
- позволить оценить уровень подготовленности студентов;
- позволить оценить степень овладения материалом и перевести его из пассивного состояния – знания в активное – умение;
- активизировать самообразование у студентов;
- сформировать плюрализм мнений и действий, многовариантность мыслительных операций, интерес к более эффективному построению профессиональной деятельности;

- позволить развивать индивидуальное профессиональное мышление, умение анализировать и прогнозировать.

В результате анализа удалось установить, что эффективность игровых методов заключается в том, что они обеспечивают интерес и включенность у каждого обучающегося, и что, в свою очередь, значительно повышает у них результативность, вне зависимости от сферы обучения [3].

Связано это с тем, что, когда человек проигрывает какой-либо процесс, то у него формируется позитивный или негативный опыт [4]. В случае, если в результате процесса опыт позитивный, то это способствует выработке эндорфинов. А эндорфины положительно влияют на фиксировании информации и сохранении её в долгосрочной памяти. Но если же опыт оказался негативным, то в правильном учебном процессе будет произведен анализ неверных действий, что в дальнейшим, позволит обучаемому получить необходимый положительный опыт.

Стоит отметить, что игровые методы подходят не только для умственного обучения, но и физического. Так, например, в одной из статей была доказана эффективность применения игрового метода в обучении плаванию детей дошкольного возраста [5]. Результатом данного исследования является тот факт, что дети в возрасте от 5 до 6 лет, которые получали обучение по игровому методу, лучше выполняли контрольные задания и имели высокие показатели усвоения навыков плавания уже на промежуточном этапе, в отличие от детей традиционной группы. В группе с традиционной методикой обучения количество детей, выполнивших контрольные упражнения, составило, в среднем, 53 %, а в группе с применением игрового метода – 75 %.

Одним из наиболее эффективных игровых методов является игра по ролям [6]. Когда заранее продумано место, содержание игры и есть всё необходимое оборудование. Игра снижает стресс, беспокойство и страх неудачи, которые испытывают студенты. Учащиеся больше вовлекаются в процесс и меньше боятся высказываться.

Для того, чтобы достигнуть цели исследования была разработана предлагаемая структура процесса обучения, которая выглядит следующим образом:

- в начале проводится оценка общего уровня осведомлённости студентов в сфере ИБ посредством тестирования. Само тестирование состоит из двух частей. В первой части необходимо выбрать вариант или варианты ответа, а во второй находятся открытые вопросы без вариантов ответа и студенту необходимо написать ответ самостоятельно;
- после тестирования подводятся итоги и выявляются «слабые» области осведомленности. Затем адаптация результатов тестирования в дальнейшей структуре обучения;
- обучение основам ИБ, совмещая учебный процесс с игрой на основе компьютерной интеллектуальной игры «SIGame» [7].

Данная игра была выбрана по причине её удобства установки, гибкой настройки, а также из-за соответствия необходимым условиям по внедрению в учебный процесс, в рамках проводимого исследования и для демонстрации результатов. Также, она позволяет создавать собственные блоки вопросов, и в зависимости от целей обучения проводить разбор конкретных ситуаций, с помощью совместных обсуждений на интерактивном уровне;

- итоговое тестирование из двух частей;
- демонстрация результатов студентам, выделение «сильных» и «слабых» областей осведомленности до и после обучения. Преуспевающим и активным студентам предоставляются поощрения в учебном процессе;
- итоговый анализ в конце процесса обучения и составление рекомендаций для студентов по сохранению нынешнего уровня осведомлённости и по его развитию.

По причине необходимости практического применения разработанной структуры, для изучения её эффективности, и для достижения цели исследования, была произведена процедура по внедрению вышеописанной структуры обучения в учебный процесс двух групп студентов, находящихся на первом курсов ИБ в УрГУПСе. У обеих групп было проведено первое тестирование, для выявления среднего значения осведомленности. По результатам, которые отображены в табл. 1 видно, что вторая группа справилась с тестированием лучше, чем первая. Поэтому было принято решение применить разработанную структуру на первой группе, а вторая группа продолжила обучаться по стандартным методам. В конце обучения было проведено итоговое тестирование, результаты которого также отображены в табл. 1. Из данных результатов можно сделать вывод, что эффективность использования разработанной структуры, вместо стандартных методов, является наглядной.

Таблица 1

Результаты тестов двух групп

	Группа №1		Группа №2	
	Первая часть	Вторая часть	Первая часть	Вторая часть
Первое тестирование	70%	65%	74%	69%
Итоговое тестирование	88%	80%	79%	75%

Заключение. Цель анализа была достигнута и его результаты демонстрируют общую положительную тенденцию развития этой тематики. Существует обширная материальная база по применению игровых форм в процессах обучения, в разных областях. Цель исследования также была достигнута и его результаты доказывают положительное влияние разработан-

ной методики на качество обучения студентов основам информационной безопасности и эффективность использования разработанной структуры, вместо стандартных методов.

Библиографический список

1. Хабаров И.А., Костюченко К.Л. Обучающая программа «Поиск закладочных устройств» // Вестник УрФО. 2022. № 2(44). С. 43–48.

2. Багатырова М.Н. Возможности игровых методов в обучении студентов // Мир науки, культуры образования. 2016. № 1. С. 187–188.

3. Шабанова О.А. Игровая технология и ее эффективность в дополнительном образовании. МБУДО Центр творчества «Правобережный».

4. Лорич И.В., Тюгаев И.М. Применение игровых технологий в обучении специалистов по информационной безопасности // Физико-математические и технические науки. 2019. Выпуск №4.

5. Качковская, Н.А. Эффективность применения игрового метода в обучении плаванию детей дошкольного возраста // Молодой ученый. 2017. № 8(142). С. 335–337.

6. Курманов М.Б. Игровые методы обучения в вузе. Жетысуский государственный университет им. И. Жансугурова.

7. Хиль В.А. Компьютерная интеллектуальная игра «SIGame». Режим доступа: https://vladimirkhil.com/si/game (дата обращения: 16.10.2023).

УДК 004.056

АНАЛИЗ ТЕХНОЛОГИЙ СОЦИАЛЬНОЙ ИНЖЕНЕРИИ

Л.А. Григоренко, В.С. Русецкас
Научный руководитель: ассистент кафедры ИИиБ Л.А. Григоренко
Магнитогорский государственный технический университет имени Г.И. Носова, г. Магнитогорск

В статье разобрана проблема социального мошенничества, в основе которой лежит психологическое манипулирование людьми с целью получить доступ к нужной информации, которая с каждым годом прогрессирует всё сильнее. Рассмотрены основные типы социальной инженерии, а также приведены основные стратегии и методы борьбы с данным видом мошенничеств. Установлено, что информационно-просветительская деятельность об угрозах мошенников, использующих методы социальной инженерии, является важной задачей для повышения безопасности и защиты от манипуляций.

Ключевые слова: информационная безопасность, методы социальной инженерии, социальная инженерия.

В мире, где информация стала нашим самым ценным активом, обеспечение безопасности данных и защита конфиденциальности стали приоритетом для организаций и частных лиц. Однако, помимо технических аспектов информационной безопасности, существует еще не менее важный аспект – социальная инженерия.

Целью статьи является анализ основных технологий социальной инженерии.

Социальная инженерия – это искусство манипуляции людьми с целью получения доступа к ценной информации или ресурсам. Злоумышленники, используя психологические приемы, могут обмануть даже самых осторожных сотрудников и, таким образом, обойти множество технических барьеров.

Согласно данным исследования компании «Антифишинг» за 2020 год, около 86% россиян подвергались воздействию социальной инженерии, а 18,6% понесли финансовые или информационные потери [1]. Ниже приведена статистика наиболее используемых мошенниками схем (рис. 1).

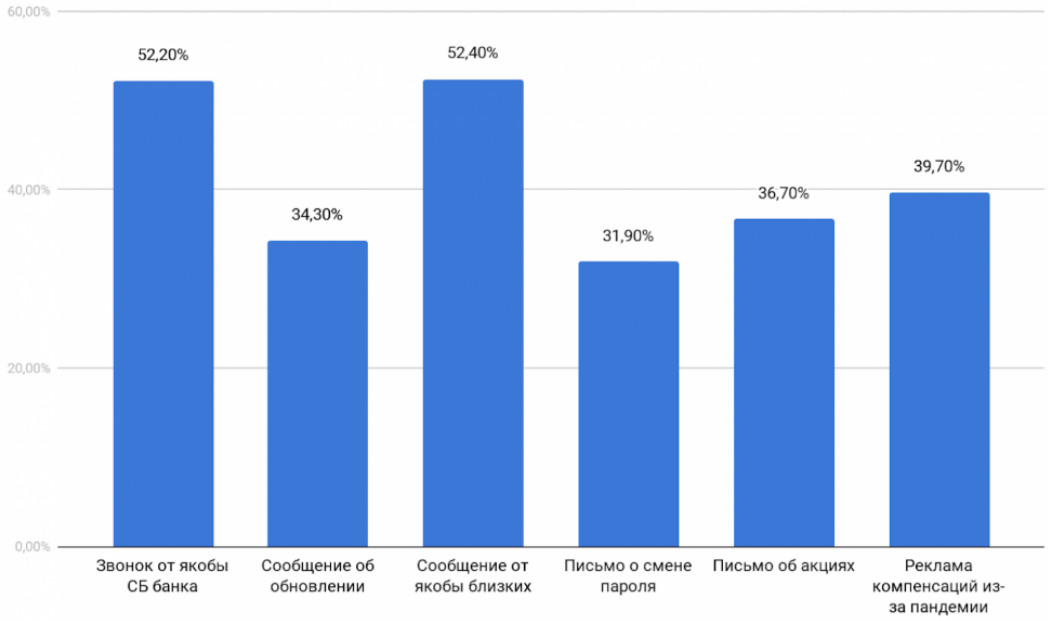


Рис. 1. Наиболее часто используемые методы применения социальной инженерии у злоумышленников

Специалистам по информационной безопасности также следует уделить особое внимание этой проблеме. Согласно отчету в блоге команды по анализу угроз Google, северокорейские хакеры начали массово атаковать их сотрудников с помощью социальной инженерии и размещении зараженного инструментария для специалистов по информационной безопасности (ИБ). Они создавали аккаунты с обычными американскими именами, публиковали статьи на тему ИБ и втирались в доверие к специалистам по ИБ из Google [2]. Как только общение становилось «теплым», бдитель-

ность сотрудников притуплялась, хакеры переходили в мессенджеры, куда они отправляли жертве вредоносный файл.

Социальный инженер действует тайно и убедительно, обманывая потенциальную жертву или, по крайней мере, убеждая ее не предпринимать меры по предотвращению таких манипуляций. Это явление приобретает особую важность в контексте информационной безопасности, и понимание методов социальной инженерии и средств предотвращения ее атак становятся ключевыми аспектами для обеспечения безопасности в цифровом мире.

Рассмотрим основные методы социальной инженерии.

1. Фишинговые атаки. Социальные инженеры активно используют фишинговые электронные письма и веб-сайты. Они маскируют свои атаки под официальные запросы или коммуникацию с коллегами, обманывая сотрудников и получая доступ к корпоративным данным [3].

2. Звонки и посещения. Социальные инженеры могут лично посещать офисы компаний или звонить сотрудникам. Они могут выдавать себя за представителей других организаций, вводя в заблуждение и получая доступ к ценной информации.

3. Мошенничество с данными. Социальные инженеры могут пытаться убедить сотрудников предоставить личные данные или конфиденциальную информацию о клиентах. Эта информация может быть использована против организации в будущем [4].

4. Манипуляция сотрудниками. Психологические методы используются для манипуляции сотрудниками. Инженеры могут убедить сотрудников выполнять действия, которые наносят ущерб организации, такие как предоставление доступа к конфиденциальным данным.

5. Внутренние угрозы. Сотрудники могут стать «внутренними угрозами», предоставляя доступ к корпоративным ресурсам и данным злоумышленникам, а порой даже действуя совместно с ними [5].

В борьбе с социальной инженерией, понимание и использование основ технологий социальной инженерии может стать мощным инструментом. Организации могут применить некоторые принципы социальной инженерии для поддержания безопасности и защиты от манипуляций. Приведем несколько практических способов использования социальной инженерии для борьбы с атаками.

1. Симуляции атак. Необходимо организовать симуляции социальной инженерии внутри организации, чтобы сотрудники могли практиковаться в распознавании и предотвращении манипуляций. Это позволит им на практике применить изученные методы безопасности;

2. Обучение эмпатии и пониманию человеческой психологии. Нужно научить сотрудников основам эмпатии и пониманию человеческой психологии. Это поможет им лучше анализировать и понимать мотивы и действия других и более успешно защищаться от манипуляции;

3. Изучение техник воздействия. Необходимо дать сотрудникам знание о том, какие методы используют социальные инженеры, чтобы они могли легче распознавать и предотвращать манипуляции;

4. Культура безопасности. Нужно попытаться создать культуру безопасности внутри организации, где каждый сотрудник чувствует ответственность за защиту информации. Поддержка со стороны руководства в этом ключевая.

Рассмотрим каждый способ более подробно.

Симуляции атак – это метод, при котором организация создает контролируемые сценарии, имитирующие атаки, которые могли бы быть проведены социальными инженерами. В ходе таких симуляций сотрудники принимают на себя роли потенциальных жертв и социальных инженеров. Основная цель заключается в обучении сотрудников распознаванию признаков социальной инженерии и формировании навыков реагирования на подобные атаки.

Симуляции атак помогают сотрудникам применить свои знания на практике, что делает их более бдительными и устойчивыми к манипуляциям со стороны социальных инженеров. Кроме того, они предоставляют организации возможность оценить эффективность программ обучения и внести необходимые изменения в свои политики и процедуры безопасности.

Обучение эмпатии включает в себя развитие способности сотрудников понимать эмоции, потребности и переживания других. Это помогает им:

- распознавать манипулятивное поведение;
- оставаться бдительными к сигналам манипуляции;
- понимать чувства и мотивы других людей;
- понимание человеческой психологии.

Обучение пониманию человеческой психологии включает в себя знание основных принципов человеческого поведения и реакций. Это помогает сотрудникам:

- понимать, какие механизмы используют социальные инженеры;
- анализировать действия и мотивы других людей;
- распознавать стратегии манипуляции и обмана.

Понимание человеческой психологии и эмоций облегчат процесс изучения техник воздействия социальных инженеров, которые содержат:

- распознавание методов манипуляции, таких как психологические приемы и ложные обещания;
- понимание психологических аспектов манипуляции, таких как воздействие социальных давлений и человеческих страхов и желаний;
- обучение разоблачению манипуляции через задавание вопросов, проверку подлинности и критический анализ ситуации [6].

Эти знания и навыки помогают сотрудникам эффективно распознавать и предотвращать манипуляцию со стороны социальных инженеров.

Культура безопасности – это обстановка и набор ценностей, которые содействуют безопасности в организации. Она содержит в себе:

- сознательность – все сотрудники осознают, что безопасность важна и касается каждого;
- обучение и обучаемость – сотрудники постоянно обучаются и готовы учиться новым аспектам безопасности;
- сотрудничество между коллегами для обмена знаниями и опытом в области безопасности;
- ответственность – каждый сотрудник несет личную ответственность за безопасное поведение и защиту данных;
- доверие и открытость – обеспечение атмосферы, в которой сотрудники могут свободно сообщать о нарушениях без страха и гарантировать открытую коммуникацию;
- следование процедурам и политикам безопасности, установленным в организации.

Эти аспекты в совокупности создают культуру безопасности, которая способствует защите организации от различных угроз и повышает уровень безопасности данных и процессов.

Таким образом, можно сделать вывод, что на данный момент важной задачей, является информационно-просветительская деятельность об угрозах, которые представляют мошенники, действующие с применением методов социальной инженерии, а также расширение знаний о самих методах социальной инженерии для укрепления безопасности и защиты от манипуляций.

Библиографический список

1. Осторожно, мошенники: 86,4% россиян сталкивались с кибератаками // URL: https://iz.ru/1074772/roza-almakunova/ostorozhno-moshenniki-864-rossiian-stalkivalis-s-kiberatakami (дата обращения: 25.10.2023).

2. Active North Korean campaign targeting security researchers // URL: https://blog.google/threat-analysis-group/active-north-korean-campaign - targeting-security-researchers/ (дата обращения: 27.10.2023).

3. Шерстяных А.С. Фишинг как инструмент социальной инженерии / А.С. Шерстяных // Актуальные проблемы борьбы с преступностью: вопросы теории и практики // Материалы XXV международной научно-практической конференции. В 2-х частях, Красноярск, 07–08 апреля 2022 года. Том Часть 2. – Красноярск: Сибирский юридический институт Министерства внутренних дел Российской Федерации, 2022. – С. 299–301. – DOI: 10.51980/978-5-7889-0334-7_2022_5_2_299. – EDN FGWHYV.

4. Михайлова У.В. Обеспечение информационной безопасности при угрозах реализации методов социальной инженерии / У.В. Михайлова, А.В. Перминова // Актуальные проблемы современной науки, техники и образования: Тезисы 80-й международной научно-технической конференции, Магнитогорск, 18–22 апреля

2022 года. Том 1. – Магнитогорск: Магнитогорский государственный технический университет им. Г.И. Носова, 2022. – С. 383. – EDN WUXUTS.

5. Яковлева К.Ю. Предотвращение использования социальной инженерии киберпреступниками в социальных сетях / К.Ю. Яковлева, А.В. Андреев // Актуальные проблемы кибербезопасности в сети Интернет // Сборник научных трудов Всероссийской конференции, Москва, 23 апреля 2020 года. – Москва: Московский университет Министерства внутренних дел Российской Федерации им. В.Я. Кикотя, 2020. – С. 178–180. – EDN CGKOBF.

6. Самойлова А.А. Методы социальной инженерии / А.А. Самойлова // Тенденции развития науки и образования. – 2019. – №56–3. – С. 25–28. – DOI:10.18411/lj-11-2019-48. – EDN NQJQFR.

УДК 004.056

АНАЛИЗ ТЕХНОЛОГИЙ РАСПОЗНАВАНИЯ НЕЙРОННЫХ СЕТЕЙ

Д.А. Ларионов
Научный руководитель: ст. преподаватель Н.В. Ганженко
Уральский государственный университет путей сообщения, г. Екатеринбург

В данной статье рассматривается проблема подмены человека нейросетевыми моделями и современные сервисы распознавания нейронных сетей. В статье делается вывод о том, что существующие методы распознавания могут потерять свою актуальность с развитием нейронных сетей, однако им на замену придут другие более совершенные методы.

Ключевые слова: информационная безопасность, дипфейк, нейронные сети, распознавание.

Нейросетевые модели уже проникли в нашу жизнь и с каждым днём они делают это все больше и больше. Однако кроме очевидных плюсов, таких как упрощение и ускорение рутинной и даже творческой работы, существует ряд очень значительных минусов. Один из них – подмена нейросетями человека. На данный момент нейросети уже умеют фотореалистично подделывать изображения, связанно писать текст, учитывая особенности конкретного человека/автора и достаточно хорошо подделывают голос. Всё это является огромной проблемой как для безопасности отдельно взятых людей, так и для отрасли информационной безопасности в целом.

Мир уже столкнулся с несколькими прецедентами подделки и кражи голоса человека. Такие подделки называют Deepfake (дипфейк).

В начале 2020 года управляющему банком в Гонконге позвонил человек, который говорил голосом директора одной из компаний-клиентов

банка. Звонивший сказал, что его фирма собирается совершить сделку на сумму в $35 млн, поэтому нужно, чтобы банк одобрил эту транзакцию. Координатором перевода директор представил юриста по имени Мартин Зелнер. Увидев на своей почте подтверждающее письмо от Зелнера, управляющий банком начал перевод денег. Позже оказалось, что директор компании никуда не звонил. Вместо него с управляющим банком говорили мошенники, использовавшие дипфейк-технологию для клонирования речи директора банка [1, 2].

Так же в 2023 году профессиональный российский диктор Алёна Андронова столкнулась с кражей собственного голоса. Несколько лет назад Алёна выполняла заказ для Тинькофф Банка по озвучанию большого количества материала для колл-центра банка. Однако все эти материалы попали в открытый доступ вследствие чего стали неправомерно использоваться в нейросетевой озвучке рекламы и других материалов из-за чего Алёна понесла не только материальные убытки в виде потенциальных клиентов, но и имиджевые, так как некоторые из рекламных роликов были сделаны для сомнительных сайтов и сервисов [3].

Чтобы избегать таких ситуаций нужно научиться определять сгенерированный контент. Для распознавания объектов работы нейросетей уже были созданы различные методы и технологии. В этой статье рассматриваются некоторые из них. Для идентификации изображений созданных нейросетями чаще всего используют эти несколько методов:

1. Проверка качества изображения. Этот тип детектора анализирует техническое качество изображения и сопутствующие параметры. Здесь учитывается то, что у картинок, созданных нейросетью, много общего: шумовые паттерны, несоответствия, сжатие.

2. Проверка содержания картинки. Детекторы этого типа на данный момент работают эффективнее всего, ведь такая модель позволяет быстро обнаружить аномалии (пальцы, глаза, текст на изображении и другие слабые места, которые нейросети так и не научились пока рисовать реалистично). Однако такой метод в скором будущем окажется неактуальным, ведь технологии развиваются и уже генератор изображений Midjourney 5 практически не создаёт визуальных аномалий.

3. Изучение внутренних составляющих изображения. Такой тип детектора проводит яркостный, спектральный и другие виды анализа изображений [4].

Именно последний метод наиболее эффективен в перспективе будущего. На нём, например, основан такой сервис по распознаванию генераций как «AI or Not» Этот сервис анализирует яркость и цвет на различных участках изображений, тем самым выявляя своеобразные «артефакты», невидимые человеческому глазу [5]. В то время как сервис «Illuminarty» основан на втором принципе. Он использует компьютерное зрение и ищет визуальные отклонения в изображении.

В распознавании музыки и голоса так же присутствуют различные как слышимые человеком, так и не слышимые особенности звука, которые допускает нейросеть при генерации. Для создания идеального искусственного голоса нужны исходные данные с одинаковым и главное низким уровнем шума. Отклонения в генерации могут возникать даже от малейших изменений в исходной записи. На это могут повлиять разное оборудование, разные помещения, даже разное настроение человека, чей голос использовался в исходных данных [6]. Именно на такие отличия и обращают внимание технологии распознавания Deepfake голосов.

Распознать сгенерированный текст сложнее всего. Для этого пока что используются только смысловые маркеры. Например, так называемый «синдром рыбки Дори» – потеря смысла текста у нейронной сети. Нейросеть как будто «забывает» то, о чём написала в предыдущем предложении из-за чего возможно появление двух перечащих друг другу предложений, расположенных по соседству.

Так же для упрощения процедуры распознавания принимаются меры со стороны разработчиков генеративных моделей и со стороны государств. Так Российским технологическим университетом было внесено предложение в Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации, ввести обязательную маркировку контента, созданного с помощью нейросетей, а также подготовить программу защиты критически важной инфраструктуры от возможных кибератак с использованием таких систем [7]. А прецеденты подобные тем, что были описаны в данной статье ранее помогут мировой судебной практике выстроить законодательную базу, обеспечивающую контроль над беспорядочным и неправомерным использованием в коммерческих или мошеннических целях.

Дальнейшее развитие нейросетевых моделей неизбежно, минусы будут исправляться и большинство вышеупомянутых способов распознавания будут терять свою эффективность. Поэтому люди будут находить другие способы и закономерности, по которым смогут определять, что является работой нейросетевой модели, а к чему приложил свою руку такой же человек, как и они сами. И помогать находить эти особенности человечеству будут так же нейронные сети, что будут созданы для этого. И немаловажную роль в этом противостоянии двух лагерей сыграют простые правила:

1. Всегда необходимо проверять информацию. Ищите первоисточник или другие упоминания. В этом может помочь встроенная функция в поисковик Google – About this image (Об этом изображении), которая покажет: когда конкретное изображение и подобные были впервые проиндексированы Google; где оно было опубликовано впервые; где ещё встречается в интернете.

2. Следует использовать сервисы по распознаванию, если не уверены в том, кем были созданы изображение/текст/голос.

3. Придерживайтесь стандартных правил личной безопасности в сети интернет, чтобы ваши данные не были украдены и использованы злоумышленниками для подмены вас нейронными сетями.

Библиографический список

1. Мошенники украли $35 млн у банка с помощью дипфейка голоса его главы // Habr URL: https://habr.com/ru/news/583590/ (дата обращения: 23.10.2023).

2. Мошенники украли $35 миллионов с помощью технологии синтеза речи // Dzen URL: https://dzen.ru/a/Yav-hza2Dhokssou (дата обращения: 23.10.2023).

3. Из голоса банка – в п0рно // Pikabu URL: https://pikabu.ru/story/iz_golosa_banka__v_p0rno_10607302 (дата обращения: 23.10.2023).

4. Как понять, что картинку сделала нейросеть – 5 сервисов // TexTerra URL: https://texterra.ru/blog/proverit-neyroset-besplatno-detektor-po-opredeleniyu-kartinok.html (дата обращения: 24.10.2023).

5. Стартап Optic запустил сервис AI or Not для распознавания созданных ИИ изображений // Habr URL: https://habr.com/ru/news/728756/ (дата обращения: 24.10.2023).

6. Разработчики нейросетей об отрасли // YouTube URL: https://www.youtube.com/watch?v=XYbqey-bdlI (дата обращения: 24.10.2023).

7. Минцифры предложили ввести маркировку контента, созданного с помощью нейросетей // Тасс URL: https://tass.ru/ekonomika/17746919 (дата обращения: 24.10.2023).

УДК 004.056.53

ХАКЕРЫ. МЕЖДУНАРОДНЫЕ АСПЕКТЫ

К.Н. Гуральский, С.В. Мухачев
Научный руководитель: канд. физ.-мат. наук, доц. С.В. Мухачев
Уральский государственный университет путей сообщения, г. Екатеринбург

Рассматриваются виды кибератак и их влияние на международный аспект. Приводятся примеры хакерских атак, история их появления и последствия для ситуации в мире. Анализируется рост хакерских атак за последние годы и возможные причины, которые могли на это повлиять.

Ключевые слова: атака отказ в обслуживании (DDoS), вирус, кибератака, кибербезопасность, кибервойна, киберпреступность, фишинг, хакеры.

Целью статьи является рассмотрение видов хакерских атак, их влияние на международный аспект, сравнение статистик количества хакерских атак, анализ причин увеличения количества атак.

В связи с распространением компьютеров, многие задачи, которые человек выполнял вручную, перестали быть энергозатратными и упростили работу людям. С появлением первых компьютеров жизнь человека значительно облегчилась, компьютеры позволили автоматизировать многие процессы, что позволило значительно уменьшить объем работы, которую должен выполнять человек лично. Онлайн-банкинг, биржи, социальные сети, доступ к нужной информации из любого места в любое время – все это лишь небольшая часть возможностей, которые нам доступны. Однако, несмотря на все перечисленные плюсы и возможности, с развитием технологий появилась и новая угроза, которая заключается в неправомерном доступе к вашей информации, ее дальнейшем распространении, использованию. Другими словами, хакерские атаки.

По сравнению с 2021 годом, по статистике (рис. 1), в 2022 количество инцидентов возросло на 20,8 процентов [1], что подчеркивает актуальность данной темы, ее дальнейшее рассмотрение и поиск оптимальных решений.

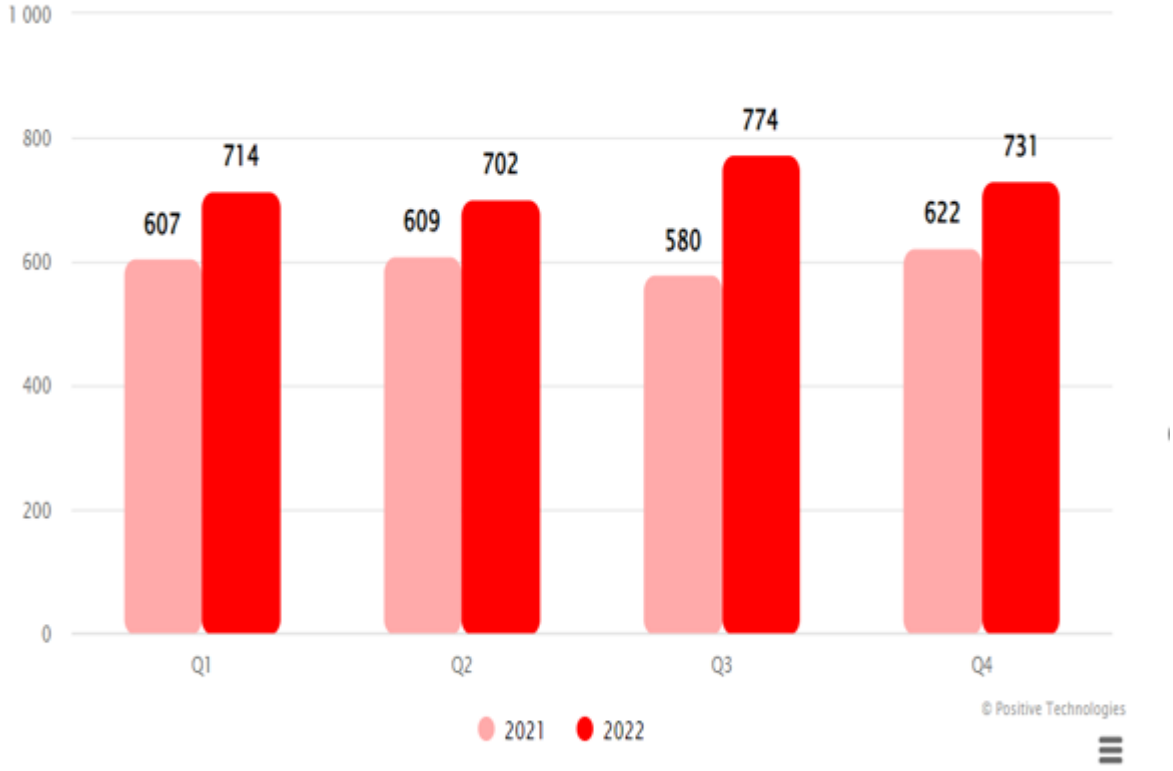


Рис. 1. Статистика количества кибератак за период 2021 – 2022

Кто же такие хакеры?

Хакер [2] – это профессионал, программист, способный найти уязвимости в системе и использования их для несанкционированного доступа в саму систему.

Данный термин не дает четкого представления о том, являются ли хакеры злоумышленниками, поэтому хакеры классифицируются на два типа:

1. Тот, кто ищет уязвимости системы с целью указать на них, чтобы в дальнейшем ее исправили.

2. Тот, кто используют полученную информацию в своих целях – соответственно.

В данной статье будет рассматриваться именно второй тип хакеров и угрозы с их стороны.

Хакерские атаки или кибератаки – это попытки незаконного доступа, манипуляции, внедрения или уничтожения компьютерных систем, сетей или данных. Хакеры или злоумышленники, могут использовать различные методы и техники, чтобы нарушить конфиденциальность, целостность или доступность информации. Цели хакерских атак могут варьироваться от кражи личной информации и финансовых данных до разрушения критической информационной инфраструктуры или шпионажа.

Типы хакерских атак [3]:

Фишинг: это атака, при которой злоумышленники пытаются получить конфиденциальные данные, такие как пароли и финансовая информация, путем маскировки под доверенное лицо или организацию.

DDoS-атака [4] (атака отказ в обслуживании): это атака, при которой злоумышленники используют множество устройств для отправки большого количества трафика на целевой сервер, что приводит к его перегрузке и недоступности для законных пользователей.

SQL-инъекция: это атака, при которой злоумышленники внедряют злонамеренный SQL-код в строку запроса к базе данных, что может привести к несанкционированному доступу к данным.

Вредоносные программы: это программное обеспечение, спроектированное для внедрения или нанесения ущерба компьютерным системам без согласия владельца. Это включает в себя вирусы, черви, троянские кони и другие вредоносные программы.

Man-in-the-middle атака (MITM): это атака, при которой злоумышленник вставляется между двумя коммуницирующими сторонами и перехватывает коммуникацию между ними.

Сетевое подслушивание: это атака, при которой злоумышленник проводит мониторинг и записывает трафик в сети с целью перехвата конфиденциальной информации, такой как логины и пароли.

Вредоносные вложения по электронной почте: это метод, при котором злоумышленники отправляют по электронной почте вредоносные вложения, которые при открытии могут заразить устройство вирусами или другими вредоносными программами.

Первые кибератаки, история появления вирусов.

Первый записанный в истории компьютерный вирус, по официальным данным, назывался «Creeper» [5] (полное название – «The Creeper Worm») и появился в начале 1970-х годов. Этот вирус был разработан программистом Робертом Томасом для операционной системы TENEX, предшествен-

ницы ARPANET, предшественницы интернета. «Creeper» был в состоянии перемещаться по сети, заражать компьютеры и выводить сообщение на экран: «I'm the creeper, catch me if you can!» («Я крипер, поймай меня, если сможешь!»). Первым антивирусом, созданным для противодействия этому вирусу, стал «Reaper» («Жнец»), который удалял «Creeper» с зараженных компьютеров.

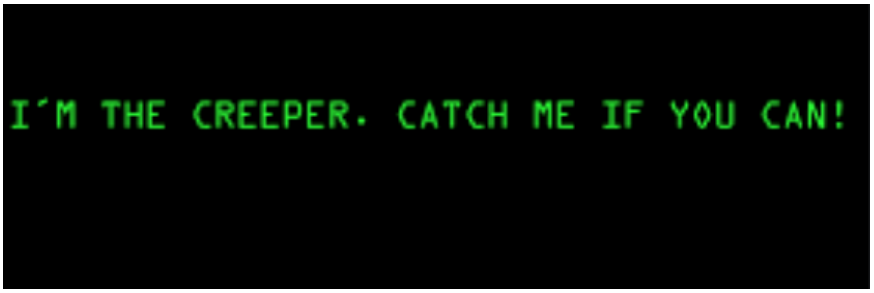


Рис. 2. Результат работы вируса Creeper

Стоит отметить, что понятие компьютерных вирусов тогда было новым, и термин «вирус» в контексте компьютеров еще не был популярен.

Первые упоминания в разных странах.

1. Америка

Когда речь заходит о вирусах, невольно вызывается в памяти фраза «Червь Морриса» [6]. Этот первоначальный вирус, получивший имя своего создателя, Роберта Морриса, заложил основу для всех подобных программ, которые сегодня значительно отличаются от своего предка. Червь Морриса впервые появился 2 ноября 1988 года и успел заразить около 65 000 компьютеров, причинив общий ущерб в размере 97 миллионов долларов. Согласно официальным источникам, его создатель мог быть осужден на пять лет тюрьмы и получить штраф в размере 250 000 долларов, однако он избежал заключения, получив три года условно, штраф размером в 10.000 долларов и 400 часов общественных работ.

2. Россия

Первый зарегистрированный компьютерный вирус в России, известный как «Каскад» (Cascade) [7], был обнаружен в 1986 году. Эта программная зараза была создана студентами Московского физико-технического института (МФТИ) и Ленинградского физико-технического института (ЛФТИ) и отслеживала основную причину своего существования – создание самореплицирующихся программ, способных распространяться через дискеты. Примечательно, что «Каскад» не наносил непосредственного вреда компьютеру, на котором он активировался. Вместо этого он отображал на экране пользовательское сообщение, в котором авторы выражали свое «творческое» видение создания вирусов и предоставляли контактные данные, чтобы связаться с ними. Вирус «Каскад» был первым широко известным примером компьютерной инфекции в России и неактивной агрессией, что привлекло внимание к вопросам компьютерной безопасности в стране.

Впоследствии Россия стала ведущим участником в области кибербезопасности, что подчеркивает серьезность проблемы в данной сфере.

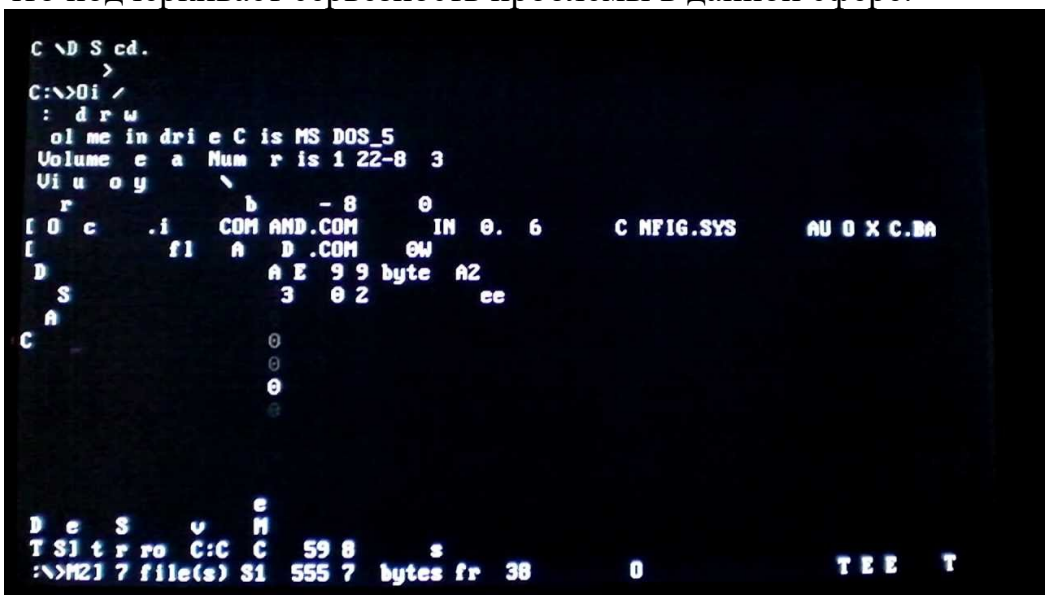


Рис. 3. Результат работы вируса Cascade

3. Китай

Среди первых компьютерных вирусов, которые стали широко известными в Китае, особое место занимает вирус Xiang Red Army, который получил также известность под названием «Fuxi Virus» [8]. Эта вредоносная программа была обнаружена в Китае в 1990 году и распространялась через зараженные дискеты. Xiang Red Army является одним из ранних примеров компьютерных вирусов, которые эффективно привлекли внимание к проблемам компьютерной безопасности в Китае и за его пределами.

Можно заметить, что первые вирусы начали появляться почти в одно и тоже время в разных странах, примерно в то время, когда начали распространяться компьютеры и открывался доступ к сети, что и стало их средой распространения.

Влияние на международный аспект.

В современное время хакеры могут оказывать значительное влияние на международный аспект в различных сферах:

1. Кибербезопасность и кибервойны: хакеры могут направлять свои атаки на государственные системы, армии и крупные корпорации. Это может вызвать международные конфликты и даже кибервойны между странами.

2. Экономические последствия: хакерские атаки на финансовые учреждения или крупные компании могут вызвать экономические потери в различных странах. Это может повлиять на мировую экономику и торговлю.

3. Шпионаж и политическое вмешательство: хакеры могут собирать разведывательную информацию для государственных интересов или вме-

шиваться в политические процессы других стран, например, взламывая электронные почты политиков или устраивая кампании дезинформации.

4. Инфраструктурные атаки: хакеры могут нацеливать свои атаки на критическую инфраструктуру, такую как энергетические системы, транспортные сети и коммуникационные системы. Это создает риски для безопасности многих стран.

5. Киберпреступность и кибермошенничество: хакеры также могут участвовать в мошеннических схемах, воровстве личных данных и финансовых преступлениях, которые пересекают национальные границы и затрагивают множество стран.

6. Оборона и профилактика: многие страны разрабатывают оборонительные меры и политику для защиты от хакерских атак. Сотрудничество между странами в сфере кибербезопасности становится все более важным для предотвращения масштабных атак.

7. Социокультурное влияние: хакерская культура и идеи также оказывают влияние на молодежь и общество, формируя отношение к технологиям, приватности и безопасности в различных странах.

8. Законодательство и международное сотрудничество: страны разрабатывают новое законодательство и международные соглашения для борьбы с киберпреступностью и защиты от хакерских атак, что влияет на международные отношения и правопорядок.

Анализ атак в России и других странах.

По статистике, за последние годы, количество кибератак безостановочно росло, наибольший прирост произошел в момент начала пандемии, и эта статистика касается не только России, но и остальных стран. Например, по сравнению с 2019 годом, в 2020 году, именно в этот год началась пандемия, количество кибератак увеличилось на 51%, а по сравнению с 2021, количество атак в 2022 году увеличилось на 20,8% [1].

Таким образом, количество кибератак сильно зависит от ситуации в мире, наибольший прирост количества атак за последние годы, как видно из статистики, наблюдается во время начала пандемии и специальной военной операции. Несмотря на то, что каждый год придумывают новые способы борьбы с атаками подобного типа, их прогресс не стоит на месте и создатели вирусов находят новые лазейки в системе. Хакерская деятельность имеет огромные последствия, поэтому укрепление международного сотрудничества и разработка эффективных кибербезопасных политик становятся ключевыми целями для предотвращения этих угроз. Важно помнить, что угрозы в области кибербезопасности могут изменяться в зависимости от событий в мире. Поэтому компании и организации должны постоянно приспосабливаться к новым угрозам и разрабатывать гибкие стратегии по защите своих данных. Чтобы защитить себя от хакерских атак, компании и организации должны уделять большее внимание кибербезопасности. Это включает в себя регулярное обновление программного

обеспечения, обучение сотрудников в вопросах безопасности и мониторинг сетевой активности для выявления подозрительных событий. Также, в связи с тем, что кибератакам подвергаются все страны, международное сотрудничество в сфере кибербезопасности становится все более важным. Страны и организации должны совместно бороться с киберугрозами и разрабатывать механизмы для выявления хакеров.

Библиографический список

1. Статистика кибератак // URL: https://www.ptsecurity.com/ru-ru/ (дата обращения: 19.09.2023).

2. Кто такие хакеры? // URL: https://skillbox.ru/media/code/khakery-kto-oni-kakie-byvayut-i-pri-chyem-tut-shlyapy/ (дата обращения: 19.09.2023).

3. Виды хакерских атак. // URL: https://www.securitylab.ru/analytics/534885.php (дата обращения: 06.10.2023).

4. Богер А.М., Соколов А.Н., Морозов И.А. Атаки на трафик обмена данными и между программируемыми логическими контроллерами SIMATIC 1510 и SIMATIC 1512 // Вестник УрФО. № 4(46). 2022. С. 88–96.

5.Creeper // URL: https://www.vesti.ru/article/2059547 (дата обращения: 22.09.2023).

6. Червь Морриса // URL: https://fb.ru/article/302225/cherv-morrisa-istoriya-poyavleniya-virusa-printsip-deystviya-i-interesnyie-faktyi (дата обращения: 23.09.2023).

7. Вирус Cascade // Энциклопедия Касперского // URL: https://encyclopedia.kaspersky.ru/knowledge/cascade-virus/ (дата обращения: 06.10.2023)

8. FuxiVirus // URL: https://virus.fandom.com/ru/wiki/ (дата обращения: 07.10.2023).

УДК 004.056

СОВРЕМЕННАЯ ПРОБЛЕМАТИКА ЗАЩИТЫ ИНФОРМАЦИИ ОТ УТЕЧКИ В РЕЗУЛЬТАТЕ ПРИМЕНЕНИЯ МЕТОДОВ СОЦИАЛЬНОЙ ИНЖЕНЕРИИ

Н.А. Михайлов, С.В. Мухачев
Научный руководитель: канд. физ.-мат. наук, доц. С.В. Мухачев
Уральский государственный университет путей сообщения, г. Екатеринбург

В эпоху развитых цифровых технологий информация является важным активом в новом современном мире. Она играет ключевую роль в бизнес-активах разных компаний, а также в экономике стран мира. Поэтому актуальность обеспечения информационной безопасности несомненна. Технологии защиты информации стремительно развиваются, однако нейтрализовать все ин-

формационные угрозы не удается. Человек становится слабым компонентом в системе информационной безопасности, что приводит к худшим сценариям развития событий. Цель статьи – изучить проблему защиты информации от утечки в результате применения инструментов социальной инженерии. Задачи статьи – охарактеризовать проблематику социальной инженерии, проанализировать особенности методов социальной инженерии, статистические данные, и предложить меры по противодействию.

Ключевые слова: защита информации, информационная безопасность, социальная инженерия, утечка информации.

Со времен античной эпохи люди разрабатывали методы защиты важной информации. Так, например, был придуман знаменитый шифр Цезаря, который используется и в наше время. Император использовал его для переписки со своими военачальниками, в целях сохранения важной информации. В течение всей истории развивались методы и технологии защиты информации, и в нынешнее время они имеют сильную защиту. Однако, несмотря на мощные криптографические алгоритмы и технические, физические средства защиты, безопасность информации остается под угрозой [1, 2].

На любом предприятии работают люди, поэтому всегда присутствует так называемый «человеческий фактор». Это справедливо и для процессов, связанных с информационной безопасностью. Любые действия человека, связанные с нарушением режима безопасности, можно разделить на две большие категории: умышленные и неумышленные действия. К неумышленным относятся: утрата носителей информации, уничтожение или искажение информации по неосторожности. Человек не осознает, что его действия ведут к нарушению информационной безопасности. К умышленным действиям относятся кражи информации, модификация информации, либо ее уничтожение (диверсии). Это крайний случай, и с ним приходиться бороться постфактум, привлекая сотрудников внутренних дел [3].

Социальная инженерия – это способ получение несанкционированного доступа к защищаемой информации и к системам обработки информации без использование технических средств, с помощью психологического манипулирования людьми. Данный способ ориентирован на психологические уязвимости и на недостаточную осведомленность в политике безопасности информации. Как показывает практика, способ является очень эффективным. Самый простой пример – преступник звонит потенциальной жертве, с целью узнать у нее важную информацию (номер банковской карты, пароли), при этом играет на чувствах собеседника, обманывает или шантажирует его [4]. Само явление появилось в конце 70-х годов прошлого века. Обычные телефонные шутники разыгрывали простых граждан, но кто-то додумался, что таким образом можно заполучить конфиденциальную ин-

формацию. Такие люди получили название «социальные инженеры». Они используют различные психологические приемы и добывают информацию.

Существуют множество сценариев атак с использованием социальной инженерии. Остановимся на основных и распространенных.

Фишинг – способ получение информации путем рассылки писем на e-mail и специальных web-страниц. Преступник отправляет письмо на почту потенциальной жертве. Это письмо замаскировано под вид официальной рассылки от банка или с место работы, которое подается как для проверки важной конфиденциальной информации. Обычно в этом письме находится ссылка на стороннюю web-страницу. Таким образом человека передает свои данные мошеннику. Такой метод направлен на использование невнимательности людей.

Еще одни метод, связанный с отправкой письма на e-mail, получил название троянский конь. Суть данной атаки заключается в том, чтобы жертва открыла рассылку и скачала файл, в котором замаскирован вирус. Чтобы потенциальная жертва выполнила необходимые действия, используются методы социальной инженерии. Например, рассылка подается как обновление программного обеспечения, антивируса, или способ быстро заработать.

«Quid pro quo» или «услуга за услугу». Злоумышленник играет роль персонала технической поддержки и предоставляет свои услуги по устранению неполадок в системе. Главная задача преступника – внушить жертве, что проблема серьезная, и требуется немедленного вмешательства, но при этом неисправности в системе нет. Завоевав доверие, мошенник просит конфиденциальную информацию жертвы, для возможности устранения проблемы.

Претекстинг – прием, когда преступник действует по отработанному заранее сценарию, добиваясь доверия потенциальной жертвы, и получает от нее конфиденциальную информацию. Социальные инженеры могут представиться как: сотрудники банка, техподдержка, дальний родственник.

Также существует обратная социальная инженерия. Суть данного метода заключается в том, чтобы жертва сама обратилась к злоумышленнику за помощью, и в процессе общения раскрыла информацию о себе. Социальные инженеры могут рекламировать услуги, например, компьютерных специалистов. Они приезжают по вызову к жертве, выполняют заказ, и параллельно получают информацию о жертве путем общения с ней. Еще вариант – при возникновении сбоя в программе или в системе, вызывают специалиста, который оказывается злоумышленником. Он устраняет проблему, а также производит действия для реализации взлома системы. При возникновении взлома социальный инженер остается вне подозрения [5].

В настоящие время социальная инженерия остается крайне актуальной проблемой. По данным исследований Российской компании «Positive

Technologies» за 2021 и 2022 годы (рис. 1), доля атак с использованием методов социальной инженерии увеличилась с 89% до 96% при атаках на частных лиц, при этом в атаках на организации доля снизилась с 52% до 43%. Таким образом, можно судить об эффективности методов социальной инженерии при атаке на различных жертв.

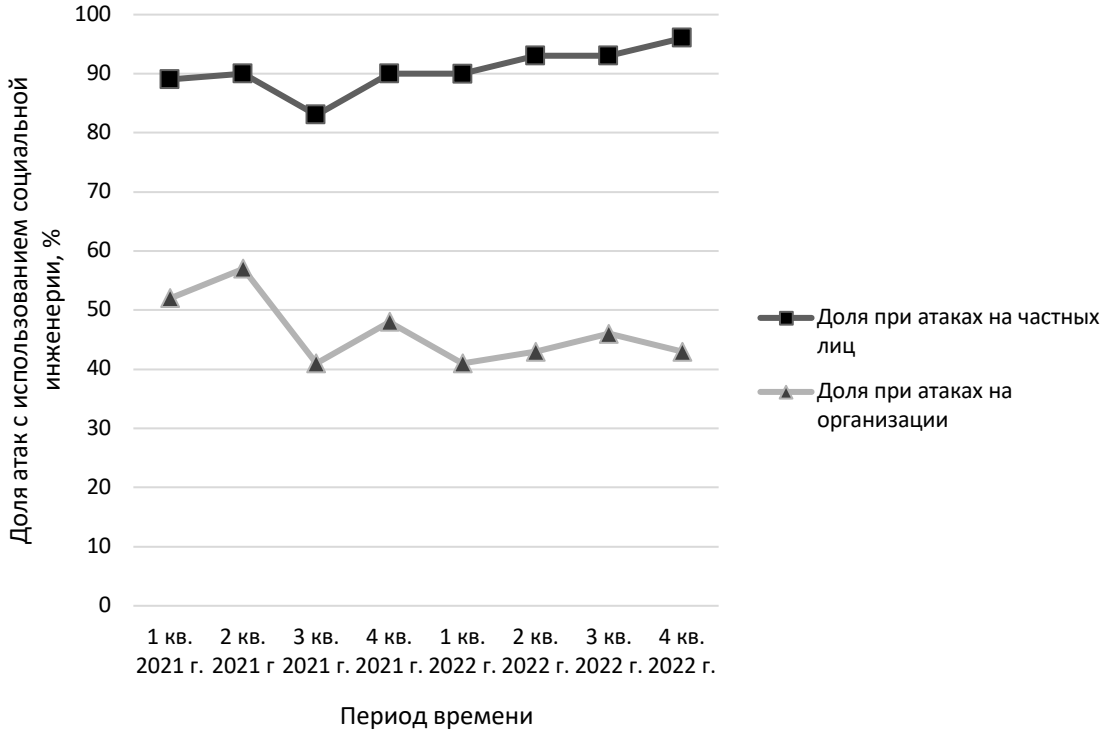


Рис. 1. Доля атак с использованием социальной инженерии

В 2022 году в 16% успешных атак, направленных на организации, преступникам удалось получить доступ к целевым системам и ресурсам с помощью компрометации учетных данных. Это могло быть достигнуто как посредством подбора паролей, так и с помощью учетных данных, скомпрометированных в результате утечек [6]. Из приведенных данных следует, что доля таких атаки на частных лиц увеличивалась в каждом квартале 2022 года. При этом доля инцидентов на организации несущественно менялась на протяжение всего 2022 года. Можно сделать вывод, что частные лица более уязвимы к атакам с использованием социальной инженерии.

Используемые злоумышленниками каналы социальной инженерии представлены на рис. 2. Из приведенных данных следует, что наиболее эффективный канал для атаки на предприятия – электронная почта. Почти в каждой успешной атаке на предприятия с использованием социальной инженерии преступники пользовали вредоносные электронные письма. Для атаки на частных лиц наиболее эффективен канал с использованием фишинговых сайтов, а также с помощью мессенджеров и СМС-сообщений и социальных сетей.

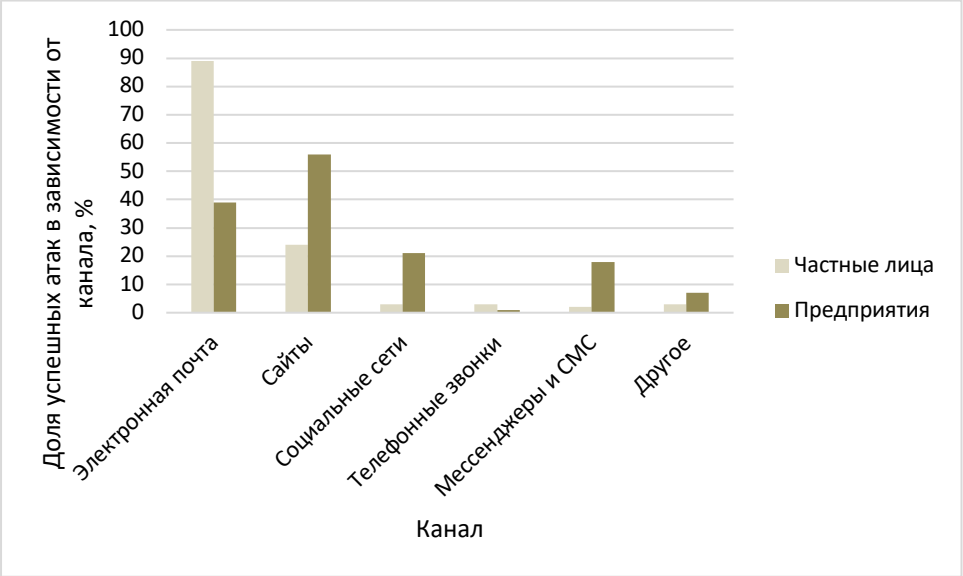


Рис. 2. Используемые злоумышленниками каналы социальной инженерии

Можно сделать вывод, что наиболее успешны атаки на электронную почту организаций. Они имеют почти 90 % шансов на успех. Частные лица уязвимы на фишинговых сайтах. Поэтому фишинговые сайты и рассылки на почту вредоносных писем для социальных инженеров в целом являются наиболее подходящими методами.

В 2022 году активно распространялась модель «phishing as a service»: злоумышленники используют в атаках готовые фишинговые комплекты, при этом в некоторых из инцидентов использовались инструменты для обхода многофакторной аутентификации. Например, в конце года был отмечен всплеск атак типа «MFA Fatigue»: злоумышленники выполняли множественные попытки входа в аккаунт, используя украденные учетные данные, вызывая бесконечный поток пуш-уведомлений, отправляемых на мобильное устройство владельца учетной записи. В итоге часть пользователей подтверждали вход на ресурс, чтобы остановить поток сообщений.

Можно выделить ряд мероприятий по защите по противодействию методам социальной инженерии:

– ознакомление с политикой конфиденциальности сотрудников предприятия. Они должны быть проинструктированы об организации работы с информацией ограниченного доступа. Только после этого их следует допускать к работе;

– для различных ресурсов в сети Интернет следует использовать разные пароли;

– необходимо сохранять осторожность при вводе личных данных на сомнительных сайтах и не переходить по подозрительным ссылкам;

– обязательно должна использоваться антивирусная программа;

– следует использовать только двухэтапную аутентификацию и ограничить лимит ввода данных. Двухэтапная аутентификация усилит условия

аутентификации, а лимит ввода заблокирует возможность злоумышленнику попытаться войти в систему путем паролей и кодов. Кроме того, при наличии большого количества уведомлений о попытках входа, пользователь обязан сообщить об инциденте сотруднику, отвечающему за информационную безопасность, поскольку это говорит о вероятной компрометации учетных данных;

– особо необходимо соблюдать осторожность в социальных сетях. Не стоит сообщать личную информацию собеседнику, и переводить деньги, если сомневаетесь в достоверности личности;

– полезно разъяснить пользователям способы применения злоумышленниками методов социальной инженерии. Это позволит подготовить их к возможным угрозам со стороны мошенников.

Таким образом, методы социальной инженерии активно применяются злоумышленниками с целью преодоления систем защиты информации. Человек, как элемент такой системы, может подвергнуться воздействию со стороны злоумышленника. При этом могут использоваться такие свойства, присущие человеку, как невнимательность, любопытство, алчность и так далее. Человеческий фактор очень часто играет ключевую роль в работе с критически важной информацией.

Обнаруживаются различия при атаках на частных лиц и предприятия. Доля атак с использованием методов социальной инженерии на частных лиц существенно выше, чем при атаках на предприятия. Эффективность каналов, используемых для атак, также различна для атак на частных лиц и на предприятия. Однако, в любом случае наиболее эффективны для атак с использованием социальной инженерии e-mail-письма и фишинговые сайты.

Несмотря на развитие технологий, степень защиты информации в первую очередь зависит от людей, которые ею владеют. Поэтому при организации защиты информации, важное значение имеют мероприятия по противодействию атак с использованием методов социальной инженерии.

Библиографический список

1. Секреты и ложь. Безопасность данных в цифровом мире / Б. Шнайер. – СПб.: Питер, 2003. – 368 с.

2. Халилаева Э.И. Система противодействия методам социальной инженерии в области информационной безопасности / Э.И. Халилаева, М.А. Маслова, В.М. Герасимов // Вестник УрФО. Безопасность в информационной сфере. – 2023. – № 2(48). – С. 54–61.

3. Краткое введение в социальную инженерию // URL: https://habr.com/ru/articles/83415/ (дата обращения 21.10.23).

4. Сивчук Е.С. Социальная инженерия как способ мошенничества / Е.С. Сивчук // Молодой ученый. – 2020. – № 41(331). – С. 128–130.

5. Фатахова Д.Р. Мошенничество в сети Интернет / Д.Р. Фатахова // Молодой ученый. – 2020. – № 49(339). – С. 341–344.

6. Актуальные киберугрозы: итоги 2022 года // URL: https://www.ptsecurity.com/ru-ru/research/analytics/cybersecurity-threatscape-2022/#id6 (дата обращения: 25.10.2023).

УДК 004.056

ИССЛЕДОВАНИЕ ВОПРОСОВ БЕЗОПАСНОСТИ ПРИМЕНЕНИЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Г. Наскидашвили, Д.А. Корженевский
Научный руководитель: асс. Д.А. Корженевский
Уральский государственный университет путей сообщения, г. Екатеринбург

В статье обозреваются вопросы применения искусственного интеллекта по отношению к субъектам персональных данных, анализ результатов опроса, предложения по созданию хранилища информации, сгенерированной искусственным интеллектом.

Ключевые слова: искусственный интеллект, подмена личности, поддельный контент, персональные данные.

С появлением возможности использования искусственного интеллекта (ИИ) в открытом доступе в последние годы началась популяризация данной тематики.

В отчете Министерства Цифрового развития и массовых коммуникаций, направленном Президенту Российской Федерации, были рассмотрены вопросы по согласованию и созданию обезличенных датацентров для развития отечественных ИИ, по защите прав граждан при обработке больших данных и по применению технологий ИИ. Предлагался прямой запрет на обработку больших данных в случае, если это может привести к риску причинения вреда здоровью, безопасности и имуществу граждан. Министерство рекомендовало создание государственных информационных систем для загрузки и обработки обезличенных датасетов, которые будут предоставляться авторизованным разработчикам. Загруженные датасеты нельзя будет выгружать из системы [1].

Помимо имеющихся вопросов, которые рассматриваются на государственном уровне, существует постоянно нарастающий интерес общества к технологиям ИИ. Растущая популярность все больше привлекает пользователей, разработчиков, крупные компании, а также злоумышленников. Персональные данные (ПДн) жертв мошеннических операций подвергаются: атакам, сбору, обработке, использованию в фишинге, вымогательстве,

подмене личности и так далее. Суть данных атак заключается в кратковременном звонке потенциальной жертве и сборе биометрических данных для дальнейшей подмены личности.

Представленная проблема актуальна. Для понимания проблемы был проведен опрос в целях исследования информированности респондентов в данной теме и их способности различать ИИ от человека, а также осознавать опасности возможных рисков. В исследовании были использованы методы теоретического анализа, опроса, методы обработки и анализа информации.

Исследование проводилось с участием студентов УрГУПС с 1–5 курса. Опрошено 37 респондентов. Среди них 27 респондентов мужского пола и 10 женского.

В одном из вопросов респондентам предстояло найти среди четырех новостей одну, сгенерированную ИИ, и указать её. Правильным вариантом ответа являлась «Новость №3». Варианты ответов на данный вопрос продемонстрированы на рис. 1.

1. Ученые нашли способ возвращения подвижности после инсульта	2. Для большинства болезней пока нет генной терапии, заявила эксперт
3. Ученые открыли новый элемент в периодической таблице - Эксонитрий (Exonitrium)	4. Инженеры разработали прорывную «человеческую кожу робота»

Рис. 1. Варианты новостей

Ответы респондентов приведены в табл. 1.

Таблица 1

Результаты опроса респондентов

Вопросы	Варианты ответов	Возрастные диапазоны			
		18–20 лет	21–23 года	23–25 лет	Более 25 лет
Ваш курс обучения?	1 курс	5	1	1	–
	2 курс	7	–	–	–
	3 курс	14	1	–	–
	4 курс	–	–	3	–
	5 курс и старше	–	1	3	1
Знакомы ли вы с ИИ?	Да / Нет	24 / 2	3 / –	5 / 2	– / 1

Окончание табл. 1

Вопросы	Варианты ответов	Возрастные диапазоны			
		18–20 лет	21–23 года	23–25 лет	Более 25 лет
Как часто вы пользуетесь ИИ?	Больше 1 раза в день	1	1	–	–
	1 раз в день	–	–	1	–
	Больше 1 раза в неделю	1	1	–	–
	1 раз в неделю	16	–	–	–
	Больше 1 раза в месяц	5	–	–	–
	1 раз в месяц	–	1	2	1
	Не пользуюсь	4	–	4	–
Можете ли вы отличить новость, сгенерированную ИИ от настоящей?	Да / Нет	6 / 20	2 / 1	5 / 2	– / 1
Найдите ложную новость	Новость 1	4	–	–	1
	Новость 2	8	1	1	–
	Новость 3	8	2	2	–
	Новость 4	6	–	4	–
Знаете ли вы о методах сбора данных для подмены личности?	Да / Нет	17 / 9	2 / 1	1 / 6	– / 1
Опасаетесь ли вы, что ваши биометрические данные будут использованы в подмене личности?	Да / Нет	13 / 13	3 / –	5 / 2	– / 1
Знаете ли вы о методах борьбы против неправомерной обработки и использования ваших персональных данных?	Да / Нет	12 / 14	2 / 1	1 / 6	1 / –

Проведен анализ полученных данных. Наиболее распространенная возрастная категория респондентов 18–20 лет. Знают про ИИ 89% и среди них часто используют ИИ 63%. Опасаются подмены личности 57%. Знают о том, что делать против неправомерной обработки персональных данных с

использованием ИИ 43%. Не боятся подмены личности 16 респондентов, среди них о методах защиты знают только половина опрошенных. Уверенность в том, что могут отличить новость, сгенерированную ИИ, от настоящей, имеют 35% респондентов. Из них правильно определивших сгенерированную новость – всего 46%.

На основании анализа можно сделать выводы: большинство респондентов осведомлены о существовании ИИ и опасаются того, что их биометрические данные будут использованы злоумышленниками. Большинство респондентов не знают, как бороться с неправомерной обработкой ПДн с использованием технологий ИИ и подменой личности. Большее количество респондентов не уверены, что смогут отличить реальную новость от сгенерированной ИИ. Среди тех, кто уверен в возможности определения сгенерированной новости, меньше половины смогли отличить сгенерированную новость.

На основании вышеизложенного становится очевидна проблема как осведомленности граждан о своих правах по защите персональных данных от неправомерного использования с применением ИИ, так и проблема определения подлинности контента. В мире звучат предложения о маркировании материалов, сгенерированными ИИ, чтобы человек мог определить источник контента. Однако данную метку можно удалить, и, следовательно, вновь невозможно будет определить авторство.

Предлагается для решения вышеописанного вопроса создание на территории Российской Федерации единого хранилища информации, сгенерированной искусственным интеллектом. Начиная, к примеру, с некоторого порога генерируемых сообщений или иного контента, направлять эту информацию в хранилище. Текстовые материалы можно хранить в явном виде, аудиовизуальные материалы преобразовывать с использованием хэш-функций для сокращения хранимых объемов информации. Таким образом, если субъект усомнился в источнике материала, он может составить запрос в систему и получить ответ о нахождении таковой информации в хранилище и/или процент совпадения.

Однако не стоит рассматривать ИИ только как угрозу обществу и инструмент для злоумышленников. Существуют компании, использующие ИИ для улучшения качества защиты от вредоносного программного обеспечения (ВПО). Например, защитник Windows от Microsoft использует методы машинного обучения для создания адаптивной системы защиты от ВПО [2].

На данный момент специалисты «Тинькофф» разработали ИИ, способный изучать звонки мошенников, играть роль невинной жертвы [3]. Таким образом, количество звонков, поступающих субъектам, снижается, следовательно, вероятность успешных мошеннических действий снижается.

В случае же, если жертва попала в ситуацию неправомерного сбора, обработки и использования ПДн с применением ИИ, то государство пред-

лагает пострадавшим следующие меры для защиты своих прав: обратиться в Роскомнадзор и в суд. Злоумышленник привлекается к ответственности согласно статье 159 УК РФ «Мошенничество» [4].

В данной работе рассмотрены предложения Министерства Цифрового развития и массовых коммуникаций и растущий интерес граждан к ИИ. Проведен опрос респондентов, проанализированы результаты, внесены рекомендации. Предложено создание хранилища сгенерированной информации ИИ.

Библиографический список

1. Обозначены новые подходы к обезличиванию больших данных // URL: https://digital.gov.ru/ru/events/45880/?utm_referrer=https%3a%2f%2fyandex.ru%2f (дата обращения: 20.10.2023).

2. Microsoft добавляет в Defender защиту от программ-вымогателей на основе искусственного интеллекта // overclockers URL: https://overclockers.ru/blog /anykey911/show/59275/microsoft-dobavlyaet-v-defender-zaschitu-ot-programm-vymogatelej-na-osnove-iskusstvennogo-intellekta (дата обращения: 20.10.2023).

3. Добавили новый навык телефонному секретарю Олегу // tinkoff URL: https://www.tinkoff.ru/finance/blog/robots/ (дата обращения: 20.10.2023).

4. УК РФ Статья 159. Мошенничество \ КонсультантПлюс // consultant URL: https://www.consultant.ru/document/cons_doc_LAW_10699/8012ecdf64b7c9cfd62e90d7f55f9b5b7b72b755/ (дата обращения: 20.10.2023).

УДК 004.056

АНАЛИЗ УГРОЗЫ ОБРАБОТКИ БИОМЕТРИЧЕСКИХ ПЕРСОНАЛЬНЫХ ДАННЫХ ПРИ ОБУЧЕНИИ НЕЙРОСЕТЕЙ

М.А. Середа, М.Н. Киченко, Н.В. Ганженко
Научный руководитель: ст. преподаватель Н.В. Ганженко
Уральский государственный университет путей сообщения
г. Екатеринбург

Статья посвящена потенциальным угрозам при нерегулируемом использовании биометрических персональных данных нейросетями. Приведены примеры инцидентов, произошедшие из-за нерегулируемого использования биометрических персональных данных пользователей. Предложены рекомендации по правовому регулированию использования биометрии при работе с нейросетями.

Ключевые слова: биометрические персональные данные, законодательство, нейросеть, регулирование.

В настоящее время нейросети широко используются в различных сферах нашей жизни, обеспечивая массу преимуществ: от экономии времени до точности анализа. Однако с постоянным развитием и применением новых технологий появляются и новые угрозы. В частности, опасность использования нейросетями биометрических персональных данных становится актуальной темой для обсуждения. Необходимо проанализировать потенциальные риски, связанные с таким использованием, включая угрозы конфиденциальности данных и угрозы нарушения прав человека.

Нейросети искусственного интеллекта (ИИ) продолжают свое быстрое развитие и уже оказывают значительное влияние на различные аспекты нашей жизни, включая развлечения. Однако, наряду с позитивными последствиями, развитие нейросетей несет и некоторые негативные риски. Одним из наиболее очевидных негативных последствий является распространение технологии дипфейков. Дипфейки – это синтетические медиафайлы, созданные с помощью искусственного интеллекта. Они могут использоваться для создания видео, в которых люди говорят или делают то, что на самом деле никогда не говорили или не делали. Дипфейки могут использоваться в различных целях, включая рекламу, создание фейковых новостей и даже распространение дезинформации. Другим негативным последствием развития нейросетей является снижение доходов творческих профессионалов в индустрии развлечений. С появлением потокового телевидения, таких платформ как Netflix и других, актеры и сценаристы сталкиваются с резким снижением своего дохода. Обычно актеры и сценаристы получают больше денег, чем больше и дольше люди смотрят их шоу, сериалы или передачи. Однако с потоковыми сервисами все меняется. Формулировки в контрактах становятся все более расплывчатыми, и авторы контента получают не такую высокую оплату, как при работе на традиционном телевидении или кабельных средствах массовой информации (СМИ). Возможности искусственного интеллекта позволяют продюсерам создавать контент, включающий лица актеров, но без их фактического участия. Это основано на обработке сотен видеозаписей, которые загружаются в программу для обучения. Используя эти данные, искусственный интеллект (ИИ) может создавать новые видео, в которых актеры не участвовали. Это открывает новые возможности для создания контента, но также вызывает вопросы о создании оригинального и авторского материала.

При использовании нейросетей для обработки биометрических персональных данных можно привести список потенциальных опасностей:

1. Нарушение конфиденциальности: биометрические персональные данные уникальны для каждого человека и могут быть использованы для идентификации или аутентификации. Однако, если эти данные попадут в неправильные руки, это может привести к серьезным нарушениям конфиденциальности и злоупотреблению ими.

2. Возможность подделки: нейросети используются для аутентификации на основе биометрических персональных данных, таких как отпечатки пальцев или распознавание лица. Однако, существует риск подделки или обмана системы с помощью синтетических данных или масок, которые могут обмануть нейросеть.

3. Технические ошибки: нейросети несовершенны, и могут допускать ошибки при обработке биометрических персональных данных. Например, система распознавания лица может ошибочно определить личность, или система аутентификации отпечатка пальца может допустить ложное совпадение. Это может привести к доступу несанкционированных лиц или отказу доступа для санкционированных пользователей.

4. Недостаточная защита данных: для обучения нейросетей требуются большие объемы данных, включая биометрические персональные данные. Если эти данные недостаточно защищены, они могут быть украдены или использованы злоумышленниками. Также сами нейросети могут быть скомпрометированы, что может привести к утечке и злоупотреблению биометрическими данными.

5. Зависимость от технологии: внедрение нейросетей для обработки биометрических персональных данных создает зависимость от технологии. Если система нейросети выходит из строя или не функционирует должным образом, это может вызвать сбои в процессах и создать потенциальные проблемы для безопасности или доступа к информации.

Такая практика нерегулируемой обработки биометрических персональных данных приводит к злоупотреблению и их незаконному использованию. Вышеописанная ситуация произошла с актрисой озвучки Аленой Андроповой и акционерным обществом (АО) «Тинькофф Банк». Алена откликнулась на вакансию, в которой требовался женский голос для озвучивания крупного текста. Ее заверили, что ее голос будет использоваться для создания и обучения искусственного интеллекта. Данная нейросеть выступала как кол-центр и голос Алены должен быть использован только для внутренних задач акционерного общества (АО) «Тинькофф Банк». Через некоторое время знакомые начали утверждать, что узнают голос актрисы в других проектах, а именно контента для взрослых. Они выяснили, что на сайте банке доступен синтез голоса Алены для общего пользования. Получить ответ от юристов, представляющих акционерное общество (АО) «Тинькофф Банк», у актрисы не получилось. А ее голос начали использовать предприятия, с которыми Алена работала раньше. Теперь ее голос можно услышать в коммерческой рекламе, на стриминговых площадках и эротических чатах [1].

В последние годы биометрия все чаще используется в различных сферах, в том числе и в киноиндустрии. Использование биометрии в киноиндустрии открывает новые возможности для создания более реалистичных и захватывающих фильмов. Например, биометрия может использоваться для

создания цифровых копий актеров, что позволяет им сниматься в нескольких фильмах одновременно. Также биометрия может использоваться для создания новых спецэффектов, например, для создания виртуальных миров или для замены актеров в сценах со сложными трюками. Однако, как показывает зарубежная практика, с использованием биометрии в киноиндустрии связаны и некоторые риски. Один из самых масштабных инцидентов, связанных с использованием биометрии, произошел в Голливуде, где члены Гильдии актеров кино и телевидения объявили забастовку в знак протеста против возможности использования их произведений и биометрических данных в генерации сценариев с помощью искусственного интеллекта (ИИ). Сценаристы опасаются того, что искусственный интеллект (ИИ) может создавать собственные сценарии на основе их произведений. Это вызывает беспокойство в отношении авторских прав и возможности утраты контроля над созданием и распространением своих идей. Если искусственный интеллект (ИИ) станет способен генерировать качественные сценарии без участия сценаристов, это может повлечь за собой серьезные последствия для профессиональной карьеры и доходов сценаристов. Актеры также имеют свои причины для опасений. Новая схема работы, предлагаемая продюсерами, включает вызов актера на один съемочный день, после чего его внешность, голос и повадки сканируются. Затем в фильме используются цифровые образы актера. Это может быть выгодно для актера, так как он получает гонорар за один день работы, в то время как цифровому «клону» деньги не нужны. Однако, это вызывает вопросы в отношении личной приватности актеров и их способности контролировать использование своего образа и имиджа. Позиция К. Нолана: Ведущий американский еженедельник «Variety» цитирует Кристофера Нолана, который заявляет: «Мы должны требовать от людей отчетности за то, что они делают с инструментами, которые у них есть». Это подчеркивает необходимость более строгого регулирования использования биометрических данных в индустрии кино и защиты прав и интересов сценаристов и актеров. Последствия забастовки: Забастовка актеров и сценаристов привела к остановке ряда крупных проектов в индустрии кино. Сериалы, фильмы и вечерние шоу не снимаются, что негативно сказывается на развитии отрасли. Список замороженных проектов, скорее всего, будет только расти, пока не будут найдены решения, удовлетворяющие обе стороны конфликта [2].

Обратимся к зарубежной практике правового регулирования обработки биометрической информации, а именно «Общеевропейский Регламент о защите персональных данных» (2018 г.) и масштабному предложению европейской комиссии по регулированию созданию и применению искусственного интеллекта. На основе вышеуказанного регламента мы относим биометрию к особой категории персональных данных. Такое присвоение биометрических данных человека нередко встречается в нормативно-правовых актах стран Европейского союза (ЕС). Ключевым аспектом дан-

ного регламента являются определенные правила и критерии, по которым страны члены Европейского союза (ЕС) обязаны присваивать категорию биометрических данных на персональные данные граждан.

21 апреля 2021 года европейская комиссия опубликовала масштабное предложение по регулированию создания и применения искусственного интеллекта. В документе отмечаются новые правила по созданию человека-ориентированного, безопасного, доверенного искусственного интеллекта (ИИ), а также извлечение максимальной пользы из новых технологий с одновременным соблюдением прав человека. Данное предложение касается компаний, которые занимаются созданием, внедрением и эксплуатацией искусственного интеллекта (ИИ) на территории Европейского союза (ЕС). Также предложение обязывает компании осуществлять лицензирование своей продукции. Важным требованием европейской комиссии является техническая поддержка продукта компаниями после вывода его на рынок и создание специальных надзорных органов, которые проводят непрерывную оценку на соответствие искусственного интеллекта (ИИ) регламенту, предложенному комиссией. В свою очередь компании обязаны уведомлять органы обо всех пришествиях, связанные с работой их продукта [3].

Проанализировав данные нормативные акты, можно выделить следующие особенности обработки биометрических данных: отнесение биометрии к специальной категории, обязательное лицензирование искусственного интеллекта (ИИ) и получения согласия на обработку специальной категории персональной информации, отправка статистики, а также непрерывная техническая поддержка.

Давайте рассмотрим правовое регулирование в Российской Федерации (РФ). Здесь мы будем ссылаться на Федеральный закон (ФЗ) № 572 от 29 декабря 2022 г. «Об осуществлении идентификации и (или) аутентификации физических лиц с использованием биометрических персональных данных…» [4]. Общая схема работы с биометрическими персональными данными выглядит следующим образом: на основе существующего опыта работы с персональными данными, который практикуется несколько лет в банковской схеме и различных государственных организациях, сервисах создается единая система хранения биометрических данных физических лиц. Под биометрическими персональными данными подразумевается хранить и обрабатывать пару: лицо + голос. Доступ к данной системе получат только уполномоченные и аккредитованные организации, соблюдающие все необходимые требования, описанные в 572-ФЗ. Если организация хочет работать с персональными данными своих клиентов и сотрудников, то она должна пройти определенную аккредитацию и подключиться к единой биометрической системе (ЕБС). Согласно закону, всем организациям нужно будет работать с единым вектором биометрических персональных данных, который появляется после сдачи гражданином своих биометрических данных в одном из отделений многофункционального центра

предоставления государственных и муниципальных услуг (МФЦ). Работа с альтернативными биометрическими данными пока остается под вопросом. Сейчас закон требует проработку юристами, составление необходимых поправок и дополнений, получения разъяснений от государственных органов. На данный момент установлен переходный период до 1 декабря 2023 года.

Глобализация привела к тому, что персональная информация пользователей уже не привязана к национальности, и государства не могут претендовать на наделение ею юрисдикцией. В сущности, определение владельца такой информации порой становится невозможным. В связи с этим, обеспечение надежной защиты и сохранности огромного объема персональной информации становится особенно важным. В данной области одной из главных проблем является отсутствие четких критериев для определения, что является биометрическими данными. До 25 мая 2018 года на различных платформах и в источниках определение «биометрические данные» имело разные значения, но после введения регламента Европейского союза (ЕС) о защите персональных данных и их свободном использовании окончательное определение было установлено. Биометрические данные – это персональная информация, полученная специальной технической обработкой и относящаяся к физическим, физиологическим или поведенческим характеристикам физического лица, которые позволяют уникально идентифицировать этого индивида [5].

Проанализировав данное определение, предложение по регулированию искусственного интеллекта (ИИ), а также инциденты в области использования и обработки биометрических данных, можно предложить несколько возможных законодательных мероприятий, выступающих мерами для защиты прав и конфиденциальности людей:

1. Закон об индивидуальном согласии: Закон должен требовать получение ясного и информированного согласия каждого отдельного лица на использование его биометрических данных нейросетями. Это позволит людям контролировать, какую информацию и как она используется.

2. Закон о защите данных: Этот закон должен обеспечивать строгое правовое регулирование сбора, хранения и использования биометрических данных. Он должен обязывать компании и организации соблюдать высокие стандарты безопасности и принимать все необходимые меры для предотвращения несанкционированного доступа к биометрическим данным.

3. Закон о прозрачности: Этот закон должен требовать, чтобы алгоритмы нейросетей, использующие биометрические данные, были прозрачны и проверяемы. Он также должен обязывать операторов систем нейросетей сообщать пользователям о том, как именно их данные будут использоваться и в каких целях.

4. Закон о минимизации данных: Согласно этому закону, компании и организации должны использовать только необходимые минимальные данные для конкретной цели. Они не должны сохранять или хранить биометрические данные дольше, чем это необходимо для выполнения своих обязанностей или целей.

5. Закон о безопасности: Компании, использующие биометрию нейросетями, должны соблюдать строгие стандарты безопасности для защиты биометрических данных от несанкционированного доступа, взлома или утечки информации. Они должны также применять меры для обеспечения конфиденциальности и целостности этих данных.

6. Закон об ответственности: Закон должен обязывать операторов систем нейросетей нести ответственность за неправомерное использование биометрических данных или нарушение прав людей. Операторам должна быть предоставлена возможность возмещения ущерба и вынесения санкций в случае нарушения закона.

С бурным развитием нейросетей и повсеместным внедрением их в нашу жизнь, такими же быстрыми темпами происходит сбор, обработка и незаконное распространение биометрических данных. Требуются рычаги регулирования данного процесса на законодательном уровне. Законы, посвященные использованию биометрии нейросетями, должны строиться на основе принципов прозрачности, конфиденциальности и защиты прав личности. Они должны ориентироваться на обеспечение безопасности и контроля данных, минимизацию рисков и обеспечение справедливого и этичного использования технологии нейросетей и биометрических данных.

Библиографический список

1. Актриса озвучки подала в суд на «Тинькофф» за использование её голоса в других проектах без разрешения // Хабр URL: https://habr.com/ru/news/758172/ (дата обращения: 30.09.2023).

2. Претендент на главную роль: как искусственный интеллект используют в кино // Известия URL: https://iz.ru/1550851/alena-svetunkova/pretendent-na-glavnuiu-rol-kak-iskusstvennyi-intellekt-ispolzuiut-v-kino (дата обращения: 30.09.2023).

3. Архипов В.В., Наумов В.Б. Теоретико-правовые вопросы охраны прав человека при использовании биометрических данных системами искусственного интеллекта: европейский опыт // Вестник Удмуртского университета. 2022. №1.

4. Закон Российской Федерации "Об осуществлении идентификации и (или) аутентификации физических лиц с использованием биометрических персональных данных, о внесении изменений в отдельные законодательные акты Российской Федерации и признании утратившими силу отдельных положений законодательных актов Российской Федерации" от 29.12.2022 № 572 // Собрание законодательства Российской Федерации.

5. Кривогин М.С. Особенности правовой охраны биометрических персональных данных в странах европейского союза // Отечественная юриспруденция. 2017. №4.

УДК 004.056

МЕТОДЫ УСОВЕРШЕНСТВОВАНИЯ МОДУЛЯ «1С: ОБЩЕЖИТИЕ» ДЛЯ ОБЕСПЕЧЕНИЯ ЗАЩИТЫ ПЕРСОНАЛЬНЫХ ДАННЫХ ПРОЖИВАЮЩИХ

Е.А. Лаптева, С.В. Бегичева
Научный руководитель: доц. С.В. Бегичева
Уральский государственный экономический университет, г. Екатеринбург

В статье рассматривается проблема обеспечения защиты персональных данных студентов в информационной системе 1С: Университет ПРОФ. Приводится ряд методов, которые помогут сделать подсистему управления кампусом вуза, и в том числе модуль 1С: Общежитие более эффективными и удобными в использовании, а также более защищенными в аспекте безопасности персональных данных.

Ключевые слова: 1С: Университет ПРОФ, защита персональных данных, заселение в общежитие вуза, персональные данные.

Проблема обеспечения защиты персональных данных (ПДн) является важной в современном мире, так как увеличивается количество собираемых и обрабатываемых ПДн, в том числе с помощью интернета и новых технологий. При этом возрастают риски несанкционированного доступа к персональным данным и их использования в незаконных целях.

Так, во время заселения студентов в общежитие возникает проблема обеспечения защиты персональных данных студентов при занесении информации о них в информационную систему и последующем хранении этой информации. Проблема вызвана рядом факторов:

– с каждым годом все больше информации о студентах собирается и обрабатывается при заселении в общежитие;

– персональные данные студентов необходимы для осуществления различных бизнес-процессов в учебном заведении, однако, возникает риск, что ПДн будут использованы не по прямому назначению или переданы третьим лицам без согласия студентов.

– в последние годы атаки на информационные системы становятся все более сложными. Киберпреступники активно ищут уязвимости в информационных системах, чтобы получить доступ к персональным данным и использовать их в мошеннических, финансовых и других целях. Это пред-

ставляет угрозу для безопасности и конфиденциальности персональных данных студентов в общежитиях.

Указанные факторы делают проблему обеспечения защиты ПДн при занесении информации о студентах во время заселения в общежитие и при хранении этих данных в информационных системах актуальной. Одним из важных шагов в оптимизации бизнес-процессов и усовершенствовании управлением и защитой информацией в Уральском государственном экономическом университете является переход от текущей единой информационной системы (ЕИС) на информационную систему 1С: Университет ПРОФ. Продукты 1С являются одними из самых популярных и широко используемых пакетов программного обеспечения для автоматизации бизнес-процессов. Сравним текущую и внедряемую информационные системы и выявим плюсы и минусы (табл. 1).

Таблица 1

Сравнение ЕИС и 1С: Университет ПРОФ

Критерий	ЕИС	1С: Университет [1]
Интегрированность	Возможны сложности при интеграции с другими системами для обмена информацией	Легкая настройка и интеграция с другими системами для обмена информацией, при этом канал по которому передаются данные, может быть защищен
Затраты	Настройка и сопровождение являются достаточно дорогостоящими	Более доступен в плане стоимости
Использование	Интерфейс является сложным, из-за чего возникает необходимость проведения длительного обучения сотрудников, а также требования больших технических ресурсов и времени при доработке	Простой и интуитивно понятный интерфейс, что упрощает работу сотрудникам и предотвращает необходимость проведения длительного обучения
Отчетность и аналитика	Предоставляет не все необходимые отчеты	Предоставляет многочисленные отчеты и аналитические инструменты для анализа предоставленных данных

Переход с ЕИС на информационную систему 1С является непростым, но выгодным шагом для организации. Правильно спланированный и реализованный переход позволит оптимизировать бизнес-процессы, улучшить управление данными и упростить работу сотрудников. В нашем ВУЗе бы-

ла введена система 1С: Университет ПРОФ, в рамках которой работает модуль 1С: Общежитие. Этот модуль позволяет вести учёт проживающих в общежитии, учёт передвижения проживающих, формировать перечень тарифов и услуг, составлять договоры на оказание услуг и другое (рис. 1).

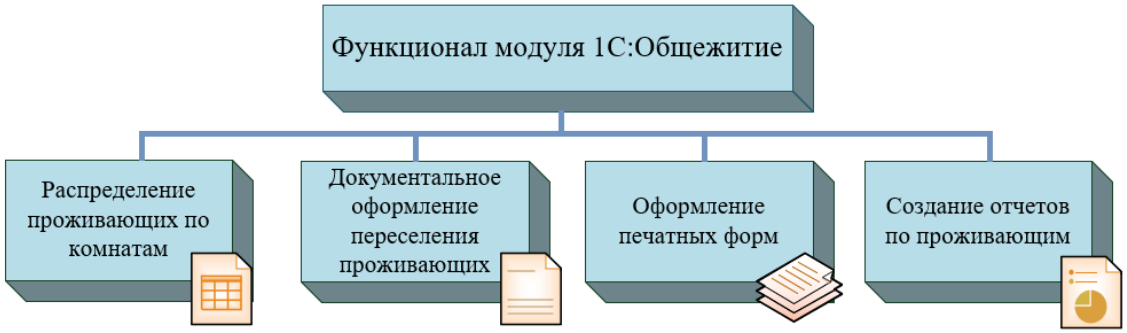


Рис. 1. Функционал модуля 1С: Общежитие

Однако, как и любая другая информационная система, модуль 1С: Общежитие требует постоянного усовершенствования и оптимизации. Одна из возможных причин такого усовершенствования – требование улучшения защиты персональных данных абитуриентов, студентов, слушающих, сотрудников [2], а также оптимизация бизнес-процессов студенческого общежития.

Основными законодательными актами, регулирующими защиту персональных данных, являются Федеральный закон от 27.07.2006 № 149-ФЗ «Об информации, информационных технологиях и о защите информации»[3], а именно п. 9 ст. 9 данного закона, в котором говориться, что порядок доступа к персональным данным устанавливается федеральным законом от 27.06.2006 № 152-ФЗ «О персональных данных» (далее – закон «О персональных данных») [4]. В соответствии со статьей 5 закона «О персональных данных» устанавливаются определенные принципы обработки персональных данных, такие как:

– обработке подлежат только персональные данные, которые отвечают целям их обработки (п. 4);

– обрабатываемые персональные данные не должны быть избыточными по отношению к заявленным целям их обработки (п. 5);

– при обработке персональных данных должны быть обеспечены точность персональных данных, их достаточность, а в необходимых случаях и актуальность по отношению к целям обработки персональных данных (п. 6);

– и другие принципы, перечисленные в статье 5 закона «О персональных данных».

Основными бизнес-процессами в общежитии, при выполнении которых возникает проблема защиты персональных данных, являются:

– заселение абитуриентов, студентов, сотрудников в общежитие;

– переселение студентов, сотрудников в общежитии;

– выселение студентов, сотрудников из общежития.

Для оптимизации перечисленных выше бизнес-процессов укажем методы, которые позволят сделать систему более эффективной и удобной в использовании:

– составление отчётов о заселившихся и проживающих непосредственно в модуле 1С: Общежитие, не прибегая к использованию других информационных систем. Таким образом снижается риск утечки персональных данных проживающих, возникающий при интеграции нескольких информационных систем. Реализация возможности анализа данных в системе 1С позволит оперативно контролировать обстановку в общежитии и принимать решения на основе фактической информации.

– обновление процедуры создания приказов, что позволит защитить персональные данные тех лиц, которые не заселяются в общежитие и брать в работу данные только тех студентов, которые подали заявление о необходимости предоставить им место в общежитии. Также необходимо добавление заселившихся в приказ с применением фильтров, что уменьшает работу с использованием персональных данных.

Усовершенствование модуля 1С: Общежитие может повысить качество обслуживания студентов, оптимизировать управление ресурсами и обеспечить более безопасное и комфортное проживание. Автоматизация учета ресурсов, улучшение системы безопасности и анализ данных – это лишь несколько путей для достижения поставленной цели. Усовершенствование 1С: Общежитие в соответствии с потребностями и требованиями учебного заведения может значительно улучшить работу общежития на базе университета.

Библиографический список

1. 1С: Университет // Отраслевые и специализированные решения 1С: Предприятие: [сайт]. – URL: https://solutions.1c.ru/catalog/university (дата обращения: 13.10.2023).

2. Терещенко Л.К., Кривогин М.С. Особенности правового регулирования общедоступных персональных данных / Терещенко Л.К., Кривогин М.С. // № 2(24), (2017): Вестник УрФО. Безопасность в информационной сфере. – г. Челябинск, 2017. – С. 57–64.

3. Федеральный закон от 27.07.2006 № 149-ФЗ «Об информации, информационных технологиях и о защите информации».

4. Федеральный закон от 27.06.2006 № 152-ФЗ «О персональных данных».

УДК 004.056

АНАЛИЗ СУЩЕСТВУЮЩИХ МЕТОДИК ОЦЕНИВАНИЯ УРОВНЯ ЗАЩИЩЕННОСТИ ИНФОРМАЦИОННЫХ СИСТЕМ

Э.Р. Кухмазов, И.Р. Зулькарнеев
Научный руководитель: доц. И.Р. Зулькарнеев
Тюменский государственный университет,
г. Тюмень

Для всех видов информационных систем необходимо регулярно проводить оценку их уровня защищенности. В статье описаны существующие методы оценки уровня защищенности информационных систем и проведен их сравнительный анализ по сформулированным авторами критериям. Были выявлены недостатки проанализированных методов и сделан вывод об их субъективности, сложности использования, превалирования качественных показателей над количественными, недостаточном учете требований НПА по защите информации. Авторы статьи предложили идеи по повышению точности и объективности вычисления текущего уровня защищенности.

Ключевые слова: аудит информационной безопасности, информационная система, количественные и качественные показатели, методика, оценка уровня защищенности.

В соответствии с действующими нормативными правовыми актами в области защиты информации владелец или оператор информационных систем (далее – ИС) должен регулярно проводить оценку защищенности эксплуатируемых ИС, результатом которой станет актуальный уровень защищенности ИС (далее – УЗ), позволяющий оценить, насколько эффективны существующие меры системы защиты информации. Данное требование применимо ко всем ИС: для информационных систем персональных данных оно описано в ст. 18.1 и в ст. 19 [1], для автоматизированных систем управления технологическим процессом в п. 16.8 [2], для государственных информационных систем в п. 18.7 [3], для объектов критической информационной инфраструктуры в пунктах 10, 36 и 37 [4].

На текущий момент не существует четких или обязательных требований по проведению оценки защищенности ИС, также как и не существует действующего стандарта или даже рекомендаций. В связи с этим определение УЗ носит хаотичный и неструктурированный характер, зависящий от компетенций и возможностей проверяющего и оператора ИС. В попытке решить данную проблему, было разработано множество различных методик оценки защищенности информационных систем.

Рассмотрим некоторые из методик определения уровня защищенности. Так, в статье [5] представлена методика анализа слабости хостов с помо-

щью CWE, CWSS и CVSS. Для каждого хоста выводится уровень его важности в сети и критичности уязвимостей. Сначала собирается банк знаний об уязвимостях и слабостях хостов. Потом к полученным данным применяется метод нечетких моделей, при котором строятся имитационные системы, где из множества частных переменных выводится актуальный УЗ. К недостаткам данного метода можно отнести сложность проводимых вычислений и то, что их необходимо выполнять для каждого АРМ, что для больших предприятий займет больше времени и ресурсов.

В работе [6] коллектив ученых предлагает использовать метод оценивания УЗ с помощью статических и динамических показателей. Статические показатели – информация, которая в предприятии практически не изменяется со временем, (например, оборудование) а динамические показатели представляют собой наиболее изменчивые характеристики ИС. (например, информация об инцидентах). С их помощью они формируют семантическую модель метрик и данных. Описанная модель в разработанной авторами онтологии, функционирует на связях типа «целое – частное», из которых определяется необходимый способ оценивания. В данной методике в основном преобладают качественные значения, которые сложно объективно измерить.

Авторы статьи [7] представляют экспертный метод анализа УЗ посредствам аудита информационной безопасности, в рамках которого происходит сравнение требований нормативно-правовых актов и реализованных защитных мер, а также используются различные программные средства для поиска уязвимостей. Полученные в ходе поиска уязвимости анализирует эксперт. Минусом данного метода, является формальный подход к оцениванию эффективности защитных мер и использование технической информации без учета связи объектов ИС между собой, а также субъективность оценки эксперта.

В статье [8] описан метод оценки УЗ с помощью формирования нечетких временных рядов. Метод заключается в формировании различных версий прогнозируемого уровня защищенности на основе эмпирических данных о системах защиты информации и их сопоставлении друг с другом. Главным недостатком данного метода будет являться нехватка эмпирических данных у предприятий, недавно начавших свою деятельность. Проводить оценку УЗ с помощью нечетких временных рядов очень сложно, так как в данном методе анализируется влияние разных версий друг на друга.

Работа [9] представляет нам методику оценивания УЗ путем создания модели графов отношений объектов ИС с их уязвимостями и воздействиями на них. На основе веса ребер делается вывод об УЗ отдельных хостов, а впоследствии и об УЗ самой организации. Данный метод подразумевает большое количество информационно-технических воздействий на критически важные объекты инфраструктуры. Основным недостатком данной методики будет являться ее направление на поиск минимального количе-

ства информационно-технических воздействий, необходимых для определения УЗ без учета проверки организационных и правовых мер, а также требований НПА.

Рассмотрев вышеперечисленные методики по отдельности, был проведен их сравнительный анализ с точки зрения используемых данных ИС для того, чтобы определить:

- элементы ИС, которые оказывают большее влияние на УЗ;
- сложности, возникают при использовании методики;
- простоту понимания полученного результата.

Для сравнения методов определим следующие критерии.

1. Средства защиты информации (далее – СрЗИ) – критерий, учитывающий количество и эффективность существующих в ИС СрЗИ, и оценивающий устойчивость ИС к воздействию со стороны злоумышленников.

2. Размер инфраструктуры – критерий, учитывающий количество автоматизированных рабочих мест, серверов, активов, интернет-ресурсов, и др.

3. Уязвимости – критерий, показывающий данные о существующих уязвимостях, их количестве, и критичности.

4. Информация по инцидентам – критерий по оценке информации о произошедших в компании инцидентах ИБ, и о предпринятых мерах по их нейтрализации и превентивной защите в будущем.

5. Проверка требований НПА – критерий, анализирующий степень соответствия реализованных мер защиты информации требованиям законодательства по защите информации.

6. Затраты на ИБ – критерий, оценивающий уровень затрат предприятия на защиту информации, в том числе СрЗИ и мероприятия.

7. Ущерб ИС – критерий подсчета материального ущерба от реализации угроз и эксплуатации уязвимостей.

8. Сотрудники – критерий, описывающий количество сотрудников-пользователей ИС, их уровень доступа, полномочия, подготовку и осведомленность в области ИБ.

9. Физическая защита – критерий, анализирующий физические способы контроля и реализации доступа, обеспечение физической безопасности технических средств.

10. Сложность использования – критерий, раскрывающий простоту использования метода и количество затрачиваемых для его реализации ресурсов.

11. Формат выходных данных – критерий, оценивающий формат и простоту понимания выходных данных после использования методики.

Основываясь на критериях, которые были разобраны выше, была составлена сравнительная таблица различных методик оценки УЗ (табл. 1).

Таблица 1

Сравнительный анализ методик нахождения УЗ

Критерии оценивания	Нечеткие временные ряды	Семантический анализ	Экспертный метод	Имитационные системы	Информационно-технические воздействия
СрЗИ	Да	Да	Да	Да	Да
Размер инфраструктуры	Нет	Да	Да	Да	Да
Уязвимости	Да	Да	Нет	Да	Да
Информация по инцидентам	Да	Нет	Нет	Нет	Нет
Проверка требований НПА	Нет	Нет	Да	Нет	Нет
Затраты на ИБ	Нет	Нет	Нет	Да	Да
Ущерб ИС	Нет	Да	Нет	Да	Да
Сотрудники	Нет	Нет	Да	Нет	Нет
Физическая защита	Нет	Да	Да	Нет	Нет
Сложность использования	Сложный	Простой	Простой	Сложный	Сложный
Формат выходных данных	Графики с прогнозируемыми УЗ	Генерируется формула вычисления интегральных метрик, отвечающих на вопросы оценивания защищенности	Отчет	Отчет	Модель защищенности объекта с использованием графов

Проведя анализ полученной таблицы, становится ясно, что в каждом из методов встречаются как качественные, так и количественные показатели, при этом доля количественных меньше качественных, следовательно каждый из этих методов в значительной степени полагается на субъективность оценки проверяющего (аудитора). Для повышения объективности процесса оценки УЗ и ее независимости от уровня аудитора можно повысить долю количественных показателей, с учетом их влияния на эффективность системы защиты информации, на реализующие ее процессы и на выполнение требований законодательства по информационной безопасности. В результате видно, что существующие методики используют лишь малую часть количественных показателей, либо не учитывают НПА, либо сложны в использовании, что ведет к неполноценности рассчитываемой оценки уровня защищенности и необходимости её совершенствования за счет повышения уровня объективности, то есть увеличения числа используемых количественных показателей информационной системы.

В ходе исследования были проанализированы различные методики определения УЗ, проведен их сравнительный анализ и составлена таблица, иллюстрирующая их схожести и различия. Было сделано предположение, что метод, основанный на подсчете количественных значений ИС, будет более объективным, нежели качественный метод, так как точность его измерений будет лишена субъективности. В рамках данной статьи было проанализировано 5 методов оценки УЗ ИС, также были выявлены критерии для сравнения различных методик. Был произведен сравнительный анализ различных методов оценивания УЗ, результатом которого стал вывод о том, что необходимо повышать точность оценки УЗ.

Библиографический список

1. Федеральный закон. «О персональных данных» от 27.07.2006 N 152-ФЗ. URL: https://www.consultant.ru/document/cons_doc_LAW_61801/ (дата обращения: 22.10.2023).

2. Приказ ФСТЭК России «Об утверждении требований к обеспечению защиты информации в автоматизированных системах управления производственными и технологическими процессами на критически важных объектах, потенциально опасных объектах, а также объектах, представляющих повышенную опасность для жизни и здоровья людей и для окружающей природной среды» от 14 марта 2014 г. № 31. URL: https://fstec.ru/dokumenty/vse-dokumenty/prikazy/ prikaz-fstek-rossii-ot-14-marta-2014-g-n-31 (дата обращения: 25.10.2023).

3. Приказ ФСТЭК России «Об утверждении требований о защите информации, не составляющей государственную тайну, содержащейся в государственных информационных системах» от 11 февраля 2013 г. № 17. URL: https://fstec.ru/dokumenty/vse-dokumenty/prikazy/prikaz-fstek-rossii-ot-11-fevralya-2013-g-n-17 (дата обращения: 26.10.2023).

4. Приказ ФСТЭК России «Об утверждении требований к созданию систем безопасности значимых объектов критической информационной инфраструкту-

ры российской федерации и обеспечению их функционирования» от 21 декабря 2017 г. № 235. URL: https://fstec.ru/dokumenty/vse-dokumenty/prikazy/prikaz-fstek-rossii-ot-21-dekabrya-2017-g-n-235 (дата обращения: 27.10.2023).

5. Курочкин С.И. Методы оценки уровня защищенности информационных систем / Курочкин С.И., Заводцев И.В. // «Перспективы развития информационных технологий – 2016 г. – С. 194–204.

6. Дойникова Е.В. Методика оценивания защищенности на основе семантической модели метрик и данных / Дойникова Е.В., Федорченко А.В., Котенко И.В., Новикова Е.С. // «Вопросы кибербезопасности» – 2021 г. – С. 29–40.

7. Запороцков П.А. Разработка метода проведения аудита системы технической защиты информации / Запороцков П.А. // «NBI-technologies» – 2020 г. – С. 18–27.

8. Пивкин Е.Н. Алгоритм прогнозирования оценок уровня защищенности объектов информатизации на основе нечетких временных рядов / Е.Н. Пивкин // «Ползуновский вестник» – 2011 г. – С. 229–232.

9. Макаренко С.И. Модель аудита защищенности объекта критической информационной инфраструктуры тестовыми информационно-техническими воздействиями / Макаренко С.И., Смирнов Г.Е. // «Труды учебных заведений связи.» – 2021. Т. 7. № 1. – С. 94–104.

УДК 004.056.5

ПРОБЛЕМА ЗАЩИТЫ НАУЧНОГО ОБОРУДОВАНИЯ ОТ УГРОЗ БЕЗОПАСНОСТИ ИНФОРМАЦИИ

А.А. Забокрицкий, А.В. Фурик
Научный руководитель: канд. техн. наук, доц. А.С. Коллеров
Уральский федеральный университет
имени первого Президента России Б.Н. Ельцина,
г. Екатеринбург

В статье рассматривается проблема защиты научного оборудования, анализируется наличие уязвимостей такого оборудования, в том числе специального программного обеспечения. В официальной базе сведения о таких уязвимостях отсутствуют. Результаты анализа отечественных и зарубежных источников показывают, что данная проблема является малоизученной. Определено, что научное оборудование, не отнесенное к значимым объектам критической информационной инфраструктуры, подлежит защите от угроз информационной безопасности. Указывается на необходимость корректировки показателей критериев значимости объектов критической информационной инфраструктуры Российской Федерации и их значений, для исключения случаев отнесения научного оборудования к незначимым объектам критической информационной инфраструктуры. Данное исследо-

вание представляет интерес для научных и образовательных организаций, использующих научное оборудование.

Ключевые слова: банк данных угроз, информационная безопасность, научное оборудование, объекты критической информационной инфраструктуры, уязвимости.

Стратегия научно-технологического развития Российской Федерации [1] определяет, что одним из значимых для научно-технологического развития Российской Федерации внутренним фактором является резкое увеличение объема научно-технологической информации, возникновение принципиально новых способов работы с ней и изменение форм организации, аппаратных и программных инструментов проведения исследований и разработок. Данное обстоятельство обуславливает необходимость обеспечение безопасности информации, обрабатываемой с использованием научного оборудования.

В Стратегии национальной безопасности [2] обозначено, что достижение цели обеспечения информационной безопасности осуществляется путем реализации государственной политики, направленной на решение, в том числе задачи предотвращения деструктивного информационно-технического воздействия на российские информационные ресурсы, включая объекты критической информационной инфраструктуры Российской Федерации (далее – объекты КИИ).

В последнее время на информационные ресурсы Российской Федерации совершаются массовые кибератаки. Об этом, в частности свидетельствует сообщение – в 2022 году в топ-6 самых атакуемых отраслей вошел научный сектор [3].

В этой связи представляется целесообразным изучение проблемы защиты информационной инфраструктуры научной организации, в частности научного высокотехнологического оборудования, предназначенного для исследований.

Целью данного исследования является выявление проблем защиты научного оборудования и пути их решения.

Согласно Классификатору научного оборудования [4], в научных организациях используется разнообразное оборудование. При этом к автоматизированным системам (далее – АС) относится не все оборудование. При рассмотрении научного оборудования как объектов КИИ, для руководства научным организациям необходим типовой отраслевой перечень. В настоящее время такой перечень не опубликован. Вместе с тем, такое оборудование как объекты КИИ можно классифицировать следующим образом:

АС управления жизненным циклом изделия (продукции);

АС управления испытательными стендами (оборудованием);

АС управления научной организацией;

информационные системы лабораторий.

На основании данных официального сайта Научно-технологическая инфраструктура Российской Федерации [5] выбрано для исследования научное оборудование, имеющееся в центрах коллективного пользования в г. Екатеринбурге, следующих иностранных производителей: BioMerieux (Франция), Bruker (США), Instron (США), JEOL (Япония), Malvern Instruments (Великобритания), NETZSCH (Германия), PerkinElmer Inc. (США), Renishaw plc (Великобритания), Rigaku Corporation (Япония), Thermo Fisher Scientific (США). Выборочный анализ научного оборудования, производимого данными фирмами, показал, что системы приборов управляются встроенными цифровыми электронными устройствами, специальным программным обеспечением, установленным на стандартном персональном компьютере под управлением операционных систем семейства Windows, которые также проводят анализ измеренных данных.

В ходе исследования, по результатам изучения банка данных угроз (https://bdu.fstec.ru) уязвимостей в специальном программном обеспечении выбранного оборудования не обнаружено. Данное обстоятельство свидетельствует о том, что поиск уязвимостей в указанном оборудовании сообществом в сфере информационной безопасности осуществляется на недостаточном уровне.

Вместе с тем, учитывая, что согласно банку данных угроз (https://bdu.fstec.ru) операционные системы семейства Windows имеют множественные уязвимости, то актуальным является вопрос принятия необходимых мер защиты информации в отношении такого научного оборудования.

Анализ отечественных и зарубежных источников по проблеме наличия уязвимостей в научном оборудовании и защиты научной информации от киберугроз, показывает, что данная проблема является малоизученной. В российских источниках имеются отдельные публикации на тему информационной безопасности медицинского оборудования [6]. В иностранной печати изучались вопросы кибербезопасности научных исследований [7, 8], также имеются руководства по обеспечению безопасности исследований [9, 10].

Авторы указанных статей в основном исследуют организационный аспект информационной безопасности научного оборудования, предложения по выявлению и устранению уязвимостей не приводятся.

Согласно действующему законодательству в области обеспечения безопасности критической информационной инфраструктуры Российской Федерации, меры защиты информации в соответствии с Требованиями по обеспечению безопасности значимых объектов критической информационной инфраструктуры Российской Федерации [11] должны приниматься в отношении научного оборудования, отнесенного к значимым объектам КИИ. При этом, согласно Требованиям, анализ угроз безопасности информации должен включать, в том числе анализ возможных уязвимостей зна-

чимого объекта и его программных, программно-аппаратных средств. Анализ уязвимостей значимого объекта проводится в целях выявления недостатков (слабостей) в подсистеме безопасности значимого объекта и оценки возможности их использования для реализации угроз безопасности информации. При этом, анализу подлежат уязвимости кода, конфигурации и архитектуры значимого объекта. По результатам анализа уязвимостей должно быть подтверждено, что в значимом объекте, отсутствуют уязвимости, как минимум содержащиеся в банке данных угроз безопасности информации ФСТЭК России, или выявленные уязвимости не приводят к возникновению угроз безопасности информации в отношении значимого объекта.

С применением инструментария нового раздела банка данных угроз ФСТЭК России проведено моделирование перечня угроз безопасности для типового объекта (автоматизированное рабочее место АС управления научным оборудованием) с минимально возможным перечнем негативных последствий: Н14 (нарушение штатного режима функционирования автоматизированной системы управления и управляемого объекта и/или процесса и Н25 [невозможность решения задач (реализации функций) или снижение эффективности решения задач (реализации функций)] для нарушителя обладающего базовыми возможностями.

Результаты моделирования показывают возможность реализации следующих угроз безопасности информации:

УБИ.3. Угроза несанкционированной модификации (искажения);

УБИ.4. Угроза несанкционированной подмены;

УБИ.5. Угроза удаления информационных ресурсов;

УБИ.6. Угроза отказа в обслуживании;

УБИ.7. Угроза ненадлежащего (нецелевого) использования;

УБИ.9. Угроза получения информационных ресурсов из недоверенного или скомпрометированного источника.

Основными мерами защиты, связанными с возможными уязвимостями, являются:

АУД.2.1. Выявление (поиск) уязвимостей, связанных с ошибками кода в программном (микропрограммном) обеспечении (общесистемном, прикладном, специальном), а также программном обеспечении средств защиты информации, правильностью установки и настройки средств защиты информации, технических средств и программного обеспечения, а также корректностью работы средств защиты информации при их взаимодействии с техническими средствами и программным обеспечением;

АУД.2.2. Разработка по результатам выявления (поиска) уязвимостей отчетов с описанием выявленных уязвимостей и планом мероприятий по их устранению; анализ отчетов с результатами поиска уязвимостей и оценки достаточности реализованных мер защиты информации;

АУД.2.3. Устранение выявленных уязвимостей, в том числе путем установки обновлений программного обеспечения средств защиты информации, общесистемного программного обеспечения, прикладного программного обеспечения или микропрограммного обеспечения технических средств;

АУД.2.4. Информирование должностных лиц оператора (пользователей, администраторов, подразделения по защите информации) о результатах поиска уязвимостей и оценки достаточности реализованных мер защиты информации;

ОПС.2.5. Определение и применение параметров настройки компонентов программного обеспечения, включая программные компоненты средств защиты информации, обеспечивающих реализацию мер защиты информации, а также устранение возможных уязвимостей информационной системы, приводящих к возникновению угроз безопасности информации.

Данные результаты показывают важность решения вопросов по выявлению уязвимостей.

Вместе с тем, в открытых источниках отсутствует информация о наличии на территории Свердловской области значимых объектов критической информационной инфраструктуры в сфере науки, несмотря на важность обеспечения защиты научного оборудования. Как следствие, меры по защите информации такого оборудования не принимаются. Данное обстоятельство является предпосылкой к необходимости корректировки законодательства в части правил категорирования объектов критической информационной инфраструктуры Российской Федерации, а также показателей критериев значимости объектов критической информационной инфраструктуры Российской Федерации и их значений.

Заключение. На основании проведенного анализа, считаем целесообразным продолжить исследования по выявлению уязвимостей научного оборудования. Установлена необходимость корректировки показателей критериев значимости объектов критической информационной инфраструктуры Российской Федерации и их значений, для исключения случаев отнесения научного оборудования к незначимым объектам критической информационной инфраструктуры. В связи с этим, научное оборудование, не отнесённое к значимым объектам КИИ, требует принятие мер защиты информации.

Библиографический список

1. О Стратегии научно-технологического развития Российской Федерации: Указ Президента Российской Федерации от 1 декабря 2016 г. № 642 // URL: http://www.kremlin.ru/acts/bank/41449 (дата обращения: 15.10.2023).

2. О Стратегии национальной безопасности Российской Федерации: Указ Президента Российской Федерации от 2 июля 2021 г. № 400 // URL: http://www.kremlin.ru/acts/bank/47046 (дата обращения: 16.10.2023).

3. Больше всего в 2022 году в России хакеров интересовали государственный, медицинский и промышленный секторы // URL: http://www.servernews.ru/1080271 (дата обращения: 16.10.2023).

4. О Классификаторе научного оборудования: Приказ Министерства образования и науки Российской Федерации от 29 июля 2016 г. № 925 // URL: https://www.garant.ru/products/ipo/prime/doc/71357882 (дата обращения: 18.10.2023).

5. Научно-технологическая инфраструктура Российской Федерации // URL: https://ckp-rf.ru/ (дата обращения: 22.10.2023).

6. Хвостов В.А. и др. Анализ угроз информационной безопасности медицинскому оборудованию медицинских организаций и основные направления защиты информации / Журнал Системный анализ и управление в биомедицинских системах. 2022, – Т.21, № 3. – С. 95–103.

7. David S. Butcher, Christian J. Brigham, J. Berhalter. Cybersecurity in a Large-Scale Research Facility – One Institution's Approach // Cybersecurity and Privacy. – 2023. – № 3. – P. 191–208.

8. Shankar A., Drake W. Effective Cybersecurity for Research // URL: https://scholarworks.iu.edu/dspace/handle/2022/27733 (дата обращения: 27.10.2023).

9. Guidance for Research Data and Materials Security // URL: https://lindenwood.edu/files/resources/20170726-research-data-and-materials-security.pdf (дата обращения: 27.10.2023).

10. Guidance for implementing national security presidential memorandum 33 (nspm-33) on national security strategy for united states government-supported research and development. A Report by the Subcommittee on Research Security Joint Committee on the Research Environment // URL: https://www.whitehouse.gov/wp-content/uploads/2022/01/010422-NSPM-33-Implementation-Guidance.pdf (дата обращения: 22.10.2023).

11. Об утверждении Требований по обеспечению безопасности значимых объектов критической информационной инфраструктуры Российской Федерации: Приказ ФСТЭК России от 25 декабря 2017 г. № 239 // URL: https://fstec.ru/dokumenty/vse-dokumenty/prikazy/prikaz-fstek-rossii-ot-25-dekabrya-2017-g-n-239 (дата обращения: 13.10.2023).

УДК 004.056

АНАЛИЗ ПОСЛЕДНИХ ИЗМЕНЕНИЙ ФЕДЕРАЛЬНОГО ЗАКОНА № 152-ФЗ «О ПЕРСОНАЛЬНЫХ ДАННЫХ»

В.А. Новожилова, Т.Ю. Зырянова
Научный руководитель: канд. техн. наук, доц. Т.Ю. Зырянова
Уральский государственный университет путей сообщения, г. Екатеринбург

В статье приведен анализ последних изменений в Федеральном законе № 152-ФЗ «О персональных данных», которые были внесены в связи с появлением новых технологий, развитием интернета и электронной коммерции, а также необходимостью укрепления защиты конфиденциальных данных населения в современных реалиях.

Ключевые слова: персональные данные, субъект, Федеральный закон № 152-ФЗ «О персональных данных».

Федеральный закон № 152-ФЗ «О персональных данных» [1] является важным средством регулирования обработки персональных данных в Российской Федерации. Закон был принят в 2006 году и в дальнейшем неоднократно изменялся и дополнялся, чтобы соответствовать современным требованиям и вызовам, связанным с развитием технологий и информационной безопасности.

Цель работы – проанализировать последние изменения, внесенные в Федеральный закон № 152-ФЗ «О персональных данных».

С 1 сентября 2022 года вступили в силу поправки [2] в Федеральный закон от 27 июля 2006 г. № 152-ФЗ «О персональных данных», утвержденные Федеральным законом от 14 июля 2022 года № 266-ФЗ «О внесении изменений в Федеральный закон «О персональных данных», отдельные законодательные акты Российской Федерации и признании утратившей силу части четырнадцатой статьи 30 Федерального закона «О банках и банковской деятельности» [3].

Внесенные поправки касаются:

- обслуживания субъектов, отказывающих в предоставлении персональных данных;
- целей обработки персональных данных;
- информирования об инцидентах уполномоченных органов власти;
- сроков подачи уведомлений в Роскомнадзор;
- уничтожения персональных данных;
- новых полномочий Роскомнадзора при передаче персональных данных за границу;
- оценки вреда утечки персональных данных.

Рассмотрим каждый пункт в отдельности.

1. Обслуживание субъектов, отказывающих в предоставлении персональных данных.

В связи с изменениями [4], субъект в праве не предоставлять свои персональные данные, если предоставление не является обязательным. Чаще всего предоставление персональных данных необходимо в работе компаний. На этот случай для операторов дополнена обязанность по разъяснению субъекту персональных данных юридических последствий отказа предоставить его персональные данные и (или) дать согласие на обработку персональных данных, в случае если предоставление таких данных и (или) получение согласия на обработку персональных данных являются обязательными.

2. Цели обработки персональных данных.

Нововведение в этом пункте предусматривает разработку перечня целей обработки персональных данных, состава персональных данных и их категорий, а также категорий субъектов персональных данных.

3. Информирование об инцидентах уполномоченных органов власти.

В случае возникновения инцидентов с принадлежащими базами персональных данных необходимо незамедлительно информировать об этом уполномоченные органы власти (Роскомнадзор, ФСБ России).

Оператор персональных данных обязан уведомить органы исполнительной власти об утечке персональных данных. Для этого он направляет уведомление в Роскомнадзор.

С момента вступления в силу Приказа Роскомнадзора от 27.10.2022 № 178 в действие вступили требования к операторам персональных данных, согласно которым необходимо составить акт, содержащий оценку возможных уровней риска при обработке персональных данных. В этом акте оператор должен указать, какой вред может быть причинен субъекту персональных данных в случае нарушения Федерального закона «О персональных данных». Оценка возможного вреда проводится ответственным за организацию обработки персональных данных или комиссией, назначенной оператором.

4. Сроки подачи уведомлений в Роскомнадзор.

Данное изменение касается статьи 22 «Уведомление об обработке персональных данных» [2]. Организации обязаны отправлять в Роскомнадзор уведомление о сборе персональных данных, а также уведомление об изменении предоставленной информации.

Ранее операторы должны были отправлять уведомление об изменении представленных персональных данных в течение 10 дней после их корректировки. Однако, с 1 марта 2023 года этот срок увеличен. Теперь, если в персональных данных произошли изменения, оператору разрешается отправить уведомление в Роскомнадзор не позднее 15-го числа месяца, следующего за месяцем, в котором произошли изменения.

5. Уничтожение персональных данных.

Согласно Приказу Роскомнадзора от 28.10.2022 № 179 операторам обработки персональных данных требуется подтверждать удаление персональных данных. Документ содержит инструкции и требования к данной процедуре.

Уничтожение персональных данных на бумажных носителях должно быть задокументировано актом. В данном акте должны быть указаны категория уничтожаемых персональных данных, Ф.И.О. и должности участников процедуры, их подписи, причины и методы уничтожения и т.д.

Если электронные персональные данные удаляются из информационной системы, то достаточно того, чтобы была сделана выгрузка журнала событий, в которой будут отображены все события, связанные с этими персональными данными. Если в выгрузке не указаны определенные данные, они должны быть дополнены бумажным актом. В нем должна быть указана недостающая информация. Срок хранения такого акта составляет 3 года.

6. Новые полномочия Роскомнадзора при передаче персональных данных за границу.

Если компания передает персональные данные за границу, то она должна сообщить об этом в Роскомнадзор. Это требование установлено Федеральным законом от 14.07.2022 № 266-ФЗ [3] и действует с 1 марта 2022 года. Предоставление персональных данных за границу может происходить, например, при сотрудничестве с иностранными партнерами, направлении сотрудников в командировку или на обучение за рубеж.

Уведомление о трансграничной передаче направляется в Роскомнадзор:

- единоразово, до осуществления отправки персональных данных за рубеж;
- отдельно от других уведомлений об обработке персональных данных;
- с указанием всех стран, куда оператор персональных данных уже передаёт персональные данные.

Форма уведомления о намерении трансграничной передачи осталась без изменений, согласно тексту поправок статьи 12 152-ФЗ [1]. Ранее Роскомнадзор не имел возможности принимать решения о запрете или ограничении отправки персональных данных. Теперь, после 1 марта 2023 года, данное ведомство получило полномочия принимать собственные решения по вопросу о возможности обеспечения должной защиты персональных данных в зарубежных странах.

Оператор персональных может сразу начать обработку данных после отправки уведомления о трансграничной передаче. В течение 10 дней Роскомнадзор проводит свою проверку иностранного получателя, и после

этого у ведомства есть право запретить или ограничить передачу персональных данных без судебного разбирательства.

Операторы персональных данных могут осуществлять трансграничную передачу только после получения уведомления от Роскомнадзора, но они также могут столкнуться с повторным запретом от ведомства.

Роскомнадзор имеет возможность запретить всем операторам передачу персональных данных в определенную страну или установить ограничения для конкретного оператора персональных данных без необходимости судебного разбирательства. Такие меры были приняты для защиты моральности, здоровья, прав и законных интересов граждан. Решение Роскомнадзора может быть оспорено в судебном порядке или у вышестоящего руководителя ведомства.

Изменения в Федеральный закон № 152-ФЗ «О персональных данных» были внесены в связи с появлением новых технологий, развитием интернета и электронной коммерции, а также необходимостью укрепления защиты конфиденциальных данных населения в современных реалиях.

Внесенные изменения направлены на регулирование использования персональных данных в сфере информационных технологий, порядка сбора, хранения и передачи персональных данных, а также на установление требований к операторам персональных данных.

Внедрение данных изменений в Федеральный закон № 152-ФЗ «О персональных данных» позволит обеспечить усиление защиты прав и свобод физических лиц при обработке их персональных данных.

Библиографический список

1. Федеральный закон от 27.07.2006 №152-ФЗ «О персональных данных» (последняя редакция) / [Электронный ресурс] //: [сайт]. – URL: https://www.consultant.ru/document/cons_doc_LAW_61801/ (дата обращения: 25.10.2023).

2. Обзор изменений Федерального закона от 27.07.2006 №152-ФЗ «О персональных данных» / [Электронный ресурс] //: [сайт]. – URL: https://www.consultant.ru/document/cons_doc_LAW_94223/fd49f79396a6da140d633e6f45c6c13d5b130535/ (дата обращения: 25.10.2023).

3. Федеральный закон от 14.07.2022 №266-ФЗ «О внесении изменений в Федеральный закон «О персональных данных», отдельные законодательные акты Российской Федерации и признании утратившей силу части четырнадцатой статьи 30 Федерального закона "О банках и банковской деятельности» / [Электронный ресурс] //: [сайт]. – URL: http://publication.pravo.gov.ru/Document/View/0001202207140080 (дата обращения: 25.10.2023).

4. Соколов В. Новации в сфере обработки и защиты персональных данных / Вячеслав Соколов [Электронный ресурс] //: [сайт]. – URL: Новации в сфере обработки и защиты персональных данных (дата обращения: 25.10.2023).

УДК 004.056

ОРГАНИЗАЦИЯ ПРОЦЕССА УПРАВЛЕНИЯ УЯЗВИМОСТЯМИ

М.Е. Трубина
Научный руководитель: канд. техн. наук, доц. Т.Ю. Зырянова
Уральский государственный университет путей сообщения,
г. Екатеринбург

Описан процесс управления уязвимостями. Основные этапы процесса и его необходимость. Приведен пример реализации процессов, описанных в Методическом документе ФСТЭК России от 17.05.2023 г. Данная тема исследования актуальна в современном мире, результаты исследования подтверждают необходимость построения процесса управления уязвимостями в организации.

Ключевые слова: информационная безопасность, обновление программного обеспечения, управление уязвимостями, ФСТЭК России.

В современном информационном обществе, где технологии играют все более важную роль, вопросы безопасности становятся неотъемлемой частью деловой среды. Уязвимости в информационных системах могут представлять серьезную угрозу для организаций, нанося значительный ущерб и нарушая их деятельность и репутацию. Организация процесса управления уязвимостями становится необходимостью для защиты информации и обеспечения своевременного реагирования на новые угрозы и атаки. В данной статье мы рассмотрим важность и основные аспекты организации процесса управления уязвимостями.

Управление уязвимостями достаточно трудоемкий процесс. Основные его этапы описаны в Методическом документе ФСТЭК России от 17.05.2023 г. «Руководство по организации процесса управления уязвимостями в органе (организации)». Руководство представляет собой документ, регламентирующий процесс обеспечения безопасности информационных систем от уязвимостей в организациях [1]. Основной целью данного методического документа является минимизация уязвимостей, которые могут быть использованы злоумышленниками для проникновения и эксплуатации информационных систем. Методический документ ФСТЭК России определяет основные принципы и методы управления уязвимостями, а также обязательные требования к процессу и процедурам обнаружения, анализа и устранения уязвимостей. В документе также прописаны требования к организации мониторинга и аудита уязвимостей, а также созданию центра управления уязвимостями. Кроме того, документ устанавливает

требования к штатной организационной структуре, включающей специалистов по управлению уязвимостями и их обязанности.

В руководстве ФСТЭК России детально описаны и представлены схематически этапы процесса управления уязвимостями. В данном документе выделяют пять этапов реализации процесса управления уязвимостями (рис. 1).

Рис. 1. Этапы работ по управлению уязвимостями

На основании данного руководства работниками отдела информационной безопасности организуется процесс управления уязвимостями, разделенный на следующие этапы (табл. 1).

На первом этапе проводится сканирование сети при помощи сканера уязвимостей MaxPatrol8 или же с использованием программного обеспечения (ПО) ScanOval. В данном случае использование ПО ScanOval нерационально, поскольку отсутствует возможность сканирования всей сети за раз. Второй этап нацелен на анализ уязвимостей на актуальность, даются ответы на вопросы «Возможна ли эксплуатация уязвимости?» и «Есть ли возможность обновления?». Для эксплуатации некоторых уязвимостей требуется удаленный доступ, который отсутствует в организации, следовательно, уязвимость неактуальна.

Разработка организационно-распорядительных документов (ОРД) и назначение ответственных лиц являются соответственно третьим и четвертым этапами процесса, в них входит разработка регламентов и политик установки обновлений ПО, в которых так же указываются лица, ответственные за установку обновлений. При дальнейшей работе подразделения обязаны руководствоваться разработанной ОРД. Пятым этапом является установка обновлений на «тестовом полигоне», представляющем собой парк виртуальных машин, с установленным на них основным ПО, используемым работниками подразделений Организации. Данный этап предна-

значен для определения работоспособности ПО после установки обновлений. Далее на шестом и седьмом этапе проводится установка обновлений и контроль работоспособности систем. Если после установки ПО будут наблюдаться сбои, то обновления откатываются до исходной версии и реализуется восьмой этап – разработка и реализация компенсирующих мер. Так же этот этап реализуется по отношению к обновлениям, которые невозможно установить по геополитическим и иным причинам. После установки обновлений и реализации компенсирующих мер проводится повторное сканирование на наличие уязвимостей. Для оставшихся уязвимостей, которые невозможно устранить, разрабатываются и реализуются компенсирующие меры, что и является заключительным этапом.

Таблица 1

Этапы работ по управлению уязвимостями

№ *n\n*	Наименование мероприятия	Ответственные подразделения
1	Выявление уязвимостей	Отдел информационной безопасности
2	Анализ уязвимостей на актуальность	
3	Работы в области организационно-распорядительных документов	
4	Назначение ответственных лиц	
5	Установка обновлений на «тестовом полигоне»	
6	Установка обновлений на информационные системы организации	Отдел информационных технологий
7	Контроль работоспособности систем	
8	Разработка и реализация компенсирующих мер	Отдел информационной безопасности
9	Повторное сканирование на наличие уязвимостей	
10	Поиск решений для устранения оставшихся уязвимостей	Отдел информационной безопасности, Отдел информационных технологий, подразделения, эксплуатирующие программное обеспечение

Данный процесс необходимо реализовывать с периодичностью, определяемой специалистами отдела информационной безопасности Организации.

Организация процесса управления уязвимостями в организации играет ключевую роль в обеспечении информационной безопасности и защите от угроз. В современном информационном обществе, где данные становятся основным активом, необходимо активно заниматься выявлением, анализом и устранением уязвимостей в информационных системах. Организация процесса управления уязвимостями позволяет компаниям минимизировать возможность проникновения хакеров и нанесения ущерба их деятельности. Более того, это также способствует соблюдению требований регуляторных органов в области информационной безопасности. Важно отметить, что организация процесса управления уязвимостями должна быть основана на систематическом подходе, который включает в себя постоянные аудиты, мониторинг и анализ уязвимостей, а также разработку соответствующих планов и мер для их устранения. Регулярное обновление и обеспечение безопасности информационной инфраструктуры также важны для минимизации рисков, связанных с уязвимостями. Организации, которые активно обращают внимание на организацию процесса управления уязвимостями, могут рассчитывать на повышение своей защищенности и снижение потенциальных угроз. Тем самым, они сохраняют доверие клиентов и партнеров, а также продолжают успешно развивать свой бизнес. В итоге организация процесса управления уязвимостями является неотъемлемой частью информационной безопасности организации. Это не только позволяет снизить уровень рисков, но и создает условия для стабильного и безопасного функционирования информационных систем. Организации, следующие рекомендациям и требованиям в этой области, получают значительные преимущества в сфере информационной безопасности и остаются готовыми к современным угрозам и вызовам.

Библиографический список

1. Методический документ Руководство по организации процесса управления уязвимостями в органе (организации) // URL: https://fstec.ru/dokumenty/vse-dokumenty/spetsialnye-normativnye-dokumenty/metodicheskij-dokument-ot-17-maya-2023-g (дата обращения: 20.10.2023).

УДК 004

АНАЛИЗ БЕЗОПАСНОСТИ ПРИМЕНЕНИЯ ЦИФРОВОГО РУБЛЯ, ОСОБЕННОСТЕЙ ПРИМЕНЕНИЯ ТЕХНОЛОГИИ БЛОКЧЕЙН

Е.С. Науменко
Научный руководитель: канд. техн. наук, доц. А.С. Коллеров
Уральский федеральный университет
имени первого Президента России Б.Н. Ельцина,
г. Екатеринбург

Работа посвящена анализу безопасности и особенностей применения цифрового рубля и технологии блокчейн и представляет собой критически важную задачу в контексте обеспечения защиты данных, конфиденциальности и надежности в сфере цифровых финансов. Проведен обширный обзор аспектов безопасности, связанных с использованием цифрового рубля и технологии блокчейн, а также рассмотрены ключевые меры и инновации, направленные на обеспечение безопасного и надежного взаимодействия в мире цифровых валют и распределенных реестров.

Ключевые слова: блокчейн, криптовалюты, цифровой рубль.

С появлением цифрового рубля и расширением практического использования технологии блокчейн, вопросы безопасности стали одними из ключевых аспектов, которые необходимо учесть и рассмотреть. Целью данной статьи является исследование двух фундаментальных элементов современной экономической экосистемы: цифровой рубль и технология блокчейн, а также выявление их важности в контексте обеспечения безопасности криптовалютных операций и хранения цифровых активов. **Блокчейн** (или blockchain) – это децентрализованный и распределенный реестр, который используется для регистрации транзакций и записи данных. Эта технология была изначально разработана как основа для криптовалюты, такой как биткоин, но с течением времени она нашла широкое применение в различных сферах [1].

- Цифровой рубль: Эволюция денег в цифровую эпоху

Цифровой рубль представляет собой собрание электронных активов, которые выдаются Центральным банком Российской Федерации и работают на базе цифровых технологий. Он представляет собой новый этап в эволюции денег, обещая улучшить множество аспектов финансовых операций, включая быстроту, доступность и удобство. Однако, в свете растущей сложности киберугроз и киберпреступлений, обеспечение безопасности цифрового рубля становится приоритетной задачей.

- Технология блокчейн: Основа безопасности

Технология блокчейн, на которой строится цифровой рубль, является децентрализованным и недеформируемым реестром, который записывает все транзакции в виде блоков. Одной из ее ключевых особенностей является прозрачность, так как информация обо всех операциях доступна публично, и не может быть изменена без согласия большинства участников сети. Эта характеристика делает технологию блокчейн надежной основой для криптовалют и других цифровых активов.

- Важность безопасности в мире криптовалют

Безопасность играет ключевую роль в обеспечении доверия к цифровому рублю и технологии блокчейн. Киберугрозы, включая взломы, мошенничество и кражи, представляют серьезные риски как для индивидуальных пользователей, так и для финансовых институтов. Безопасность также связана с защитой личных данных и обеспечением конфиденциальности финансовых транзакций.

В этой статье будет подробно рассмотрен как цифровой рубль, так и технология блокчейн, и будут выявлены меры и инновации, которые используются для обеспечения их безопасности. Также будут рассмотрены актуальные вызовы, с которыми сталкиваются разработчики и пользователи криптовалют, и будет выполнен поиск пути укрепления безопасности в контексте цифрового будущего.

Люди склонны считать, что цифровой рубль и всем привычные денежные средства [3], хранимые на банковских картах это одно и то же, но на самом деле это два разных формата хранения денежных средств наравне с наличными деньгами (табл. 1).

Таблица 1

Сравнение использование привычной банковской карты и технологии цифрового рубля

Критерий	Цифровой рубль	Банковская карта
Форма хранения и передачи	Это цифровая валюта, которая существует в цифровой форме и хранится в цифровых кошельках. Его транзакции и баланс записываются в распределенном реестре (блокчейне)	Это физическая карта, на которой хранится информация о вашем банковском счете. Деньги хранятся на банковском счете, и банковская карта используется для доступа к этим средствам через банкоматы или точки продаж
Доступность	Доступен для использования в онлайн-среде и может использоваться для мгновенных электронных транзакций в любой точке мира, поддерживающей цифровой рубль	Для использования банковской карты требуется физическая карта и доступ к банкомату или точке продаж. Это ограничивает ее использование вне физического мира

Окончание табл. 1

Критерий	Цифровой рубль	Банковская карта
Транзакционные издержки	Могут быть небольшие комиссии при обмене и переводах, но они часто ниже, чем банковские комиссии	Банки часто взимают комиссии за использование банковских карт, а также за международные операции или снятие наличных средств из банкомата
Прозрачность и безопасность	Использует технологию блокчейн, которая обеспечивает высокий уровень прозрачности и безопасности транзакций	Здесь безопасность в значительной степени зависит от мер безопасности, принимаемых банками, и иногда может быть подвержена мошенничеству
Валютные ограничения	Используется для операций в российских рублях	Может использоваться для операций в разных валютах, но может потребовать конвертации с дополнительными издержками

В общем, цифровой рубль и банковская карта предоставляют разные средства для хранения и управления денежными средствами. Выбор между ними зависит от потребностей, предпочтений и того, как будут использоваться деньги.

Внедрение цифрового рубля в России представляет собой стратегический шаг, направленный на решение ряда современных потребностей экономики и финансовой системы [2]. Основные цели внедрения цифрового рубля и связь с современными потребностями включают:

1. Содействие цифровой экономике: Цифровой рубль способствует развитию цифровой экономики, увеличивая доступность цифровых платежей и финансовых услуг. В мире, где онлайн-коммерция и цифровые платежи становятся все более распространенными, это содействует росту экономики.

2. Увеличение прозрачности и борьба с коррупцией: Использование технологии блокчейн позволяет создавать неизменяемые и прозрачные записи транзакций, что может снизить возможности финансовых мошенничеств и коррупции.

3. Повышение доступности финансовых услуг: Цифровой рубль может увеличить доступность финансовых услуг для тех слоев населения, которые ранее не имели доступа к традиционным банковским услугам. Это способствует финансовой инклюзии и социальной справедливости.

4. Увеличение эффективности банковской системы: Использование цифрового рубля может упростить и ускорить процессы банковских операций, что может снизить затраты на обслуживание клиентов и увеличить эффективность финансовой системы.

5. Содействие инновациям: Цифровой рубль создает новые возможности для развития инноваций и стартапов, связанных с криптовалютами и

блокчейном, и может привлечь инвестиции в сферу финансовых технологий.

6. Международные финансовые операции: Цифровой рубль может упростить международные финансовые транзакции и деловую активность, что способствует развитию международного бизнеса.

7. Защита от внешних угроз: В условиях геополитических и экономических рисков цифровой рубль может обеспечивать большую степень независимости и безопасности от внешних воздействий.

Внедрение цифрового рубля ориентировано на создание современной, эффективной и безопасной финансовой системы, которая отвечает потребностям как граждан, так и бизнеса в эпоху цифровой трансформации. Это также позволяет России активно участвовать в глобальном движении к развитию цифровой экономики и финансовой инновации.

Технология блокчейн – это распределенная и недеформируемая система, предназначенная для регистрации транзакций и данных [5]. Она обеспечивает безопасность транзакций и данных путем ряда ключевых механизмов (рис. 1):

- Децентрализация:

Блокчейн не имеет центрального управления; вместо этого, он состоит из множества узлов (компьютеров), которые образуют сеть. Каждый узел имеет копию всего блокчейна, что делает его децентрализованным.

Децентрализация повышает безопасность, поскольку отсутствие единой центральной точки означает, что нет одной точки отказа. Каждый узел в сети проверяет и подтверждает транзакции, что делает манипуляции трудными.

- Шифрование:

Для обеспечения конфиденциальности данных и безопасности транзакций блокчейн использует криптографию. Все транзакции в блокчейне шифруются и подписываются цифровыми подписями.

Шифрование делает данные неразборчивыми для нежелательных глаз и обеспечивает безопасность персональной информации.

- Распределенный журнал:

Блокчейн сохраняет все транзакции в виде блоков, которые связаны в цепь. Это создает неизменяемую историю транзакций.

Поскольку данные хранятся на всех узлах в сети, изменение одного блока требует согласия большинства участников, что делает манипуляции данными крайне сложными.

- Консенсус и проверка:

Блокчейн использует механизмы консенсуса для определения правильности транзакций и добавления их в блокчейн.

Примеры механизмов консенсуса включают "доказательство работы" (Proof of Work) и "доказательство участия" (Proof of Stake). Они обеспечивают согласие всех участников на правильность транзакций.

- Надежность и устойчивость:

Блокчейн обеспечивает высокую надежность и устойчивость за счет распределения и дублирования данных на всех узлах сети.

Даже если некоторые узлы выходят из строя или подвергаются атакам, система продолжает работать.

- Смарт-контракты:

Блокчейн также может использовать смарт-контракты, программные коды, которые автоматически выполняют и обеспечивают выполнение условий сделки без необходимости посредников.

Смарт-контракты повышают надежность и уровень доверия в сделках.

Таким образом, технология блокчейн обеспечивает безопасность транзакций и данных за счет децентрализации, шифрования, надежности, исключения посредников и дублирования данных. Эти характеристики делают блокчейн подходящим для различных применений, включая криптовалюты, цифровые контракты и обеспечение прозрачности в различных отраслях.

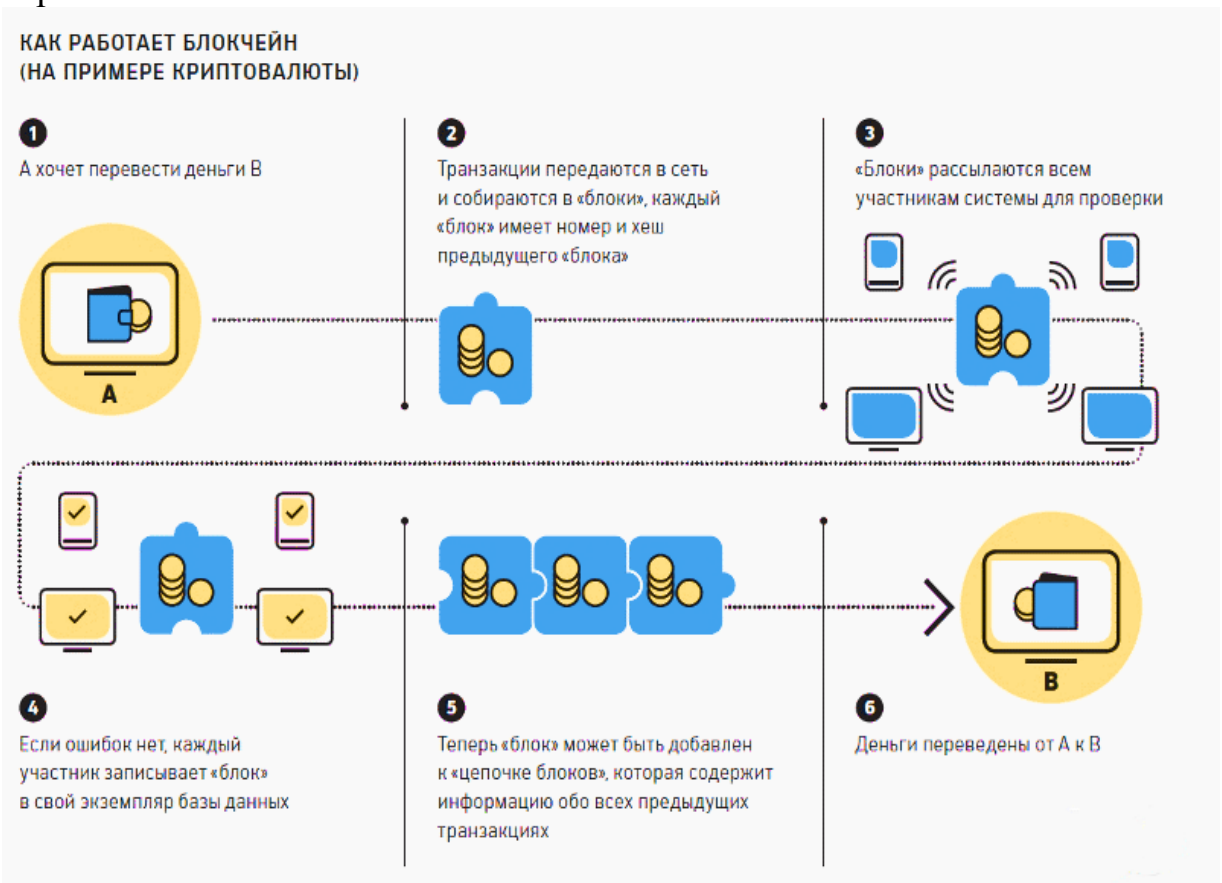


Рис. 1. Схема работы системы блокчейн

Использование цифрового рубля и технологии блокчейн предоставляет множество преимуществ, но также сопряжено с рядом потенциальных угроз и рисков, которые необходимо учитывать (табл. 2).

Таблица 2

Риски, угрозы и предлагаемые меры при использовании цифрового рубля

Риски	Угрозы	Меры
Кибератаки и хакерские атаки	Киберпреступники могут пытаться взломать цифровые кошельки или блокчейн-сети для кражи криптовалюты	Использование надежных кошельков, многозначных аутентификаций и обновление программного обеспечения для безопасности
Потеря доступа к кошельку	Забытый пароль или потеря доступа к кошельку может привести к потере средств	Надежное хранение паролей, создание резервных копий кошельков и использование восстановительных фраз
Мошенничество и фишинг	Мошенники могут пытаться обмануть пользователей и получить доступ к их кошелькам или данным	Бдительность при обращении с незнакомыми источниками, проверка адресов и сетевых ресурсов
Недостоверные ICO и проекты	Инвестиции в нечестные или мошеннические ICO (Initial Coin Offering) могут привести к финансовым потерям	Тщательный анализ проектов перед инвестициями и использование надежных платформ
Регуляторные риски	Изменения в законодательстве и регулировании могут повлиять на использование криптовалют и блокчейна	Соблюдение местных нормативов и законов [4]
Потеря ключей и кошельков	Потеря закрытых ключей или доступа к кошельку может быть источником финансовой утраты	Тщательное хранение ключей и создание резервных копий
Ценовая волатильность	Криптовалюты известны своей ценовой волатильностью, что может повлиять на инвестиции и стоимость активов	Разумное планирование инвестиций и диверсификация портфеля
Использование в незаконных целях	Криптовалюты могут использоваться для незаконных операций, таких как отмывание денег и финансирование терроризма	Соблюдение законодательства и надлежащей практики

Обеспечение безопасности при использовании цифрового рубля и технологии блокчейн важно для защиты активов и данных (табл. 3).

Таблица 3

Меры и методы обеспечения безопасности

Меры безопасности	Методы достижения безопасности
Многофакторная аутентификация (MFA)	Использование MFA для дополнительного уровня безопасности при входе в кошельки и аккаунты. Это может включать в себя пароль, ПИН-код и биометрическую аутентификацию, такую как скан отпечатков пальцев
Хранение криптовалюты	Следует хранить криптовалюту в надежных кошельках. Холодные кошельки (hardware wallets) считаются наиболее безопасными, так как они не подключены к интернету и не подвержены сетевым атакам
Резервные копии и восстановление	Создание резервных копий ключей и кошельков, и хранение их в надежном месте. Это позволит восстановить доступ в случае утери или повреждения основного кошелька
Обновление программного обеспечения	Регулярное обновление программного обеспечения своего кошелька и других инструментов. Обновления часто включают улучшенные меры безопасности
Бдительность при фишинге	Следует внимательно обрабатывать сообщения и электронные письма, особенно при запросах о предоставлении личных данных или ключей. Так же нужно проверять подлинность веб-сайтов и отправителей писем
Регуляторные меры и соблюдение законодательства	Соблюдение местного законодательства и регулирования в области криптовалют. Работа с легитимными и регулируемыми платформами может снизить риски
Параноидальная безопасность	К безопасности следует подходить с повышенным уровнем внимательности. Не следует разглашать свои ключи и данные, и следует всегда подозревать потенциальные угрозы
Антивирусное ПО	Использование надежного антивирусного программного обеспечения, для защиты активов от вредоносных программ и кибератак
Образование и осведомленность	Обучение и поддержание осведомленности в области криптовалют и блокчейна помогут принимать более осознанные решения и избегать рисков
Защита от физического доступа	Физическая защита устройства, на которых хранятся криптовалюты. Это может включать в себя сейфы и другие средства защиты

Соблюдение этих мер и методов поможет обеспечить безопасное использование цифрового рубля и технологии блокчейн, а также защитить ваши активы от потенциальных угроз и рисков.

В заключении следует отметить, что несмотря на имеющийся ряд рисков и угроз при использовании цифрового рубля и технологии блокчейн существует множество преимуществ, которые отвечают потребностям как граждан, так и бизнеса в эпоху цифровой трансформации, что дает силы России активно участвовать в глобальном движении к развитию цифровой экономики и финансовой инновации.

Библиографический список

1. Цифровой рубль: большой справочник по технологиям, возможностям и рискам // URL: https://habr.com/ru/articles/754148/ (дата обращения: 18.10.2023).

2. Цифровой рубль как инструмент обеспечения финансового контроля // URL: https://cyberleninka.ru/article/n/tsifrovoy-rubl-kak-instrument-obespecheniya-finansovogo-kontrolya/viewer (дата обращения: 17.10.2023).

3. Цифровой рубль: что это такое? // URL: https://journal.tinkoff.ru/guide/digital-currency/ (дата обращения: 26.10.2023).

4. ФЗ от 24.07.2023 № 339-ФЗ "О внесении изменений в статьи 128 и 140 части первой, часть вторую и статьи 1128 и 1174 части третьей Гражданского кодекса Российской Федерации" // URL: http://publication.pravo.gov.ru/document/0001202307240009?index=1 (дата обращения: 25.10.2023).

5. Использование технологии блокчейн для проверки подлинности электронных документов и файлов // URL: https://www.info-secur.ru/index.php/ojs/article/view/408 (дата обращения: 26.10.2023).

УДК 004.056.5

ПРОБЛЕМЫ ЗАЩИТЫ ИНФОРМАЦИИ ПРИ РАЗРАБОТКЕ И ЭКСПЛУАТАЦИИ СИСТЕМ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ НА ПРЕДПРИЯТИЯХ РОЗНИЧНОЙ ТОРГОВЛИ

С.А. Сабельников
Научный руководитель: канд. техн. наук, доц. А.Н. Соколов
Южно-Уральский государственный университет,
г. Челябинск

В статье сформулирована проблема защиты информации в современных системах поддержки принятия решений на российских предприятиях розничной торговли. Результаты работы являются основой для проведения исследования, целью которого будет разработка и научное обоснование модели системы поддержки принятия решений на предприятиях розничной торговли с учетом требований и мер защиты информации.

Ключевые слова: защита информации, системы поддержки принятия решений.

В последние годы развитие корпоративных информационных систем в российских розничных сетях характеризуется активной разработкой, внедрением и использованием систем поддержки принятия решений (СППР). Этому способствует ряд факторов таких как: рост объема накопленных предприятиями данных, развитие технологий анализа данных и повышение требований к эффективности принятия решений.

Основная задача СППР – обеспечить автоматизированный анализ и генерацию данных и управленческих решений [2]. Основными тенденциями развития в области СППР можно выделить использование искусственного интеллекта и машинного обучения для анализа данных и предоставления рекомендаций. Наиболее подробно необходимо рассматривать методы поддержки принятия решений:

- методы, основанные на математическом аппарате алгебры нечеткой логики;
- методы, основанные на использовании систем искусственного интеллекта [2].

Использование данных методов при разработке СППР позволяет автоматизировать процесс принятия решений и повысить его точность и эффективность. Также наблюдается увеличение использования мобильных приложений и облачных сервисов для доступа к данным розничных сетей, с целью принятия решений в реальном времени. Это позволяет организациям быстро реагировать на изменения внешней среды, однако, в этом случае, вопрос защиты облачной среды является актуальной задачей информационной безопасности [3]. Еще одной тенденцией развития СППР, является интеграция их с другими системами, такими как: учетные системы, системы прогнозирования потребительского спроса, системы бизнес-аналитики, корпоративными порталами. Это позволяет улучшить координацию между различными подразделениями организации и контрагентами, повысить эффективность работы в целом. Кроме того, растет интерес к использованию социальных медиа и анализа больших данных для получения информации о поведенческих моделях потребителей и конкурентах. Это позволяет компаниям лучше понимать потребности своих клиентов и принимать более обоснованные маркетинговые решения. Наконец, наблюдается рост интереса к использованию блокчейн-технологии в СППР, которая может позволить создать надежный верификационный центр для электронных документов и файлов [7].

Тенденции предполагают, что СППР будут использовать более быстрый доступ в реальном времени к крупным интегрированным базам данных [4]. Принимая во внимание характер использования систем поддержки принятия решений как одного из основных инструментов, используемых для ана-

лиза и обработки больших объемов данных в корпоративных системах розничных торговых сетей. Учитывая описанные выше тенденции, которые оказывают прямое влияние на процесс разработки, эксплуатации СППР и ведут к необходимости осмысления подходов и проблем защиты информации в данных системах.

С ростом использования СППР возрастает и риск потери, искажения или несанкционированного доступа к информации ограниченного доступа. В первую очередь, актуальность защиты информации в СППР обусловлена тем, что утечка данных может привести к значительным финансовым потерям, а также ущербу репутации организации, ее контрагентов и нарушению их конкурентных преимуществ. Неконтролируемый доступ к информации может привести к серьезным юридическим последствиям, включая судебные иски и штрафы. Отдельным аспектом актуальности защиты информации является и то, что СППР используются для принятия важных решений, которые могут оказать влияние на жизнь и благосостояние большого числа людей. Если информация, используемая в СППР, будет скомпрометирована, это может повлечь за собой серьезные последствия.

Руководствуясь тенденциями развития СППР, очевидным становится тот факт, что при их разработке нужно учитывать риски и угрозы информационной безопасности. Актуальность проблемы защиты информации в СППР заключается в отсутствии комплексного подхода к обеспечению процесса защиты информации при разработке и внедрении таких систем, который бы учитывал современные тенденции развития СППР и потребности российских розничных сетей. Это позволяет определить возможные меры защиты информации при разработке, эксплуатации СППР в современных условиях:

1. Обеспечение конфиденциальности информации. Ввиду особенностей современных СППР, не только авторизованные пользователи имеют доступ к необходимым данным, но и авторизованные приложения, сервисы анализа данных, web-сервисы. СППР, которые используют большие объемы данных для анализа и принятия решений, представляют собой особую угрозу для розничных предприятий. Современный масштаб СППР накладывает требования к организации процесса разграничения прав доступа пользователей. Развитие в сторону интеграции с учетными системами, облачными сервисами влечет за собой существенный рост числа пользователей и применение различных методов разграничение информационных потоков и прав. Обеспечение конфиденциальности информации в СППР включает в себя ряд мер, направленных на защиту данных от несанкционированного доступа, утечки и искажения. Эти меры могут включать в себя шифрование данных, контроль доступа, аудит безопасности и обучение пользователей.

2. Обеспечение целостности информации. Архитектурные особенности СППР, создают дополнительные риски целостности информации, кроме тех, которые присущи любым распределенным базам данных. Решением

данной проблемы может быть внедрение технологии блокчейн при разработке СППР.

3. Обеспечение безопасности данных при передаче. Шифрование данных является одним из наиболее эффективных способов защиты конфиденциальной информации. Оно позволяет скрыть содержимое данных от неавторизованных пользователей, даже если они получат доступ к самим данным. Методы машинного обучения могут быть эффективно применены в области инженерно-технической защиты информации. Существует возможность использования нейронных сетей для классификации частот, которые могли быть угрозой утечки информации [7].

4. Управление рисками. Риск-менеджмент включает в себя оценку возможных угроз и определение мер по их предотвращению. Под рисками информационной безопасности понимается возможность того, что угроза сможет воспользоваться уязвимостью актива или группы активов и тем самым нанесет ущерб организации [6].

5. Обучение пользователей. Пользователи должны знать о возможных угрозах и мерах защиты информации в СППР. Обучение пользователей необходимо для того, чтобы они понимали важность конфиденциальности.

6. Мониторинг и аудит. Система защиты информации должна быть постоянно контролироваться и анализироваться для выявления возможных уязвимостей. Аудит безопасности позволяет выявить и устранить уязвимости в системе, которые могут использоваться для несанкционированного доступа к данным. Однако, обеспечение конфиденциальности в СППР не ограничивается только техническими мерами. Важно также учитывать организационные аспекты, такие как политика безопасности, процедуры обработки данных и ответственность за нарушение конфиденциальности. Обеспечение безопасности информации в СППР требует комплексного подхода и постоянного мониторинга и аудита системы.

Реализация данных мер должна находить свое отражение в процессе разработки всех основных функциональных компонентов СППР: базы данных, базы моделей и программной подсистемы, которая состоит из трех подсистем: системы управления базой данных (СУБД), системы управления базой моделей (СУБМ) и системы управления интерфейсом между пользователем и компьютером [5].

Заключение. В статье описаны современные тенденции разработки, эксплуатации систем поддержки принятие решений с точки зрения вопросов защиты информации. Анализ тенденций развития СППР показывает, что сокращение рисков реализации угроз информационной безопасности возможно добиться только применением комплексного подхода к защите информации с использованием методов искусственного интеллекта при разработке и эксплуатации СППР.

Библиографический список

1. Витенбург Е.А. Формализованная модель системы интеллектуальной поддержки принятия решений в области защиты информации // Известия ТулГУ. – Тула: 2017. – С. 268–273. // URL: https://cyberleninka.ru/article/n/formalizovannaya-model-sistemy-intellektualnoy-podderzhki-prinyatiya-resheniy-v-oblasti-zaschity-informatsii/viewer (дата обращения: 18.10.2023).

2. Васильченко А.Д. Система поддержки принятия решений для обеспечения информационной безопасности в облачной среде // Шаг в науку. 2020. №1. С. 4–8. // URL: https://cyberleninka.ru/article/n/sistema-podderzhki-prinyatiya-resheniy-dlya-obespecheniya-informatsionnoy-bezopasnosti-v-oblachnoy-srede/viewer (дата обращения: 19.10.2023).

3. Шабанов Р.М., Микушин Н.А. Интеллектуальная информационная система поддержки принятия решений // Молодой исследователь Дона. 2019. № 4. С. 91–97. // URL: https://cyberleninka.ru/article/n/intellektualnaya-informatsionnaya-sistema-podderzhki-prinyatiya-reshenii (дата обращения: 21.10.2023).

4. Лабабиди М.Р., Кельчевская Н.Р. Система поддержки принятия решений (СППР) как инструмент принятия эффективных управленческих решений на промышленных предприятиях // Весенние дни науки: сборник докладов (Екатеринбург, 21–23 апреля 2022 года). 2022. С. 377–381. // URL: https://elar.urfu.ru/bitstream/10995/116873/1/978-5-91256-557-1_2022_070.pdf (дата обращения: 22.10.2023).

5. Селифанов В.В. и др. Метод оценивания рисков в системах принятия решений с учетом защиты информации // Вестник СибГУТИ. 2023. Т. 17. № 2. С. 84–92. URL: https://cyberleninka.ru/article/n/metod-otsenivaniya-riskov-v-sistemah-prinyatiya-resheniy-s-uchetom-zaschity-informatsii (дата обращения: 22.10.2023).

6. Короткова А.А., Бобылева С.В. Применение методов машинного обучения в области инженерно-технической защиты информации // Системный анализ в науке и образовании. 2023. № 2. С. 45–55. URL: https://sanse.uni-dubna.ru/index.php/sanse/article/download/578/532 (дата обращения: 25.10.2023).

7. Гончаренко Ю.Ю. Использование технологии блокчейн для проверки подлинности электронных документов и файлов // Вестник УрФО. Безопасность в информационной сфере. – 2023. – №1(47). – С. 98–101. URL: https://www.info-secur.ru/index.php/ojs/article/view/408/366 (дата обращения: 25.10.2023).

УДК 004.056

ПЕРСПЕКТИВНЫЕ МЕРЫ ПРОФИЛАКТИКИ И ПРЕДУПРЕЖДЕНИЯ АТАК НА ОБЪЕКТЫ КРИТИЧЕСКОЙ ИНФРАСТРУКТУРЫ РОССИЙСКОЙ ФЕДЕРАЦИИ

И.Ю. Куринная, К.В. Шнейдер, Е.В. Стойчина, Д.В. Шевченко, А.Б. Шабров
Научный руководитель: канд. юр. наук, доц. В.В. Челноков
Уральский федеральный университет имени первого Президента России Б.Н. Ельцина, г. Екатеринбург

В рамках статьи рассмотрены перспективные направления профилактики компьютерных инцидентов на объектах критической инфраструктуры и государственных информационных системах, совершенных лицами, находящимися под иностранным влиянием.

Ключевые слова: государственные информационные системы, информационная безопасность, лица, находящиеся под иностранным влиянием, объекты критической инфраструктуры.

Поднятые во время ежегодных форумов проблемы обеспечения информационной безопасности (SOC-форумы), прошедшие в г. Москве 15–16 ноября 2022 года на тему: «Практика реализации атак на кибернетическую инфраструктуру и построение мониторинга информационной безопасности» [1] и 14–15 ноября 2023 года на тему: «Кто, как и зачем атаковал корпоративные сети в России в 2023 году» [2] свидетельствуют о формировании новых вызовов и угроз объектам критической информационной инфраструктуры (КИИ) и государственным информационным системам (ГИС) нашей страны, появившихся на фоне беспрецедентного санкционного давления на производителей оборудования, а также кратное увеличение (исходя из статистических сведений за предыдущие периоды) противоправных действий лиц и организаций осуществляющих подобные атаки. Уход значительного количества профильных игроков (поставщиков программных и аппаратных средств, задействованных для обеспечения информационной безопасности), а также увеличение количества компьютерных атак на объекты критической информационной структуры осуществляемых преимущественно из-за рубежа, позволяют говорить о негативной (по отношению к России) системной работе, направленной, в том числе, на сдерживание потенциала развития нашей страны.

По данным журнала «Positive Technologies» (рис. 1) виды компьютерных атак по различным отраслям в разных странах в 2022 году составили: 30% – на государственные предприятия, 16% – на IT-компании и по 10 % –

энергетический, финансовый и промышленный сектора экономики, 24 % – иные виды деятельности [3].

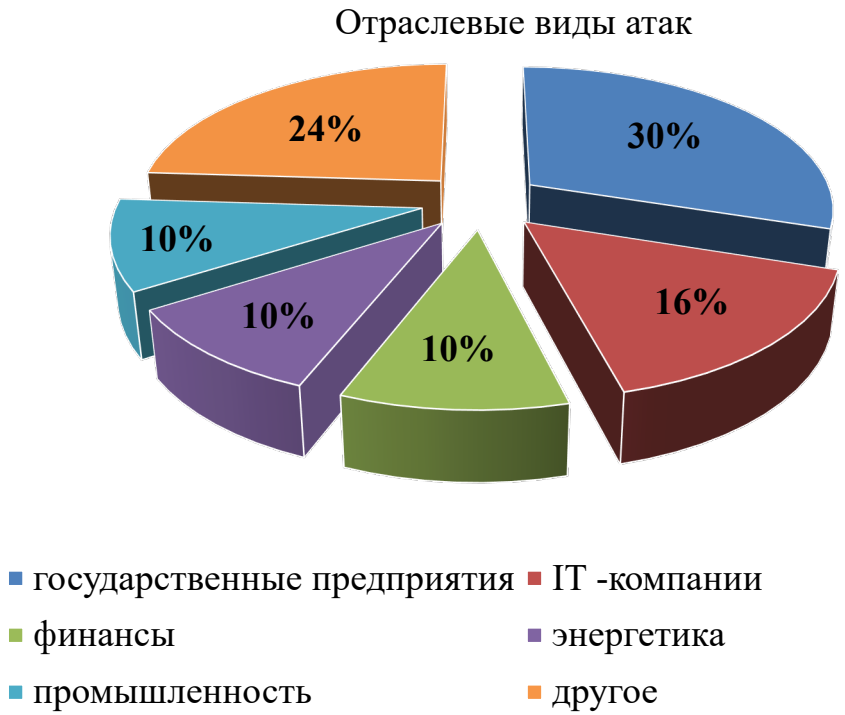


Рис. 1. Отраслевые виды атак

Вместе с тем, в соответствии со ст. 4 Федерального закона от 26 июля 2017 г. № 187-ФЗ «О безопасности КИИ РФ» [4] к важнейшим принципам реализации безопасности объектов КИИ отнесен приоритет предотвращения компьютерных атак. Результаты анализа сведений об атаках на российские объекты критической инфраструктуры и их характере свидетельствуют об их инициировании преимущественно из-за рубежа. Так, по данным StormWall [5] количество DDoS-атак в I полугодии 2023 года на российские компании увеличилось на 47% (в сравнении с аналогичным периодом 2022 года), наиболее подверженными отраслями явились финансовый сектор 32%, электронная коммерция 26%, развлечения 14%, телекоммуникация 10%, страхование 7%, логистика 7%, другие сферы 2%. По мнению специалистов компании, основная причина атак – «нестабильная политическая ситуация в мире и, частности в нашей стране», и как следствие, возникновение «хактивистов», цель которых – нанесение вреда различным сферам административной и хозяйственной деятельности Российской Федерации.

В качестве превентивной меры, направленной на предупреждение компьютерных атак на объекты КИИ и ГИС авторы статьи предлагают, помимо прочих, рассмотреть инструменты, предусмотренные в п.1 ст. 4 ФЗ от 14 июля 2022 года № 255-ФЗ «О контроле за деятельностью лиц, находящихся под иностранным влиянием» [6]. Так, фиксация и публикация гос-

органом сведений о лицах или организациях, осуществляющих противоправные действия против интересов Российской Федерации и находящихся под иностранным влиянием, будет иметь профилактическое воздействие и не позволит на основании 187-ФЗ «участвовать в эксплуатации значимых объектов КИИ, допустить их к обеспечению их безопасности указанных объектов». Кроме того, превентивная мера внесения в реестр «иностранных агентов» позволяет Роскомнадзору накладывать ограничения на доступ к информационным ресурсам, применяемым лицами, находящимися под влиянием из-за рубежа, а также в случае осуществления противоправных действий привлекать их к административной ответственности. За нарушение ограничений для указанной категории лиц предусмотрена административная ответственность в соответствии с п.п. 1-9 ст. 19.34 КоАП РФ «Нарушение порядка деятельности» [7].

Поиск и анализ подобных противоправных проявлений, по мнению авторов статьи, возможен путем организации тесного межведомственного взаимодействия государственных органов с российскими юридическими и физическими лицами, отвечающими за эксплуатацию объектов КИИ и ГИС.

Вместе с тем, законодательством не ограничено право лиц, находящихся под иностранным влиянием, осуществлять эксплуатацию и деятельность по обеспечению безопасности государственных ИС, не относящихся к значимым объектам КИИ, хотя она также сопряжена с целенаправленным сбором информации в области военно-технической деятельности Российской Федерации, которые в случае их получения иностранными акторами потенциально будут использоваться против безопасности нашей страны.

Вероятно, в этом случае имеет место законодательный пробел, который несложно устранить, добавив в ч. 17 ст. 11 ФЗ-255 помимо значимых объектов критической информационной инфраструктуры ограничение на эксплуатацию ГИС, а также обеспечение их безопасности.

Библиографический список

1. SOC-Форум 2022: итоги по версии «Ростелекома» [Электронный ресурс]: пресс-релиз – М.: Digital Russia, 2022. – URL://d-russia.ru/soc-forum-2022-itogi-po-versii-rostelekoma/ (дата обращения 20.10.2023).

2. Скулкин О. Кто, как и зачем атаковал корпоративные сети в России в 2023 году [Электронный ресурс]: – М.: SOCForum2023, 2023. – URL://forumsoc.ru/ reports (дата обращения 24.11.2023).

3. Новиков А.В. Кто и как атакует российские организации [Электронный ресурс]: статья/ А.В. Новиков – М.: Positive research, 2023. – 238 с. – URL: //ptsecurity.com/ru-ru/research/analytics/positive-research-2023/ (дата обращения 22.10.2023).

4. О безопасности критической информационной инфраструктуры Российской Федерации [Электронный ресурс]: федер. закон от 26 июля 2017 г. № 187-

ФЗ (с изм. и доп.). «КонсультантПлюс» URL://consultant.ru/document/cons_doc_LAW_220885/ (дата обращения 22.10.2023).

5. Обзор первого квартала 2023 года: отчет StormWall о DDoS-атаках. [Электронный ресурс]: – М.: StormWall, 2023. – URL://stormwall.pro/otchet-o-ddos-atakah-stormwall-pervyj-kvartal-2023 (дата обращения 24.11.2023).

6. О контроле за деятельностью лиц, находящихся под иностранным влиянием [Электронный ресурс]: федер. закон от 14 июля 2022 года № 255-ФЗ (с изм. и доп.). «КонсультантПлюс» – URL://consultant.ru/cons_doc_LAW_421788_(дата обращения 22.10.2023).

7. Кодекс Российской Федерации об административных правонарушениях [Электронный ресурс]: федер. закон от 30.12.2001 № 195-ФЗ (ред. от 27.01.2023). Доступ из справ. – правовой системы «КонсультантПлюс» – URL: //consultant.ru/document/cons_doc_LAW_34661 (дата обращения 22.10.2023).

СЕКЦИЯ «ПРИКЛАДНЫЕ И НАУЧНЫЕ ИССЛЕДОВАНИЯ В ОБЛАСТИ ЗАЩИТЫ ИНФОРМАЦИИ ОТ УТЕЧЕК ПО ТЕХНИЧЕСКИМ КАНАЛАМ»

УДК 004.056

СПОСОБЫ ПОДАВЛЕНИЯ РАДИОЗАКЛАДНЫХ УСТРОЙСТВ

И.И. Баранкова, У.В. Кузьмина, А.Р. Федорова, Ю.Я. Кульевич
Научные руководители: д.т.н., зав. каф. ИиИБ И.И. Баранкова, к.т.н., доц. каф. ИиИБ У.В. Кузьмина
Магнитогорский государственный технический университет имени Г.И. Носова, г. Магнитогорск

В данной статье рассматриваются радиозакладные и радиоподавляющее устройства, две технологии, связанные с передачей и блокированием радиосигналов. В статье описывается радиозакладное устройство, работающее на частотах в диапазоне 88-108 МГц, с использованием биполярного транзистора и электретного микрофона для амплитудной и частотной модуляции звуковых сигналов. Также приводится информация о радиоподавляющем устройстве, способном создавать мощные помехи для подавления радиосигналов. Графики и технические детали демонстрируют разницу в мощности между сигналами радиоподавляющего и радиозакладного устройства, что делает последнее недоступным для прослушивания. Эта статья предоставляет важные сведения о технологиях передачи и блокирования радиосигналов, их работе и потенциальных областях применения.

Ключевые слова: подавление радиосигналов, радиозакладное устройство, радиоподавляющее устройство, радиопомехи.

В современном мире технологий и связи радиосигналы играют ключевую роль в передаче информации. Они служат основой для беспроводной связи, радиовещания и множества других приложений. Однако, существует не только необходимость в эффективной передаче радиосигналов, но и в их блокировании, когда это необходимо. В данной статье мы рассмотрим два аспекта радиосигналов: создание радиозакладного устройства (РЗУ), которое генерирует сигнал для передачи звуковой информации, и радиоподавляющего устройства (РПУ), способного заглушать другие радиосигналы [1].

Для создания устройства, способного генерировать радиосигналы в диапазоне 88-108 МГц, мы использовали LC-генератор, построенный на основе биполярного транзистора 2N3904 и схемы «ёмкостной, трёхточки». Основной параметр этого генератора – его частота, которая определяется параметрами конденсатора и катушки индуктивности в колебательном

контуре. Данный генератор имеет возможность точной настройки частоты с помощью переменной ёмкости конденсатора.

Для добавления звуковой информации в создаваемый сигнал мы использовали электретный микрофон. Он преобразует звуковой сигнал в электрический, который затем подается на базу транзистора через резистор и конденсатор. В результате этого процесса ток через транзистор изменяется, что приводит к изменению как частоты, так и амплитуды генерируемого сигнала. Таким образом, мы достигли амплитудной и частотной модуляции сигнала звуковой информацией.

Резистор и конденсатор, соединенные последовательно, выполняют роль RC-фильтра, который позволяет снизить уровень шума, поступающего с микрофона. Резистор также ограничивает ток, поступающий на транзистор, обеспечивая его стабильную работу. Конденсатор, подключенный параллельно, способствует сглаживанию импульсов от источника питания, что также способствует повышению качества сигнала [2-3].

Принципиальная схема РЗУ на одном транзисторе представлена на рис. 1.

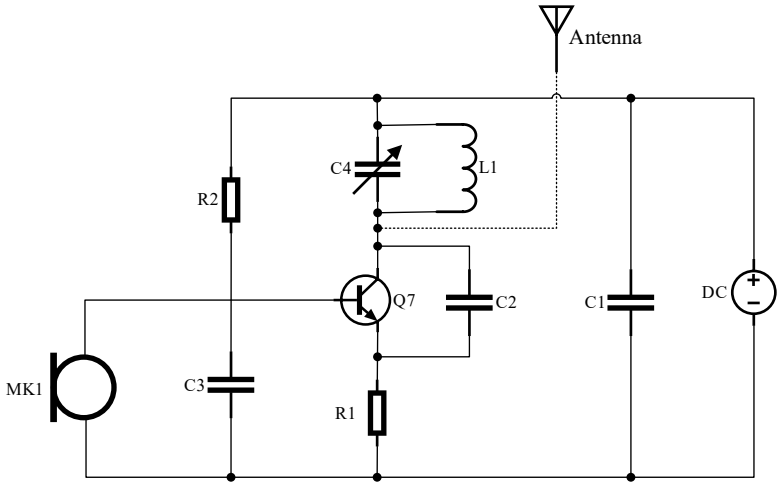


Рис. 1. Принципиальная схема РЗУ на одном транзисторе

Все элементы, кроме выключателя и батареи питания, монтируются на макетной плате. Смонтированная плата РЗУ показана на рис. 2.

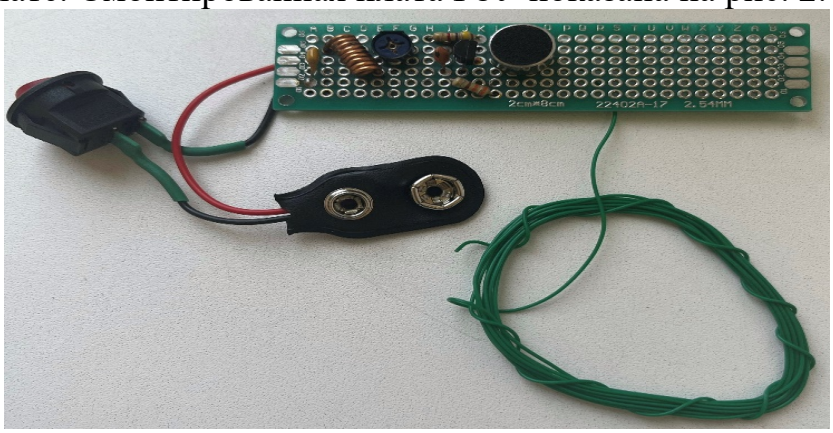


Рис. 2. Смонтированная плата РЗУ

РЗУ, собранное по схеме рисунка 1, должно вещать в FM диапазоне (88–108 МГц). Для начала нужно опередить на какой именно частоте вещает радиопередатчик, для этого воспользуемся программно-аппаратным комплексом радиоконтроля «Кассандра С6» [4].

Запустим сканирование радиочастот до включения РЗУ. Выставим диапазон сканирования 80–110 МГц и зафиксируем результат (рис. 3).

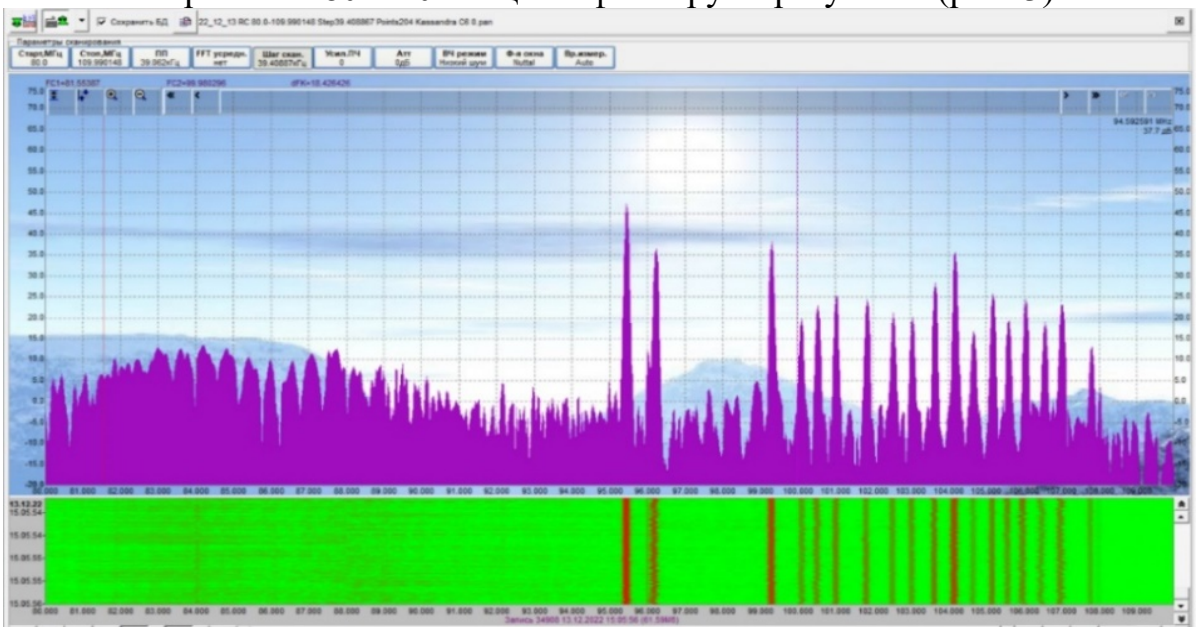


Рис. 3. Сканирование радиоэфира в диапазоне 80–110 МГц

Подключим схему к источнику питания, в нашем случае это 9 Вольтовая батарейка крона, и для точного определения частоты вещания уменьшим диапазон сканирования (80–98 МГц), посмотрим результат поиска (рис. 4).

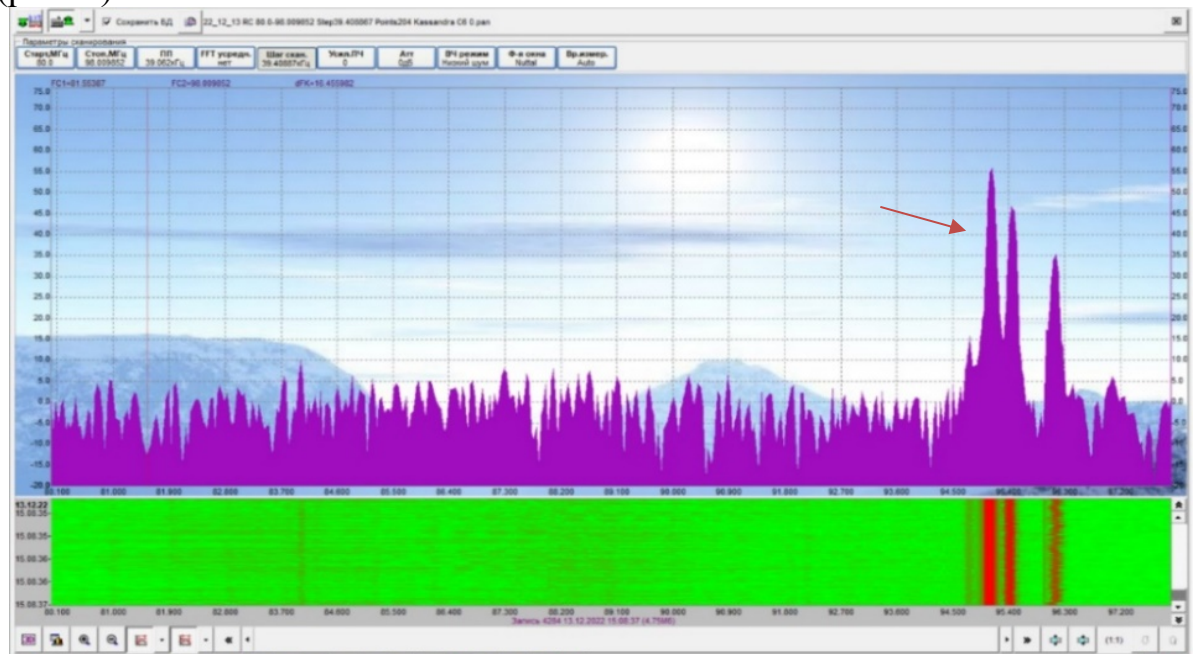


Рис. 4. Сканирование радиоэфира в диапазоне 80–98 МГц с включенным РЗУ

В процессе использования РЗУ, все аудиосигналы, которые попадают на микрофон, автоматически транслируются на предварительно установленную частоту, как показано на представленном выше графике сканирования.

Также по результатам сканирования, представленного на рис. 4, видно, что РЗУ передает информацию на частоте около 95 МГц. Радиус действия, равный 12 м, был определен экспериментальным путем. Иными словами, при отдалении РЗУ от приемника больше чем на 12 м мощность сигнала была сравнима с радиошумом.

В ряде ситуаций возникает потребность в подавлении нежелательных радиопередач, создаваемых РЗУ, которые мешают работе или вмешиваются в нее незаконно. Для этой цели существуют специальные устройства, называемые РПУ. Это устройство состоит из двух биполярных транзисторов типа 2CS3355, двух резисторов и двух конденсаторов, и функционирует как мультивибратор, определяя необходимую выходную частоту. В нагрузке для каждого транзистора используются катушки индуктивности. Помимо этого, схема включает отдельный резистор для обратной связи питания эмиттеров транзисторов.

Для правильной работы данной схемы требуется подключение двух антенн разной длины. Первая антенна имеет длину 14 см, в то время как вторая антенна – 45 см. Катушки индуктивности в данной схеме состоят из 7 витков провода диаметром 0,8 мм, намотанных на оправке диаметром 4 мм. Конденсаторы и резисторы выполняют функцию сглаживания генерируемых импульсов, что способствует стабильности работы устройства [5, 6].

Принципиальная схема РПУ на двух транзисторах представлена на рис. 5.

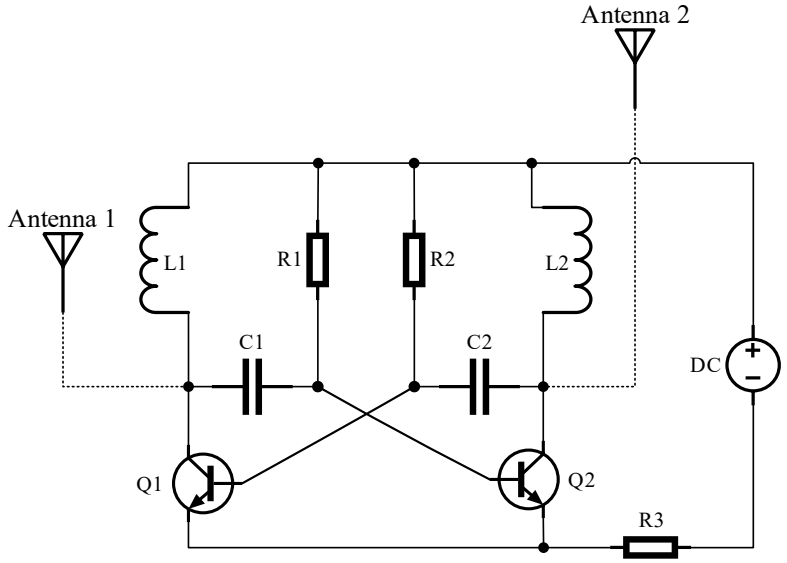


Рис. 5. Принципиальная схема РПУ на двух транзисторах

Все элементы монтировались на весу. Смонтированное РПУ на двух транзисторах показана на рис. 6.

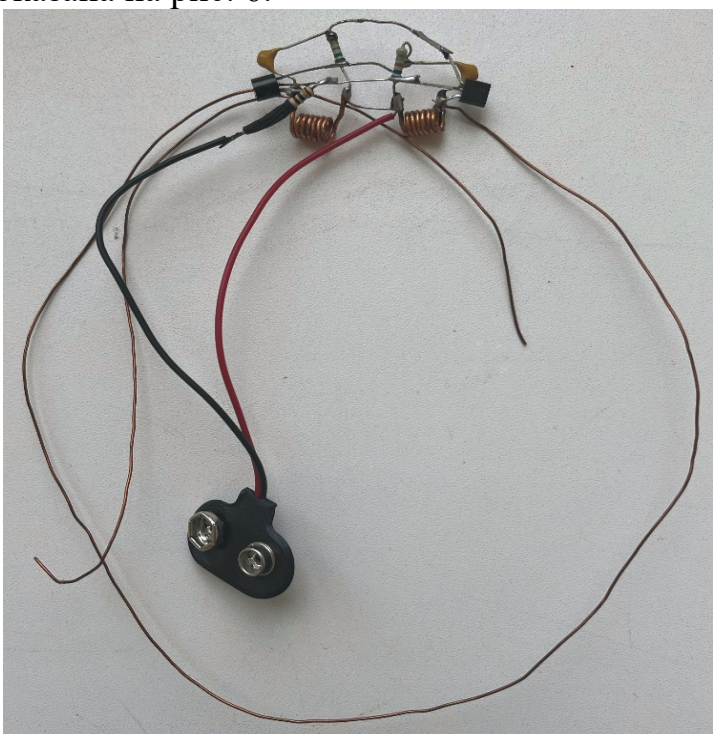


Рис. 6. Смонтированное РПУ на двух транзисторах

Определим диапазон работы схемы, для этого подключим антенну 0–600 МГц, подключим питание к устройству и запустим сканирование (рис. 7).

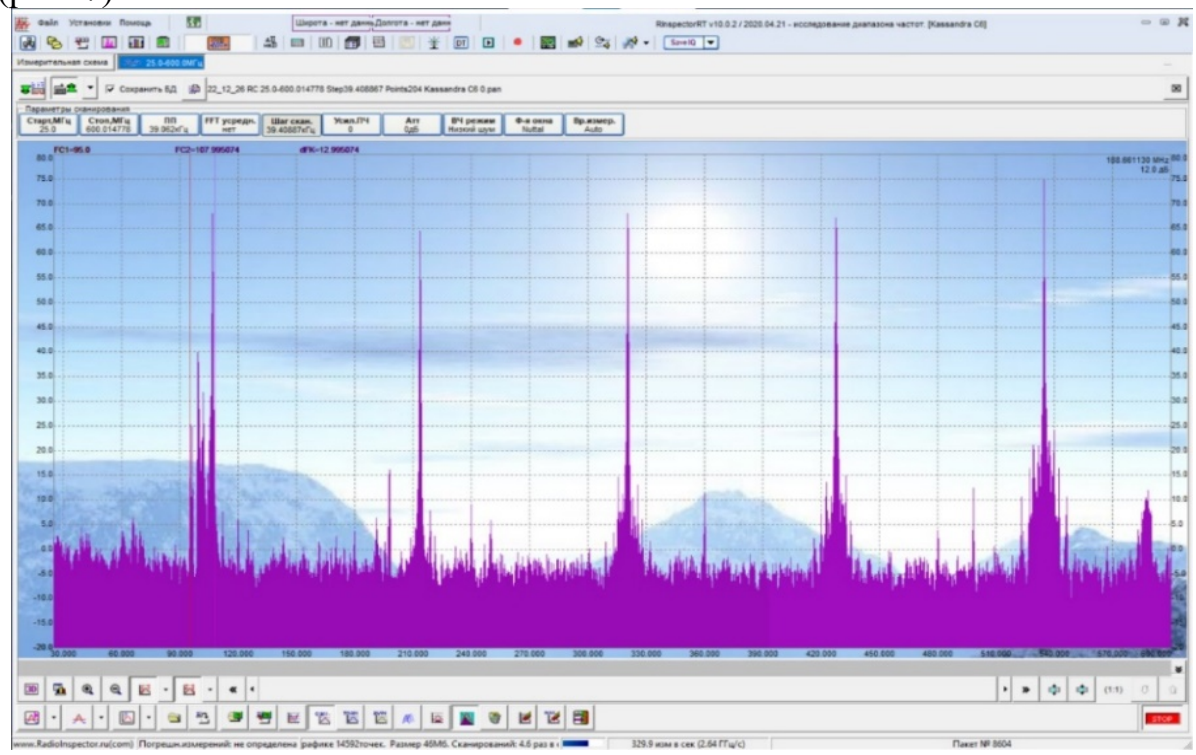


Рис. 7. Сканирование радиоэфира 25–600 МГц

При использовании РПУ возникают радиопомехи, которые начинают перекрывать и заглушать сигналы, передаваемые на определенных радиочастотах. Устройство для радиоподавления обладает более мощным сигналом, что позволяет успешно подавлять более слабые радиопередачи, такие как РЗУ. Представленный выше график показывает, что мощность сигнала от РПУ существенно превышает мощность сигнала от РЗУ. Из-за этого, при попытке прослушивания частоты, на которой работает РЗУ, пользователь слышит только шум и помехи, не воспринимая информативный сигнал.

Также по результатам сканирования, представленного на рис. 7, видно, что устройство генерирует сигналы через каждые 100–110 МГц. Подключим антенну способную принимать сигнал в диапазоне 600–6000 МГц, посмотрим на результат (рис. 8).

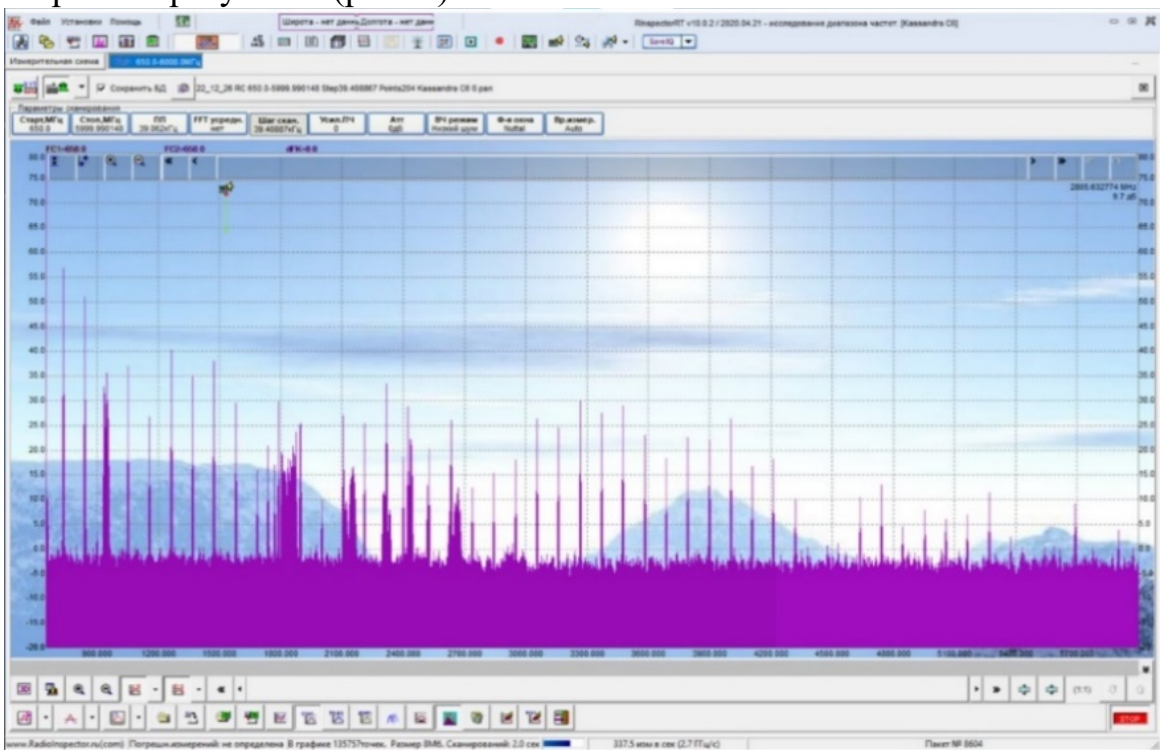


Рис. 8. Сканирование радиоэфира 650–6000 МГц

Диапазон работы данного устройства составляет примерно 4000 МГц, основная мощность приходит на первые 1000 МГц, далее постепенно уменьшается. Дальность работы сигнала составляет до 15 метров.

В рамках данной работы были рассмотрены и проанализированы два важных аспекта радиосвязи: создание РЗУ и использование РПУ. Эти технологии представляют интерес в контексте радиочастотных систем и имеют широкий спектр применения в различных сферах, включая средства связи, безопасность информации и научные исследования.

РЗУ, реализованное на основе биполярного транзистора и электретного микрофона, представляет собой эффективное средство для передачи

аудиосигналов на заданных частотах в радиодиапазоне. Амплитудная и частотная модуляция, в сочетании с использованием конденсаторов и резисторов, обеспечивают стабильность и качество передачи звуковой информации.

РПУ, с другой стороны, представляют собой средство для блокирования радиосигналов. Их способность создавать мощные помехи позволяет подавлять другие радиопередачи, что находит применение в обеспечении безопасности и конфиденциальности данных. РПУ также используются для борьбы с нелегальными радиопередачами и предотвращения нежелательных радиосигналов.

Таким образом, данная статья подчеркивает актуальность и важность технологий радиосвязи и блокирования радиосигналов в современном информационном обществе. Понимание их работы и принципов функционирования существенно для обеспечения эффективности и безопасности радиочастотных систем.

Библиографический список

1. Бузов Г.А. Защита информации ограниченного доступа от утечки по техническим каналам / Бузов Геннадий Алексеевич. – М.: Горячая линия – Телеком, 2016. – 909 с.

2. Малогабаритная радиоаппаратура. Вопросы конструирования, производства и эксплуатации. – М.: Издательство иностранной литературы, 2014. – 372 с.

3. Горшелев В.Д. Основы проектирования радиоприемников / В.Д. Горшелев, З.Г. Красноцветова, Б.Ф. Федорцов. – Москва: СИНТЕГ, 2015. – 384 с.

4. Михайлова У.В., Фаткуллин А.Р. Использование возможностей комплекса радиомониторинга «Кассандра» для обнаружения современных технических средств с передачей информации по радиоканалу// Безопасность информационного пространства. Сборник трудов XVIII Всероссийской научно-практической конференции студентов, аспирантов и молодых ученых. 2019. С. 124–127.

5. Иванов-Цыганов А.И. Электротехнические устройства радиосистем / А.И. Иванов-Цыганов. – М.: Высшая школа, 2019. – 490 с.

6. Фаткуллин А.Р., Михайлова У.В. Современные средства радиоразведки// Актуальные проблемы современной науки, техники и образования. Тезисы докладов 79-й международной научно-технической конференции. 2021. С. 402.

УДК 004.056

БПЛА – НОВАЯ УГРОЗА БЕЗОПАСНОСТИ ИНФОРМАЦИИ

Н.А. Байтяков, К.Н. Гуральский, К.Л. Костюченко
Научный руководитель: канд. техн. наук, доц. К. Л. Костюченко
Уральский государственный университет путей сообщения,
г. Екатеринбург

В статье показана актуальность развития беспилотных летательных аппаратов. Описывается характеристики и сферы применения БПЛА. Рассмотрены основные проблемы, связанные с их криминальным использованием. Представлены, появляющиеся, в связи с этим, угрозы безопасности информации.

Ключевые слова: антидроновые системы безопасности, беспилотный летательный аппарат, дрон, защита информации, информационная безопасность, радиомикрофон, технические каналы утечки информации, угрозы безопасности информации.

Целью данной статьи является представление новой угрозы безопасности информации, связанной с применением беспилотных летательных аппаратов (БПЛА) в качестве элементов каналов утечки информации, а также средств несанкционированного воздействия на объекты информатизации.

Для достижения поставленной цели рассматривается классификация БПЛА, области их применения, тенденции в конструировании. Отдельно определяются возможности криминального применения БПЛА в информационной сфере, а также способам и средствам противодействия этому.

БПЛА используются в деятельности человека достаточно давно. Они имеют свою историю, обширную классификацию, различные типы конструкций с соответствующими функциями и областями применения [1].

Развитие техники, электроники, энергетики, информационных технологий и программирования в начале XXI века позволило определить некоторые тенденции в создании БПЛА: постоянно повышается энерговооруженность; происходит миниатюризация элементов, происходит универсализация компоновки аппаратов, появляются новые конструкционные решения, увеличивается время и дальность полетов при одновременном улучшении точности.

Названные тенденции обусловили широту гражданского и военного применения БПЛА. В гражданской сфере это: контроль различных объектов инфраструктуры (сооружения, дороги, ЛЭП, трубопроводы, теплосети и т.п.), научные исследования и мониторинг (климат, экология, геологоразведка, археология), осмотр местности (охрана, спасательные работы и т.п.), фото- и видеосъемка (репортажи, папарацци), доставка грузов и еды (курьерская служба, аэротакси), распыление химикатов (вызов дождей,

сельское хозяйство), использование в системах телекоммуникации, визуальные эффекты (реклама, шоу). В военной сфере это: ведение разведки в реальном масштабе времени, целеуказание, гранатометание, бомбометание, нанесение ударов («камикадзе»), создание ложных целей, радиоэлектронная борьба, охрана и патрулирование, ретрансляция данных, доставка боеприпасов и медикаментов [2].

Причем подавляющую часть применяемого арсенала аппаратов составляют мультироторные беспилотные летательные аппараты – коптеры (в основном квадрокоптеры), оснащенные камерой и передающие видео в реальном времени на устройство пилота. За ними закрепилось название FPV-дроны (First Person View – в переводе с английского – вид от первого лица). Управляющий ими оператор пользуется очками виртуальной реальности (VR-очки) и обычным телевизионным каналом. Действительно, благодаря своим высоким характеристикам БПЛА вертолетного типа практически вытеснили остальные (например, самолетного типа) [3].

Массовое применение БПЛА, а, следовательно, и массовое производство привело к их удешевлению (например, цена FPV-дронов начинается с нескольких десятков тысяч рублей, а в категории легких дронов – с полутора тысяч). Это способствовало вовлечению населения (прежде всего молодежи) в новое увлечение (развлечение) – в конструирование и программирование дронов, а также в участие в различных конкурсах и соревнованиях.

Ситуация изменилось с началом Специальной военной операции, которая резко подстегнула разработку беспилотников, поскольку стало очевидно, что FPV-дроны прочно заняли нишу основного оружия на оперативно-тактическом уровне, БПЛА вообще стали одним из самых эффективных и дешевых вооружений.

Сегодня в России созданием и производством FPV-дронов занимаются не только крупные производственные компании и конструкторские бюро, но и многочисленные изобретатели-энтузиасты, налаживающие выпуск дронов-камикадзе практически «на коленке». Многие разработки сразу проходят проверку в боевых условиях. Такие дроны или подтверждают свои высокие качества, или отправляются обратно для внесения необходимых корректировок, учитывающих пожелания военных.

Появилась новая специальность – оператор FPV-дронов. А с 1 января 2024 года начнет свою работу национальный проект по развитию беспилотных авиационных систем.

Следует отметить, что кроме позитива, связанного с прогрессом в сфере БПЛА, существует и значительный негатив, состоящий в криминальном использовании дронов. Большое количество произведенных аппаратов, определенная часть которых является неучтенными; низкая заметность; постоянное совершенствование конструкции и используемых материалов;

расширение возможностей; отработанные тактики применения; относительная легкость управления; значительное число обученных операторов; отсутствие некоторых правовых норм не могут не привести к ситуациям, связанных с нарушением закона (подобных незаконному обороту оружия, наркотических средств, алкоголя и т.п.).

Чтобы использовать дроны в криминальных целях, некоторые из них не придется даже переделывать. Прежде всего, это касается конструкций «двойного назначения», то есть тех, которые могут использоваться в гражданской и военной сфере (рис. 1). Определенная часть БПЛА дорабатывается, а часть выполняется специально под криминальные задачи.

Рис. 1. Структура применения БПЛА

Такие задачи террористического направления известны [2]: доступ за периметр охраняемых объектов и ведение там наблюдения; точечное уничтожение отдельных важных лиц; заброска самодельных средств поражения; нанесение повреждений зданиям, памятникам культуры, объектам инфраструктуры и транспортным средствам; транспортировка запрещенных средств или их заброска на охраняемую территорию; распыление токсических веществ; препятствование воздушному движению в аэропортах и т.д.

Определенный набор задач криминального информационного направления очевиден (наблюдение, фото- и видеосъемка), а остальные только начинают формулироваться (либо просто нет никаких сведений ввиду высокой степени закрытости таких сфер как шпионаж, псевдошпионаж, промышленный шпионаж).

Несмотря на то, что в специальной литературе по защите информации и информационной безопасности, даже наиболее свежей [4], нет упоминания про БПЛА, все же можно составить примерный перечень появляющихся угроз безопасности информации.

Фото- и видеосъемка в контролируемой зоне (КЗ) объекта информатизации. БПЛА, в первую очередь малые, ввиду слабой заметности могут достаточно просто проникать в КЗ. Более тяжелые и оснащенные сильной оптикой способны делать снимки на подлете к КЗ.

Доставка различных устройств съема информации в зоны, где доступ человека затруднен, а потому и не включенные ранее в КЗ.

Доставка радиомикрофонов (и иных устройств съема информации) в КЗ. Возможен вариант «камикадзе», когда аппарат не возвращается и используется как источник питания.

Доставка в КЗ (или к КЗ) другого дрона, способного самостоятельно перемещаться по поверхности до необходимой точки.

Доставка в КЗ устройств, способных внедряться в беспроводные сети либо нарушать их работу (подавители, блокираторы).

Ретрансляция сигнала от радиомикрофона, находящегося в КЗ. При этом сам радиомикрофон может быть выполнен менее мощным, из-за чего поиск будет затруднен.

Сеансовый прием записей от радиомикрофона с диктофоном при периодических подлетах в КЗ.

Пролет в КЗ с целью изъятия документа (промышленного образца).

Перечисленные угрозы становятся еще более актуальными из-за сложности обнаружения и нейтрализации БПЛА.

БПЛА может обнаруживаться при помощи оптической, акустической, радарной и радиочастотной детекции. А противодействие может осуществляться огневым поражением, радиоэлектронным подавлением систем навигации и радиосвязи, функциональным поражением сверхвысокочастотным и лазерным излучениями, физическим перехватом (с использованием специальных БПЛА-перехватчиков, горючих аэрозолей, сетей, специальных клейких и вязких аэрозолей, специально тренированных птиц) [5–7].

Поскольку большинство из указанных способов больше подходит для военной сферы, то для гражданской разрабатываются системы с модулями направленной генерации помех, основанные на искусственном интеллекте и нейронных сетях, например, программно-аппаратный комплекс Kaspersky Antidrone [8].

Указанные примеры использования БПЛА в большинстве технических каналов утечки информации позволяют сделать ряд выводов.

БПЛА в определенных ситуациях становятся серьезной угрозой безопасности информации.

Уровень риска реализации таких угроз будет постоянно возрастать, а количественные показатели можно рассчитать по появляющимся новым методикам [9].

Способы и средства противодействия неправомерному использованию БПЛА в информационной сфере находятся на начальной стадии и требуют дальнейшего развития.

Библиографический список

1. Беспилотная авиация: терминология, классификация, современное состояние / В.С. Фетисов, Л.М. Неугодникова, В.В. Адамовский, Р.А. Красноперов // Под ред. В.С. Фетисова. – Уфа: ФОТОН, 2014. – 217 с.

2. Макаренко С.И., Тимошенко А.В., Васильченко А.С. Анализ средств и способов противодействия беспилотным летательным аппаратам. Часть 1. Беспилотный летательный аппарат как объект обнаружения и поражения // Системы управления, связи и безопасности. 2020. № 1. С. 109−146.

3. Меринов К. FPV-дроны: технология, тактическое применение и будущее // Деловая газета «Взгляд». 15 сентября 2023 г. // URL: https://vz.ru/information/2023/9/15/1230301.html (дата обращения: 25.10.2023).

4. Новиков В.К., Краснов М.Г., Рекунков И.С. Средства, технологии, системы и технические каналы утечки информации для осуществления киберслежки за человеком и его деятельностью. – М.: Горячая линия – Телеком, 2023. – 160 с.

5. Макаренко С.И., Тимошенко А.В. Анализ средств и способов противодействия беспилотным летательным аппаратам. Часть 2. Огневое поражение и физический перехват // Системы управления, связи и безопасности. 2020. № 1. С. 147−197.

6. Макаренко С.И. Анализ средств и способов противодействия беспилотным летательным аппаратам. Часть 3. Радиоэлектронное подавление систем навигации и радиосвязи // Системы управления, связи и безопасности. 2020. № 2. С. 101−175.

7. Макаренко С.И. Анализ средств и способов противодействия беспилотным летательным аппаратам. Часть 4. Функциональное поражение сверхвысокочастотным и лазерным излучениями // Системы управления, связи и безопасности. 2020. № 3. С. 122−157.

8. Kaspersky Antidrone – экосистема мониторинга и защиты от дронов // Лаборатория Касперского // URL: https:// antidrone.kaspersky.com/ru/ (дата обращения: 25.10.2023).

9. Повышев А.А., Соколов А.Н., Мищенко Е.Ю. Универсальная классификация угроз безопасности информации и её применение для разработки модели угроз и оценки рисков // Вестник УрФО. 2023. № 3(49). С. 68–80.

УДК 621.396.67.012.12

ПРИМЕНЕНИЕ АДАПТИВНОЙ ФАЗИРОВАННОЙ АНТЕННОЙ РЕШЕТКИ ДЛЯ ЗАЩИТЫ ДАННЫХ ПРИ ИСПОЛЬЗОВАНИИ ОТКРЫТОГО КАНАЛА СВЯЗИ

Р.И. Баимов, А.Н. Рагозин
Научный руководитель: канд. техн. наук, доц. А.Н. Рагозин
Южно-Уральский государственный университет,
г. Челябинск

В статье представлено использование адаптивных фазированных антенных решеток (АФАР), как метод повышения помехоустойчивости беспроводных сетей передачи и защиты данных по открытому каналу связи. Естественные помехи и постановщики помех представляют угрозу целостности данных, передаваемых по открытому каналу связи. Предлагаемая антенная система с адаптивной фазированной решеткой использует интеллектуальные методы формирования диаграммы направленности (ДН) антенны, имеющей узкий луч в направлении информационного источника сигнала и глубокие нули ДН в направлении постановщиков помех, также низкий фон боковых лепестков ДН, снижающих воздействие естественных помех с произвольных направлений. Низкий фон боковых лепестков ДН задаётся подобранными по критериям оптимальности весовыми коэффициентами распределения на элементах АФАР.

Ключевые слова: адаптивная фазированная антенная решетка, амплитудное весовое распределение, вектор весовых коэффициентов, разностная диаграмма направленности, суммарная диаграмма направленности, открытый канал связи, помехоустойчивость.

В условиях быстрого роста сетей беспроводной связи, технологии IoT обеспечение безопасности данных стало критически важной задачей. Данные открытого канала связи подвергаются воздействию помех от различных источников, таких как пользователи соседнего и совмещенного каналов, атмосферные условия и внешние электромагнитные сигналы, внешние постановщики преднамеренных помех. Эти помехи могут повредить пакеты данных, ухудшить качество сигнала и обеспечить несвоевременный прием данных [1].

Существуют несколько методов защиты от воздействия радиопомех: организационный, энергетический, сигнальный и пространственный.

Организационный метод основывается на достижение необходимого уровня электромагнитной совместимости между источниками радиоизлуче-

ния (ИРИ). Данный метод исчерпал своё применение в условиях большой насыщенности ИРИ и ограниченности радиочастотного диапазона.

Энергетический метод борьбы с помехами предусматривает увеличение мощности передатчика до уровня, который гарантированно подавляет возможные помехи. Недостатком данного метода является повышение затрат энергоресурсов передатчика (ИРИ).

Сигнальный метод борьбы с помехами основывается на фильтрации сигнала (ограничении полосы пропускания). Это предполагает использование фильтров для удаления нежелательных частот или шума из сигнала. Однако недостатком этого метода является то, что он также может ослаблять или удалять полезные компоненты сигнала, если они попадают в пределы фильтруемой полосы пропускания. Это может привести к потере информации или ухудшению качества сигнала. К сигнальному методу также можно отнести технологии помехоустойчивого кодирования.

Перспективным методом защиты от помех является пространственная обработка сигналов с помощью адаптивных фазированных антенных решеток (АФАР). АФАР – это антенная система, которая включает в себя массив из нескольких антенн и алгоритма адаптивной обработки сигналов. Метод пространственной обработки автоматически формирует диаграмму направленности АФАР для улучшения приема информационного сигнала на фоне различных помех (рис. 1).

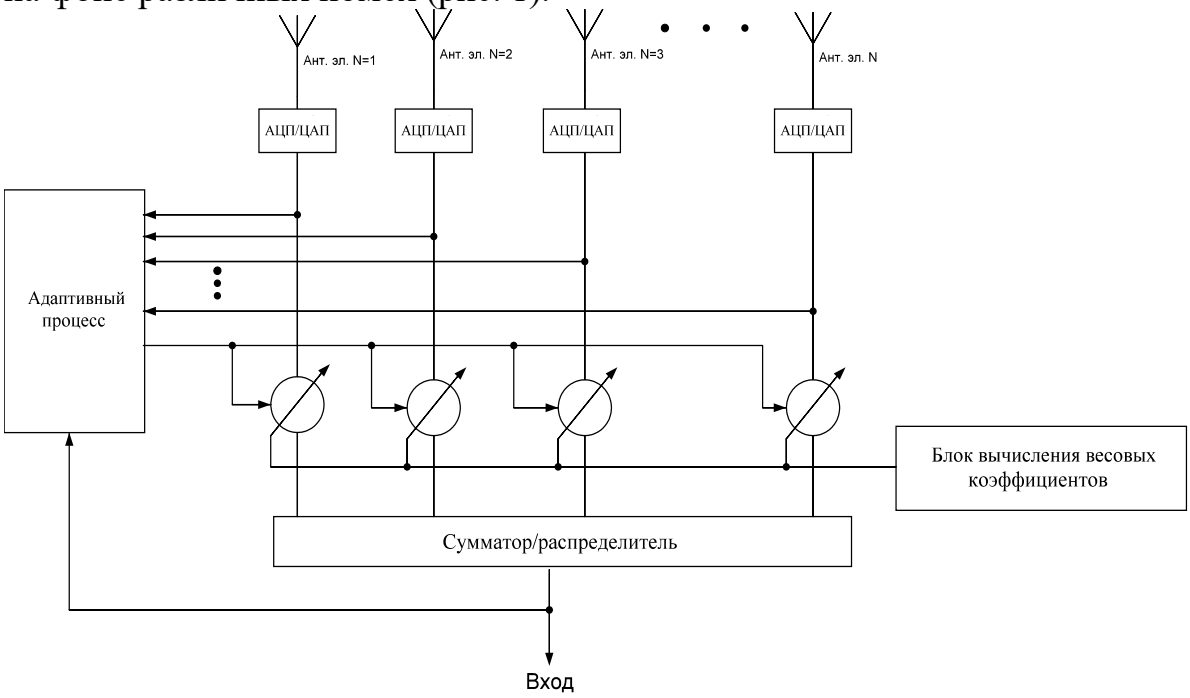


Рис. 1. Функциональная схема АФАР

Использование адаптивных фазированных антенных решеток (АФАР) позволяет создавать желаемые диаграммы направленности (ДН), тем самым максимизировать направленность в сторону полезного сигнала, одновременно сводя к минимуму помехи с других направлений в реальном времени

[2-5]. Это достигается за счет адаптивных методов обработки сигналов, которые вычисляют векторы весовых коэффициентов (ВВК) на элементах АФАР. Изменение ВВК от исходного состояния до оптимального ($w_{опт}$) зависит от выбранного критерия оптимальности. [4]

Выбор критерия оптимальности тесно связан с показателем, количественно измеряющим качество приема полезного сигнала на фоне помех. Целевая функция $J(w)$ описывает, как изменяется показатель качества в зависимости от значений весовых коэффициентов. Экстремум целевой функции представляет собой критерий оптимальности, которого можно достичь за счет оптимизации функции весовых коэффициентов.

В табл. 1 определены оптимальные ВВК ($w_{опт}$) в зависимости от критерия оптимальности.

Таблица 1

ВВК $w_{опт}$ по критериям оптимальности

Критерий оптимальности	$w_{опт}$
Оптимальное с общей шириной лепестков	$w_{опт} = \dfrac{\sin\left(\pi \mathrm{n} / \mathrm{N}\right)}{\left(\pi \mathrm{n} / \mathrm{N}\right)}$
Оптимальное с минимальной боковой лепестковой энергией	$w_{опт} = exp\left\{-\dfrac{[sin(\pi n / N)]^2}{2\sigma^2}\right\}$
Оптимальное с минимальной суммой квадратов пьедесталов	$w_{опт} = \left\{1, \textit{если } \mathrm{n} = 0; \dfrac{\sin\left[\left(\pi \mathrm{n} / \mathrm{N}\right)/2\right]}{\left[\left(\pi \mathrm{n} / \mathrm{N}\right)/2\right]}, \textit{если } \mathrm{n} > 0\right\}$
Оптимальное с минимальной площадью под кривой диаграммы направленности	$w_{опт} = \sqrt{\dfrac{2}{N}} \cdot \cos\left(\dfrac{\pi n}{N}\right)$
Оптимальное с минимальным количеством несовпадений фаз	$w_{опт} = \cos\left(\dfrac{\pi \mathrm{n}}{\mathrm{N}}\right) + \mathrm{j} \cdot \sin\left(\dfrac{\pi \mathrm{n}}{\mathrm{N}}\right)$
Оптимальное с наименьшей мощностью на пьедестале	$w_{опт} = \sqrt{\dfrac{2}{N}} \cdot \cos\left(\dfrac{\pi \mathrm{n}}{\mathrm{N}}\right) \cdot \exp\left\{-\mathrm{j} \cdot \sin\left(\dfrac{\pi \mathrm{n}}{\mathrm{N}}\right)\right\}$

На рис. 2 приведены результаты расчётов ДН удовлетворяющих критериям адаптивного метода пеленгования на элементах линейной ФАР (ЛФАР). Угол направления полезного сигнала (максимума) равен 10°. Углы направления мешающих сигналов (минимумов) равны 30° и -20°.

Формула для расчета СДН адаптивного метода пеленгования:

$$СДН = S_1 - \frac{S_1 S_2^H S_2}{N}, \qquad (1)$$

где S_1 – полезный сигнал; S_2 – мешающий сигнал.

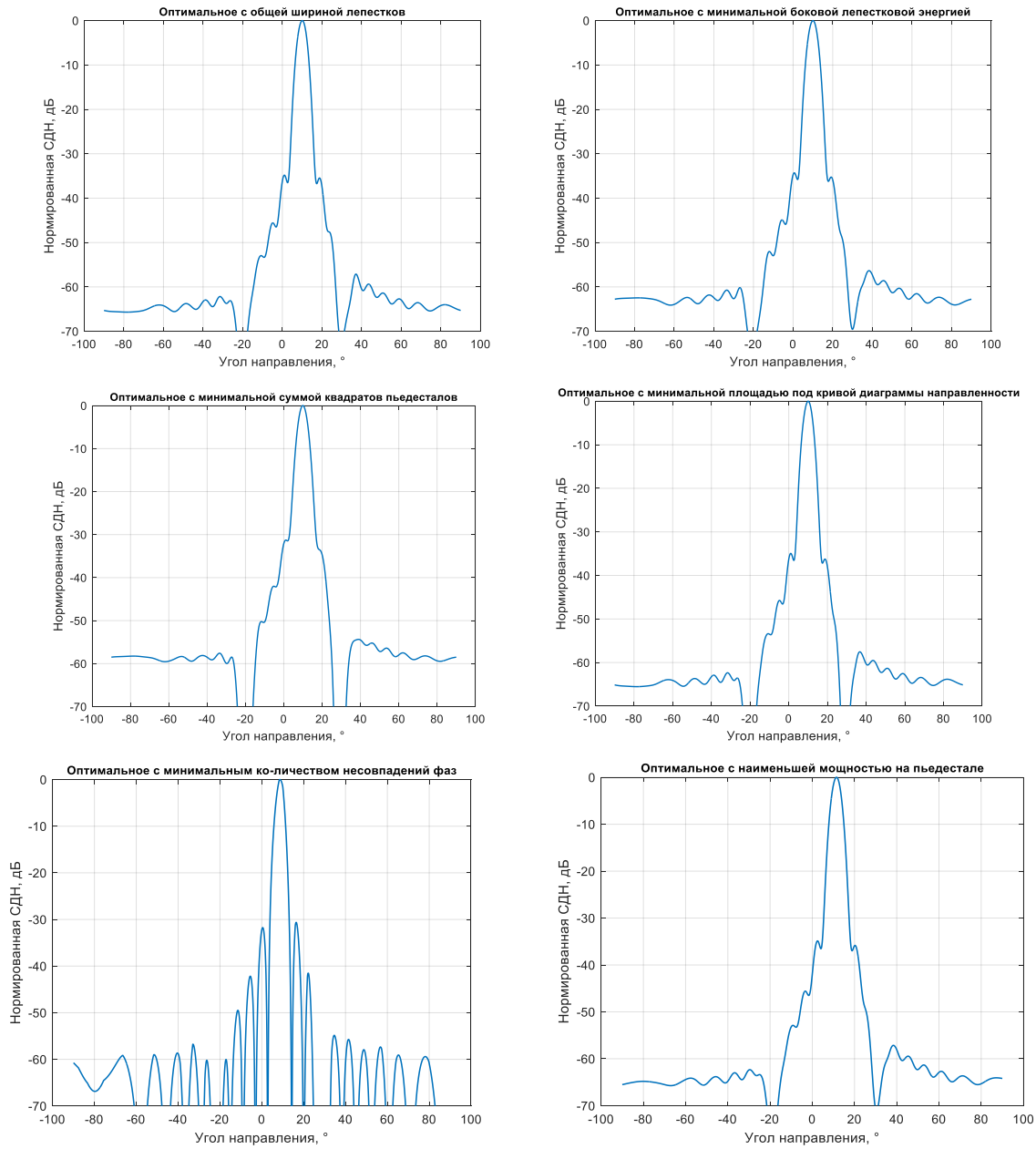


Рис. 2. Результаты расчетов ДН АФАР

Выбор амплитудного распределения, удовлетворяющего моноимпульсному методу пеленгования.

В работе для определения направления из точки приёма на информативный источник сигнала используется моноимпульсный метод пеленгования. Определённое направление на информативный источник с использованием моноимпульсного метода позволяет сформировать узкий луч АФАР в направлении информативного источника сигнала. При реализации моноимпульсного метода пеленгования формируются также суммарная и разностная ДН АФАР – СДН, РДН.

На рис. 3 приведены результаты расчётов СДН, РДН, дискриминационной характеристики для раннее не исследуемых амплитудных распределе-

ний на элементах ЛФАР при реализации моноимпульсного метода пеленгования.

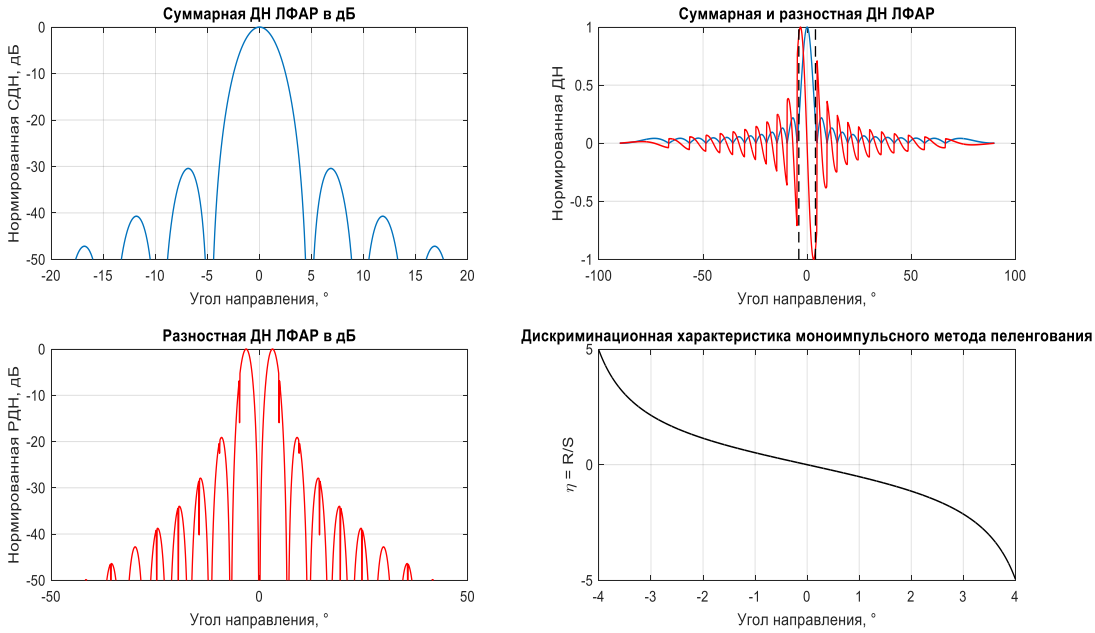


Рис. 3. Результаты расчётов СДН, РДН, дискриминационной характеристики при реализации моноимпульсного метода пеленгования

Необходимо формировать узкий луч АФАР в направлении информационного источника и формировать глубокие «нули» ДН или низкий уровень боковых лепестков ДН АФАР в направлении постановщиков помех и естественных помех. Естественные помехи – сигналы от соседних устройств и отражения от конструкций помещения, постановщики помех – умышленно формируемые помеховые сигналы для подавления и нарушения работы канала передачи данных. Суммарная и разностная ДН АФАР формируется для автоматического определения (пеленгования) направления на полезный источник сигнала для формирования узкого луча ДН АФАР в направлении полезного источника сигнала. Глубокие нули ДН АФАР формируются в направлении постановщиков помех, низкий уровень боковых лепестков ДН АФАР обеспечивает существенное снижение помеховых воздействий с любых направлений прихода естественных помеховых сигналов.

В табл. 2 приведены результаты расчетов весовых окон и их влияния на параметры СДН, РДН ЛФАР по критериям:

1) наличие нуля и максимума в ДН АФАР.

2) минимизация максимального уровня боковых лепестков суммарной и разностной ДН (СДН, РДН) при минимальном угловом расширении главного лепестка суммарной ДН ЛФАР.

3) максимальная крутизна дискриминационной характеристики моноимпульсного метода пеленгования.

Таблица 2

Сравнение весовых амплитудных распределений

Весовое окно	Ширина главного лепестка, град.	Уровень боковых лепестков СДН, дБ	Уровень боковых лепестков РДН, дБ	Крутизна РДН/СДН
Оптимальное с общей шириной лепестков	6,6	–34,70	–35,53	0,551
Оптимальное с минимальной боковой лепестковой энергией	4,7	–30,77	–21,68	0,93
Оптимальное с минимальной суммой квадратов пьедесталов	6,3	–31,38	–34,88	0,49
Оптимальное с минимальной площадью под кривой диаграммы направленности	4,5	–31,517	–21,78	0,58
Оптимальное с минимальным количеством несовпадений фаз	4,4	–30,68	–19,28	0,82
Оптимальное с наименьшей мощностью на пьедестале	6	–35,35	–35,41	0,58

Заключение. В исследовании исследуется потенциал технологии АФАР в обеспечении безопасности передачи данных по открытым каналам связи. С помощью моделирования и анализа продемонстрирована эффективность метода адаптивного формирования луча с целью уменьшения помех и повышения защиты данных. В ходе проведённого исследования по результатам анализа данных, приведённых в табл. 2 и по результатам анализа графиков (рис. 2, 3) можно сделать вывод, что оптимальное весовое окно с минимальным уровнем боковых лепестков реализует критерии качественной адаптации АФАР в борьбе с помехами и защиты информации при передаче по открытым каналам связи.

Библиографический список

1. Швырев Б.А. Технический канал утечки информации за счет имитации легального канала стандарта IEEE 802.11 // Швырев Б.А., Цимбал В.Н., Антонов А.С. // Вестник УрФО. Безопасность в информационной сфере. № 3(49) / 2023. С. 81–89.

2. Баимов Р.И. Выбор весового окна амплитудного распределения на элементах линейной фазированной антенной решетки по критерию ширина луча - уровень боковых лепестков диаграммы направленности / Р.И. Баимов, А.Н. Рагозин //

Инфокоммуникационные технологии: актуальные вопросы цифровой экономики : Сборник научных трудов III Международной научно-практической конференции, Екатеринбург, 25–26 января 2023 года / Под редакцией В.П. Шувалова, сост. М.П. Карачарова. – Екатеринбург: Уральский государственный университет путей сообщения, 2023. – С. 72–77. – EDN HMYRDD.

3. Баимов Р.И. Повышение точности пеленгования источника радиоизлучения за счёт выбора весового амплитудного распределения на элементах линейной фазированной решетки радиопеленгатора в составе радиоугломерной системы посадки/ Р.И. Баимов, А.Н. Рагозин // Материалы 10-й научной выставки-конференции научно-технических и творческих работ студентов. Министерство науки и высшего образования Российской Федерации Южно-Уральский государственный университет. 2023. С. 90–93.

4. Григорьев В.А., Щесняк С.С., Гулюшин В.Л., Распаев Ю.А., Лагутенко О.И., Щесняк А.С. //Адаптивные антенные решетки: учебное пособие в 2-ух частях. Часть 1. Санкт-Петербург: СПб: Университет ИТМО, 2016. С. 45–46.

5. Рагозин, А.Н. Определение угловых координат источника радиоизлучения в системах радионавигации / А.Н. Рагозин // Наука ЮУрГУ. Секции технических наук: материалы 74-й научной конференции, Челябинск, 19 апреля 2022 года/ Министерство науки и высшего образования Российской Федерации Южно-Уральский государственный университет. – Челябинск: Издательский центр ЮУрГУ, 2022. С. 343–349.

УДК 004.051 + 007.52 + 004.056.53 + 004.716 + 004.732 + 004.772

ФОРМИРОВАНИЕ ПЕРЕЧНЯ КРИТЕРИЕВ ДЛЯ ВЫБОРА ОПТИМАЛЬНОГО КАНАЛА ПЕРЕДАЧИ ИНФОРМАЦИИ УРОВНЯ ДАТЧИКИ – ЦЕНТР В КОНЦЕПЦИИ SMART CITY

А.А. Абдулов
Научный руководитель: канд. юр. наук, доц. В.М. Жернова
Южно-Уральский государственный университет,
г. Челябинск

Приведена модель связи датчиков и центра принятия решений. Приведены каналы передачи информации. Исследованы характеристики, примерная стоимость и защищенность каналов связи для используемой модели. Приведено сравнение каналов связи. Приведен вывод по выбору каналов передачи информации.

Ключевые слова: безопасность, каналы передачи данных, передача данных, стоимость, smart city.

Умный город – концепция устройства социального и экономического городского пространства. Эта технология внедряется для улучшения условий жизни городских жителей, решения экологических и демографических

проблем. В реализации этих программ помогают различные новаторские решения, обычно связанные с IT-сферой [1].

По данным Международного института управленческого развития, расположенного в Швейцарии, на 2021 год Москва заняла 54 место по рейтингу умных городов [2].

Внутри умного города для обеспечения его потребностей и увеличения комфорта, повышения качества жизни, уменьшения загрязнения окружающей среды используются несколько десятков подсистем.

В России движение в сторону «умных» городов началось в 2018 году. Министерство строительства и жилищно-коммунального хозяйства Российской Федерации приняло 4 марта 2019 г. документ «Базовые и дополнительные требования к умным городам (стандарт «Умный город»)», в котором представило свое видение концепции «smart city». Для реализации стандарта Министерство строительства и жилищно-коммунального хозяйства Российской Федерации заключило договоры с 19 городами-пилотами из 11 регионов России [3].

Для взаимодействия подсистемы с внешним миром, подсистеме нужны датчики, которые будут предоставлять информацию о внешнем мире. Эти датчики должны взаимодействовать между собой и центром принятия решений подсистемы через каналы передачи информации. В данной работе будут представлено сравнение нескольких видов каналов передачи данных.

Для проведения сравнения, пусть участок, обслуживаемый одним центром управления, состоит из двух километровых пересекающихся в центре участков дороги, и примем, что на одном участке дороги по 29 умных фонарей с каждой стороны, и расстояние между ними – 35 м. Тогда один центр управления должен контролировать 116 умных фонарей.

Различают две основные группы каналов передачи информации: проводные и беспроводные. Проводные каналы разделяются по виду кабеля: воздушные, витая пара, коаксиальный тип и оптоволокно.

Воздушные каналы использовать неуместно, так как у них отсутствует помехозащищенность, и пропускная способность у них минимальна.

Витая пара 7 категории используется для сетей с пропускной способностью до 10 Гбит/с при длине магистрали до 100 метров. Свыше 100 метров качество передачи информации сильно проседает.

Коаксиальные кабеля разделяются на тонкие и толстые. Толстый кабель категории RG-11 имеет максимальную длину сегмента до 500 м и скорость передачи до 10 Мбит/с [4]. Так как пропускная способность кабеля относительно невысокая, предполагаем, что для обеспечения принятого участка нужно провести свой кабель каждому умному фонарю, что в сумме составит примерно 29400 м. При стоимости в 8 тыс. рублей за 305 м, стоимость только кабеля составит около 800 тыс. рублей.

Для оптоволоконных кабелей будем рассматривать категорию OM2. Пропускная способность 1,25 Гбит/с [5]. Так как пропускная способность высокая, можем предположить, что для одной стороны дороги достаточно 2 линий кабеля, тогда всего нужно будет 8000 м кабеля, и при стоимости 340 рублей за метр, стоимость только кабеля составит примерно 2,72 млн рублей.

Все кабельные каналы связи нужно будет зарывать в землю для обеспечения их безопасности, что несет с собой дополнительные расходы на их установку.

В беспроводной связи будем рассматривать следующие технологии: WiMAX, LTE, Wi-Fi. Технологии 5G в данной статье рассматриваться не будут, так как они уже хоть и существуют, но не применяются повсеместно.

Также не будет рассматриваться и технология Li-Fi. Li-Fi – это система связи видимого спектра света, использующая его для отправки данных. Хоть Li-Fi может достигать скоростей передачи, которые в 100 раз превышают современный традиционный Wi-Fi, но она не может быть развернута на улице при солнечном свете или в любых других нестабильных условиях [6].

WiMAX – это технологический стандарт беспроводных сетей дальней связи для мобильных и фиксированных соединений, основанный на стандарте IEEE 802.16. Пропускная способность до 140 Мбит/с, а радиус действия 2-5 км [7].

Базовая станция (БС) MAXBridge BS 50 Pico может обслуживать одновременно свыше 100 абонентских станций (АС). Максимальная пропускная способность БС в канале шириной 10 МГц - 32 Мбит/с. Данная скорость обеспечивается на дальности максимально до 15-20 км в условиях прямой видимости, потребляемая мощность 20 Вт, стоимость в районе 90000 р. Приемник – например, Gembird PCMCIA-SATA2 стоимостью 250 рублей, и стоимость 116 этих приемников составит 29 тыс. рублей. Потребляемая мощность – 4,5 Вт, на 116–522 Вт. На принятый участок общая стоимость сети составит 119 тыс. рублей, потребляемая мощность – 542 Вт.

Одним из недостатков стандарта IEEE 802.16d является отсутствие аутентификации между БС и АС. В IEEE 802.16e постарались решить эту проблему с использованием сервера аутентификации RADIUS. При обмене сервера и АС используется пакет, содержащий поле, для вычисления которого инициатор сообщения должен иметь секретный код сервера. Только в этом поле используется секретный код сервера. Таким образом, злоумышленник, перехвативший пакет с этим полем, может вычислить значение этого секретного кода. Нужно заметить, что секретный код может не изменяться в течение месяца, причем для многих АС [8].

На сегодняшний день крупные компании уже свернули разработку WiMAX и перешли на LTE, а в России в 2022 г. не стали продлевать разрешение на использование частот, используемых операторами стандарта WiMAX [9].

LTE – это основное направление эволюции сетей сотовой связи третьего поколения (3G). Пропускная способность может составлять до 3 Гбит/с. Радиус покрытия может составлять от 3,2 км [10].

Стоимость базовой станции R45F, вместе с программным обеспечением для запуска и настройки сети составляет примерно 4,5 млн рублей. Потребляемая мощность 1 станции 400 Вт. Стоимость промышленного приемника Termit CR41P примерно 13 тыс. рублей, и стоимость 116 таких приемников составит примерно 1,5 млн рублей. Потребляемая мощность около 7 Вт, и общая потребляемая мощность составит 812 Вт. И таким образом, общая стоимость комплектующих сети составит 6 млн рублей, а потребляемая мощность – 1212 Вт.

В современных сетях LTE обнаружено множество угроз информации. Например, кража идентификатора с применением технических средств, следствием чего может являться перехват информации от датчика, отключение его от сети 4G. Также злоумышленник может совершить DoS-атаку на базовую станцию, парализовав работу всего участка [11].

Wi-Fi – технология беспроводной локальной сети на основе стандартов IEEE 802.11. Дальность связи не превышает 300 м, а скорость передачи данных одного из новейших стандартов 802.11ax составляет до 9,6 Гбит/с [12]. При использовании мощных узконаправленных антенн можно добиться дистанции связи в 1 км со скоростью 54 Мбит/с [13].

Стоимость точки доступа Edimax OAP1750 составляет 96 тыс. рублей, потребляемая мощность – 22 Вт. Стоимость одной узконаправленной антенны Ubiquiti AirMax Sector 5G-17-90 – 13 тыс. рублей, а четырех, для каждого направления, составит 52 тыс. рублей. Стоимость приемника Archer TX55E – 3,5 тыс. рублей, потребляемая мощность – 9,5 Вт, а 116 таких приемников будут стоить 406 тыс. рублей и потреблять 1102 Вт. Общая стоимость составит 554 тыс. рублей, а потребление энергии – 1124 Вт.

Наиболее распространенные угрозы, которые могут нести опасность в принятой организации сети – это подбор пароля, перехват и возможное изменение информации, атаки на протоколы защиты WPA и WPS, причем WPS наименее защищен, а также DoS-атаки для перегрузки сети [14].

Для улучшения безопасности системы с беспроводными каналами передачи информации можно ввести внешние вспомогательные узлы. Вспомогательный узел может быть глушителем для ухудшения качества канала перехватчика или ретранслятором для улучшения канала назначения за счет обеспечения разнесения связи. Но их введение несёт дополнительные расходы [15].

При малой защищенности каналов передачи информации злоумышленники могут перехватывать данные с этих каналов и использовать их в своих целях, могут блокировать передачу данных и этим скрывать преступления, могут передавать ложные данные, что может привести к неблагоприятным последствиям.

Заключение. Система «умный город» – будущее современных городов. И хотя для её реализации нужны более совершенные технологии и время, это будущее не так далеко. Поэтому нужно задумываться над этим будущим и его проблемами уже сейчас. И выбор каналов передачи информации, по которым подсистемы города будут транслировать данные одна из важнейших проблем.

Варианты с использованием витой пары и коаксиального кабеля имеют значительные недостатки, которые не перекрываются их положительными сторонами. Вариант с использованием оптоволоконных кабелей для постройки каналов передачи информации надежнее, имеет значительно более высокие дальность и пропускную способность, к кабелям практически невозможно незаметно подключиться. Однако он имеет и более высокую стоимость создания таких сетей и требует место для прокладки кабелей.

Использование беспроводных сетей LTE и Wi-Fi сопряжено с большими рисками, меньшими дальностью связи и пропускной способностью, но также и с меньшей стоимостью и большим удобством установки сетей. Использование WiMAX нецелесообразно из-за прекращения его поддержки.

Таким образом, предлагается компромиссное использование разных видов каналов связи. Использование оптоволоконных каналов на больших по площади, и соответственно, более загруженных, участках, использование Wi-Fi на меньших по величине участках, на которых сложно проложить оптоволокно, и использование LTE на средних по размеру участках, где также сложно проложить оптоволокно, а характеристик Wi-Fi не хватает.

Библиографический список

1. Умный город: концепция, технологии, примеры. – Текст: электронный // ТРАССКОМ. – 2022. – URL: https://trasscom.ru/blog/umnyj-gorod?ysclid= lfle63r0ed439945805 (дата обращения: 19.10.2023).

2. Рейтинг самых умных городов мира. – Текст: электронный // NoNews. – 2022. – URL: https://nonews.co/directory/lists/cities/smart-city-index (дата обращения: 19.10.2023).

3. Фантастические «умные» города сегодня. – Текст: электронный // Хабр. – 2021. – URL: https://habr.com/ru/company/sezinnopolis/blog/598147/ (дата обращения: 19.10.2023).

4. Оборудование сетей. Стандарты кабелей. Коаксиальные кабели. - Текст: электронный // StudFiles. – 2016. – URL: https://studfile.net/preview/5847825/ (дата обращения: 19.10.2023).

5. Виды оптических волокон. – Текст: электронный // Modultech. – 2019. – URL: https://modultech.ru/opticheskoe-volokno-printsip-raboty-vidy/?ysclid= lnyinvhxid499905592/ (дата обращения: 19.10.2023).

6. Сушко Р.Е. Технология LI-FI и её сферы применения // Форум молодых ученых. 2021. // URL: https://cyberleninka.ru/article/n/ tehnologiya-li-fi-i-eyo-sfery-primeneniya?ysclid=loi66m2w1t463151217 (дата обращения: 19.10.2023).

7. Вишневский В.М., Портной С.Л., Шахнович И.В. Энциклопедия WiMAX. Путь к 4G: монография. М., 2009. 472 с.

8. Безопасное применение технологии «мобильного Wimax». – Текст: электронный // Российская академия естествознания. – 2010. – URL: https://natural-sciences.ru/ru/article/view?id=7616&ysclid= lo1sjxhibt628338856/ (дата обращения: 20.10.2023).

9. Телеком-операторы в России больше не будут использовать частоты 3,4–3,8 ГГц. – Текст: электронный // Интерфакс. – 2022. – URL: https://www.interfax.ru/digital/850413/ (дата обращения: 20.10.2023).

10. Технические науки в России и за рубежом: материалы III Междунар. науч. конф. (г. Москва, июль 2014 г.). – М.: Буки-Веди, 2014. 40 с.

11. Обзор угроз безопасности сетей стандарта LTE. – Текст: электронный // Синергия Наук. – 2021. – URL: http://synergy-journal.ru/archive/article4664?ysclid= lo1to2clw8962773934/ (дата обращения: 21.10.2023).

12. Тюхтяев Д.А., Морозов Д.А. Аналитический обзор нововведений стандарта IEEE 802.11ax // NBI-technologies. 2022. № 2. Т. 16. С. 21–26. URL: https://cyberleninka.ru/article/n/analiticheskiy-obzor-novovvedeniy-standarta-ieee-802-11ah/viewer (дата обращения: 21.10.2023).

13. Промышленные беспроводные сети: какую выбрать? – Текст: электронный // Habr. – 2019. – URL: https://habr.com/ru/companies/phoenix_contact/articles/441146/ (дата обращения: 22.10.2023).

14. Уязвимости WiFi сетей и методы защиты - подробный разбор со скриншотами и комментариями. – Текст: электронный // Overclockers. – 2023. – URL: https://overclockers.ru/blog/melok/show/96949/uyazvimosti-wifi-setej-i-metody-zaschity-podrobnyj-razbor-so-skrinshotami-i- kommentariyami#b_ugr/ (дата обращения: 22.10.2023).

15. Danda B. Rawat and Kayhan Zrar Ghafoor (Eds.), "Smart Cities Cybersecurity and Privacy", Elsevier Press, ISBN: 9780128150320, November 2018.

УДК 004.357 + 004.056 + 537.874.2 + 537.874.7 + 621.391.8

ЭКРАНИРОВАНИЕ КОЛОНКИ ДЛЯ ПРОВЕДЕНИЯ СПЕЦИАЛЬНЫХ МЕРОПРИЯТИЙ

К.Б. Храмцов
Научный руководитель: ст. преп. кафедры «Защита информации»
И.С. Антясов
Южно-Уральский государственный университет,
г. Челябинск

Исследованы методы экранирования акустических излучателей для проведения специальных мероприятий, измерены характеристики специализированной колонки и обычной на наличие побочных электромагнитных излучений. Установлено, что обычная колонка не может использоваться для проведения исследований, предложен вариант ее модернизации.

Ключевые слова: экранирование, АЭП, ПЭМИН, экранированная колонка, специальные мероприятия, колонка.

Введение

Утечка защищаемой информации представляет реальную угрозу для государственных учреждений, крупных организаций, информационных систем различного характера и государства в целом. Одной из необходимых мер для исключения утечки защищаемой информации является проведение специальных мероприятий, направленных на поиск электронных устройств негласного получения информации и потенциальных технических каналов утечки информации. Некоторые из них необходимо проводить с использованием тестового акустического воздействия, для создания которого используется акустическая колонка.

1. Эффект акустоэлектрических преобразований

Под акустоэлектрическим преобразованием (АЭП) понимают преобразование механической энергии акустического сигнала отдельными устройствами в электрический сигнал (напряжение, ток, заряд), модулированный по закону изменения акустического сигнала. В свою очередь электрические сигналы создают электрическое и магнитное поля, которые также могут образовать канал утечки информации. [1]

Акустоэлектрическим эффектом обладают многие элементы электронных технических средств обработки информации и вспомогательных технических средств.

Для прямого акустоэлектрического преобразования измерение величины сигналов речевого диапазона частот исследуемого технического средства (ТС) рекомендуется типовая схема (рис. 1). [2]

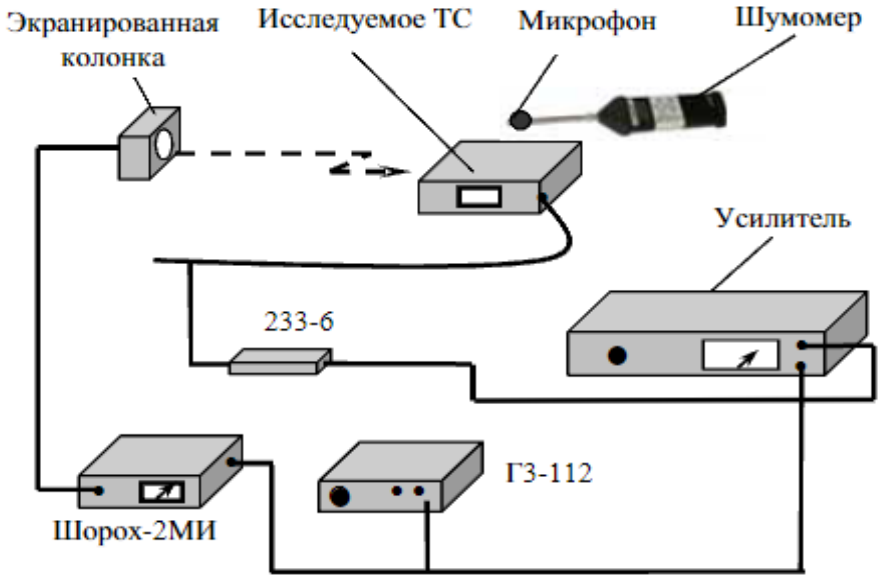


Рис. 1. Типовая схема измерения прямого акустоэлектрического преобразования

При проведении исследований АЭП необходимо воздействовать на техническое средство чистой акустической волной, соблюдая требуемый уровень звукового давления и допустимый уровень электромагнитных наводок. Для соблюдения требований используются специальные экранированные колонки.

2. Принципиальная схема колонки

Рассмотрим принципиальную схему колонки (рис. 2). Электромагнитную волну излучают динамики, основным элементом которых является магнит, раскачивающий мембрану акустической колонки.

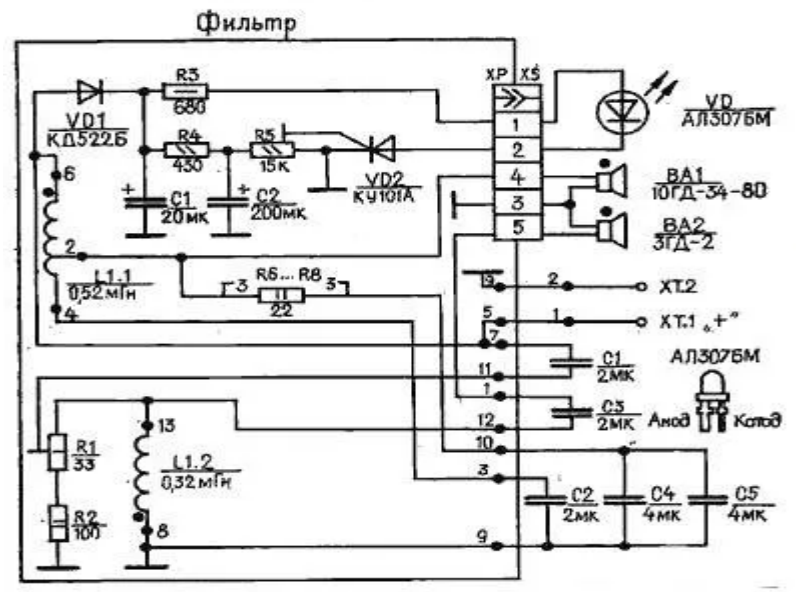


Рис. 2. Принципиальная схема акустической колонки

Зачастую конденсаторы исследуемых технических средств являются наиболее эффективными акустоэлектрическими преобразователями, так как обкладки под действием звука могут перемещаться относительно друг друга в поперечном направлении, изменяя емкость конденсатора. Также достаточно часто причиной акустоэлектрических преобразований являются керамические конденсаторы, содержащие материалы с пьезострикционным эффектом и являющиеся подобием пьезоэлектрического микрофона [1].

Однако данный эффект может быть слабо выражен на фоне сильного воздействия побочных электромагнитных излучений (ПЭМИ) от динамиков акустической колонки, содержащих в себе магнит. В связи с чем в ходе исследований технических средств существует риск ошибочного принятия эффекта ПЭМИ вместо акустоэлектрических преобразований. В соответствии с законом электромагнитной индукции, открытым Фарадеем, электродвижущая сила (ЭДС) наводится при движении проводников в магнитном поле под действием энергии звуковой волны. Магнитное поле всегда присутствует в ферромагнитных сердечниках за счет остаточной индукции.

3. Требования, предъявляемые к колонкам

Поскольку измерения проводятся на шумовом тест-сигнале (что не исключает и других сигналов), то источник шума, в общем случае колонка, должен быть мощным, но помехоустойчивым. Звуковое давление, развиваемое на расстоянии 1 м источником желательно иметь не менее 100 дБ в каждой октавной полосе [3]. При меньших величинах акустического давления выделение опасных сигналов на фоне помех канала утечки достаточно сложно или вообще невозможно. Также должна иметься возможность гибкого регулирования амплитудно-частотной характеристики источника и возможность увеличения уровня сигнала в заданной полосе частот, когда это необходимо.

4. Экранирование

Экранированием называется локализация электромагнитной энергии в определенном пространстве за счет ограничения ее распространения всеми возможными способами [4]. Экранирование колонки играет важную роль в получении достоверных результатов проведенных работ. В случае акустического излучателя прежде всего необходимо экранировать динамики.

Экранирование динамиков от возникновения побочных электромагнитных излучений и наводок (ПЭМИН) можно добиться следующим путем [5]:

1. Экраны, сделанные из магнитопроводящих материалов, могут быть размещены вокруг магнита динамика, чтобы уменьшить распространение магнитных полей. Это помогает предотвратить электромагнитные помехи, которые могли бы повлиять на близлежащие устройства и измерения в целом;

2. Кожухи из металлических материалов, могут быть установлены вокруг динамиков, чтобы уменьшить излучение магнитных полей и предотвратить внешние помехи;

3. Для динамиков, генерирующих сильные электрические поля, могут использоваться дополнительные экранированные оболочки или заземление элементов;

4. Для компенсации магнитного поля можно использовать магнит, расположенный обратной полярностью по отношению к имеющемуся в динамике.

Самым эффективным методом экранирования является комбинирование компенсации магнитного поля и магнитного кожуха. Необходимо взять магнит от такого же динамика, установленного в колонке и расположить его обратной стороной, после чего закрепить и экранировать с помощью магнитопроводящего материала (рис. 3). Такой подход реализован в акустической системе Miller & Kreisel® S150T Tripole [6].

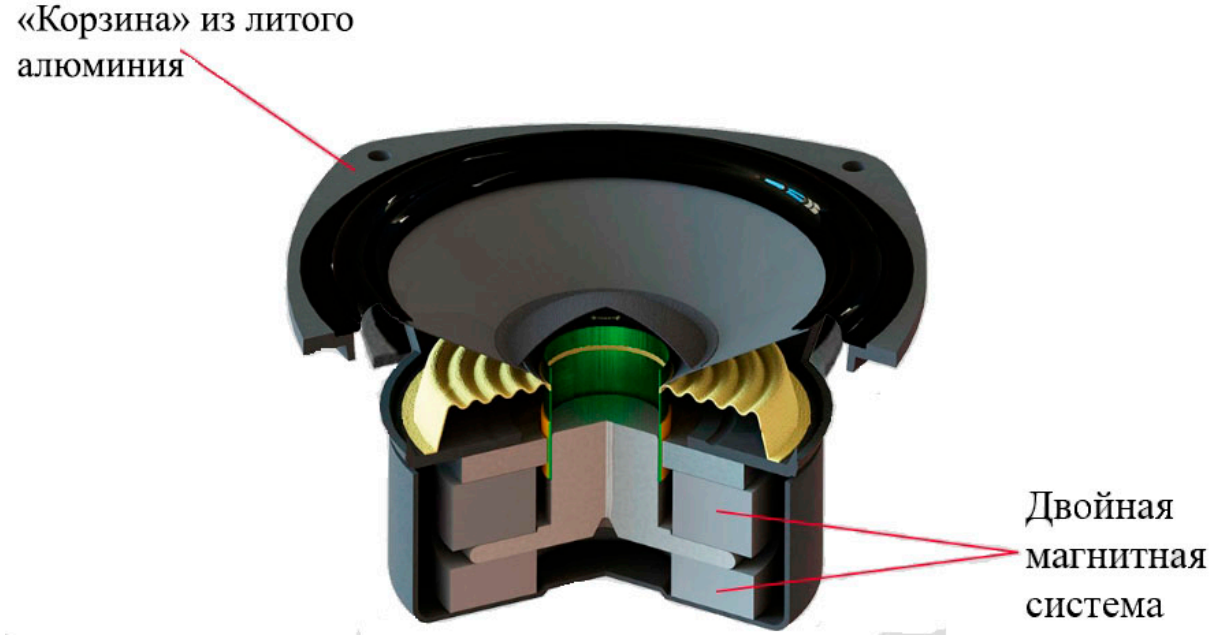


Рис. 3. Пример экранирования динамика

5. Обзор существующих на рынке решений

На сегодняшний день на рынке защиты информации существует несколько вариантов экранированных колонок (табл. 1), рассмотрим их.

Таблица 1

Экранированные акустические системы

	АС-1	**АС-10**	**АТ-2**	**УЭК**
Производитель	ЗАО НПЦ «НЕЛК»	ЗАО НПЦ «НЕЛК»	"Дип Электроникс"	«МАСКОМ»
Диапазон рабочих частот генератора	10 Гц – 20 кГц	–	–	–

Окончание табл. 1

	АС-1	**АС-10**	**АТ-2**	**УЭК**
Уровень звукового давления (в диапазоне 70 Гц-6кГц)	100 дБ	90 дБ – 95 дБ	116 дБ	–
Уровень звукового давления (в диапазоне 6 кГц – 19кГц)	90 дБ	100дБ – 105 дБ	96 дБ	–
Частотный диапазон	70 Гц – 19 кГц	100 Гц – 11,2 кГц	70 Гц – 20 кГц	200 Гц – 8 кГц
Стоимость, тыс. р.	от 180	от 320	от 150	от 90
Экранирование	Экранированнный короб	Экранированный короб	Экранированный короб в комплекте	Экранированный короб

6. Измерение электромагнитной напряженности

Основной целью экранирования является уменьшение электромагнитной напряженности поля, создаваемого магнитом динамиков, измерим данную величину на примере двух колонок.

Для измерения были взяты: экранированная колонка АС-10 и обычная неэкранированная колонка (рис. 4). Измерения проводились на альтернативной измерительной площадке ООО ЧОО «Аргумент». Антенны для проведения измерений установили на расстоянии одного метра от исследуемых колонок (рис. 5).

Уровень громкости колонок при проведении измерений был взят приблизительно равным (рис. 6). Громкость измерялась с помощью шумомера «Экофизика-110А».

Рис. 4. Измеряемые колонки

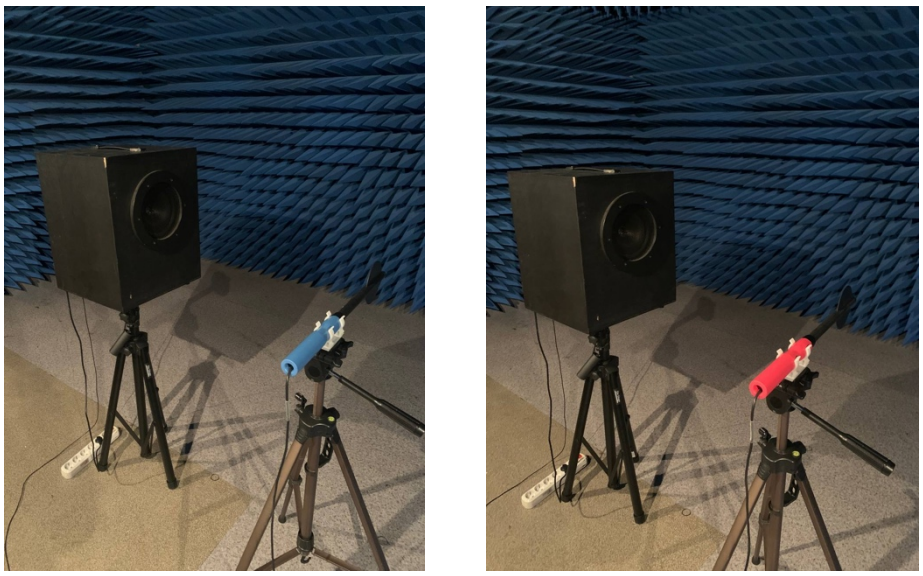


Рис. 5. Установка антенн

F, Гц	Громкость,Дб АС-10	Громкость,Дб колонка
100,000	98,280	96,771
200,000	109,162	106,941
300,000	105,805	104,742
400,000	110,503	109,915
500,000	111,425	109,266
600,000	107,875	104,422
800,000	112,367	112,412
1000,000	106,920	105,364
1200,000	101,746	100,712
1600,000	105,547	105,256
2000,000	106,631	104,921
2500,000	106,127	104,630
3000,000	105,463	102,118
4000,000	110,110	108,326
5000,000	106,711	105,787
6500,000	102,076	101,252
8000,000	99,174	98,024

Рис. 6. Громкость колонок

По результатам измерений были построены графики (рис. 7).

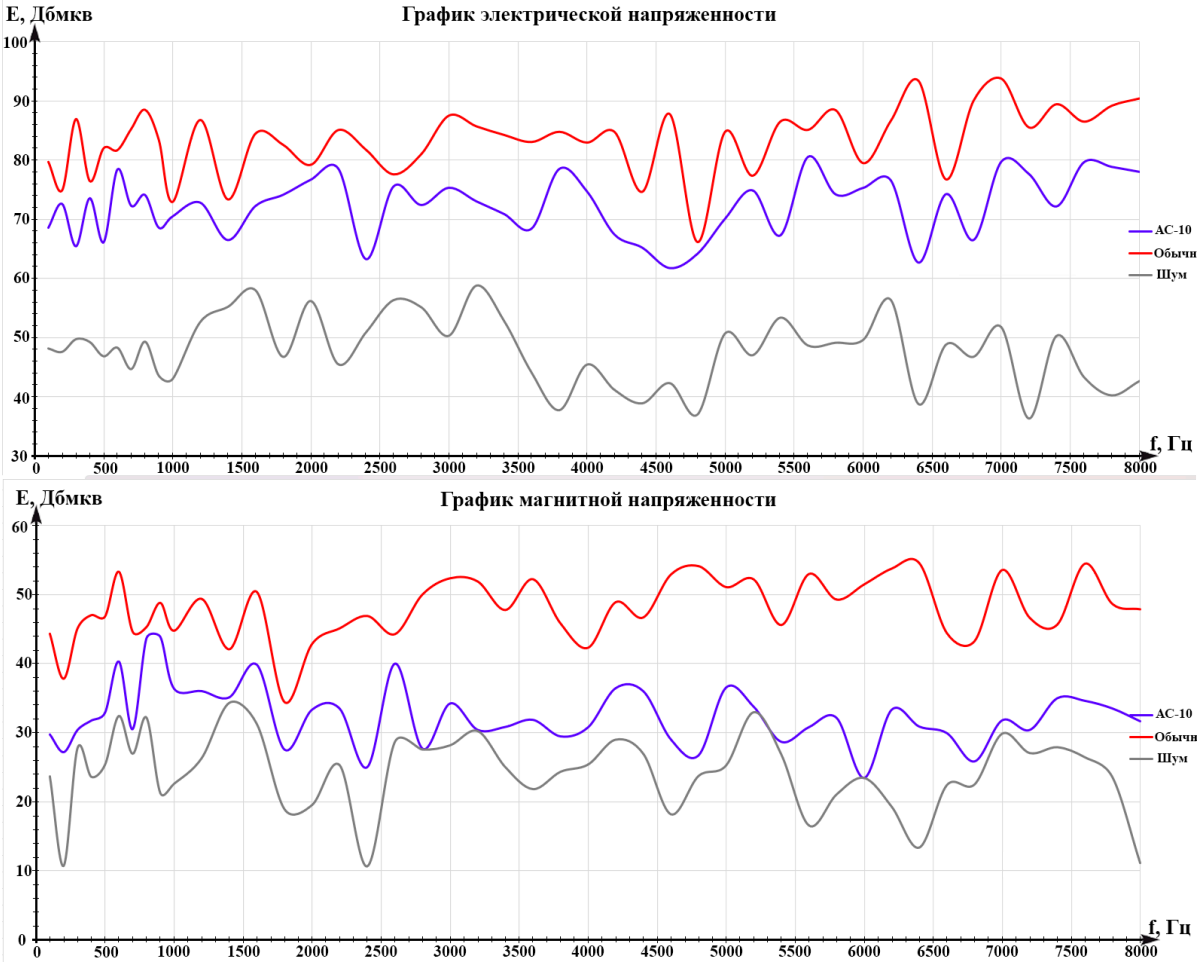


Рис. 7. Графики электромагнитной напряженности

Из графиков электромагнитной напряженности видно, что в среднем показатель напряженности экранированной колонки ниже неэкранированной на 11 дБ для электрической и на 15 дБ для магнитной. При этом электрическая напряженность АС-10 в среднем выше шума на 24 дБ, магнитная на 8 дБ. В нескольких точках график магнитной напряженности АС-10 приближается к графику шума, что говорит о наилучших показателях экранирования на данных частотах.

Неэкранированная колонка заметно хуже себя показывает на всем диапазоне частот. Это может привести к неверным результатам при использовании излучателя для проведения измерений.

Для улучшения ситуации обычную колонку следует экранировать, для этого необходимо ее разобрать, подобрать магнит с такими же свойствами как у магнита динамика, а также изготовить экран из магнитопроводящего материала. При проведении работ стоит учитывать возрастание массы акустического излучателя и уменьшение внутреннего объема колонки. В дальнейшем предполагается разработка собственного варианта колонки на основе комбинированного подхода к электромагнитному экранированию.

Библиографический список

1. Зайцев А.П. Технические средства и методы защиты информации: учебник / А.П. Зайцев, Р.В. Мещеряков, А.А. Шелупанов. – 7-е изд., испр. – Москва: Горячая линия-Телеком, 2018. – 359 с.

2. Бузов Г.А., Калинин С.В., Кондратьев А.В. Защита от утечки информации по техническим каналам: учеб. пособие. – М.: Горячая линия – Телеком, 2005. – 416 с.

3. ПРОТОКОЛ испытаний колонки экранированной активной акустической «Гамма КЭАА-20» БЮЛИ.465317.002 // URL: https://nppgamma.ru/upload/iblock/9ab/9ab5a848fcb53895687369393e527ba4.pdf?ysclid=lom1w6ob9b220283681 (дата обращения: 22.10.2023).

4. Сагдеев К.М. Физические основы защиты информации: учебное пособие / К.М. Сагдеев, В.И. Петренко, А.Ф. Чипига. – Ставрополь: СКФУ, 2015. – 394 с.

5. Петров И.С. Локализация и ослабление побочных электромагнитных излучений от средств вычислительной техники путем экранирования электромагнитных волн / И.С. Петров // Вестник Южно-Уральского государственного университета. Серия: Компьютерные технологии, управление, радиоэлектроника. – 2012. – № 23. – С. 189–191. – ISSN 1991-976X.

6. Официальный сайт производителя активной акустической системы «Miller & Kreisel® S150T Tripole» // URL: https://mksound.com/products/150-series/s150t/ (дата обращения: 24.10.2023).

СЕКЦИЯ «ПРОГРАММНО-АППАРАТНЫЕ СРЕДСТВА ЗАЩИТЫ ИНФОРМАЦИИ И КОМПЬЮТЕРНАЯ БЕЗОПАСНОСТЬ»

УДК 004.056

РАЗРАБОТКА ВИРТУАЛЬНЫХ МАШИН ДЛЯ ОБУЧЕНИЯ ТЕСТИРОВАНИЮ НА ПРОНИКНОВЕНИЕ

М.В. Афанасьева, У.В. Кузьмина, А.Р. Федорова, О.А. Казаков
Научный руководитель: ст. преп. каф. ИиИБ М.В. Афанасьева
к.т.н., доц. каф. ИиИБ У.В. Кузьмина
Магнитогорский государственный технический университет имени Г.И. Носова, г. Магнитогорск

Статья представляет один из подходов к обучению в области кибербезопасности, основанный на создании виртуальных машин с заранее спроектированными уязвимостями и многочисленными алгоритмами шифрования, воплощая в себе идею Capture The Flag. Виртуальные машины разработаны на разные уровни сложности, что позволяет адаптировать обучение к разнообразным уровням знаний и опыта студентов. Благодаря этому студенты получат возможность овладеть практическими навыками в области кибербезопасности и познакомиться с различными методами шифрования. Этот подход способствует индивидуальному росту студентов и подготовке нового поколения специалистов в сфере информационной безопасности.

Ключевые слова: анализ уязвимостей, виртуальные машины, обучение в области ИБ, тестирование на проникновение.

Современный мир информационных технологий требует постоянного развития и совершенствования навыков в области кибербезопасности и управления информацией для того, чтобы опередить действия злоумышленника. Важнейшим аспектом этого процесса является не только понимание теории, но и практические навыки, необходимые для обнаружения и устранения уязвимостей в компьютерных системах. В этой статье рассмотрен один из подходов к обучению, который мы реализовали для студентов и нового поколения специалистов в сфере информационной безопасности, помогая им повысить компетентность в области кибербезопасности и информационных технологий.

Данный подход основан на создании специальных виртуальных машин, которые были спроектированы с заранее созданными ошибками в конфигурации и уязвимостями. Эти виртуальные среды не только воспроизводят реальные ситуации, в которых специалисты по кибербезопасности могут столкнуться с вызовами, но и стимулируют студентов искать решения для их устранения.

Виртуальные машины предоставляют студентам уникальную возможность изучить различные аспекты кибербезопасности, начиная с анализа и определения уязвимостей, и заканчивая их устранением. Важно отметить, что каждая из уязвимостей и ошибок в наших виртуальных средах тщательно связана между собой, создавая сложные сценарии, которые требуют глубокого анализа и многогранных навыков для решения [1, 2].

Схема реализации виртуальной машины с использованием уязвимостей для нахождения флага показана на рис. 1.

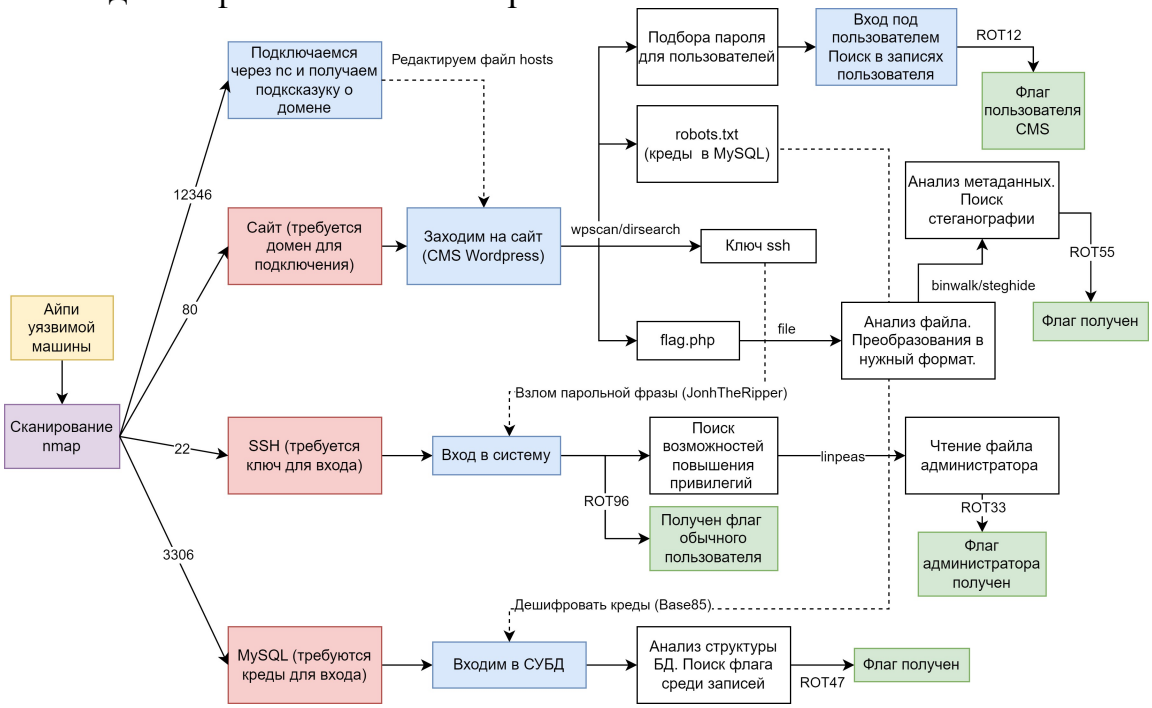


Рис. 1. Схема использования уязвимостей для нахождения флага

Рассмотрим вектор атаки на CMS Wordpress, когда был получен доступ к файлам в одной из директорий (рис. 2).

```
┌──(kali㉿kali)-[~/Desktop/Example]
└─$ wget http://192.168.0.74:8000/flag.php
--2023-10-22 20:17:09--  http://192.168.0.74:8000/flag.php
Connecting to 192.168.0.74:8000... connected.
HTTP request sent, awaiting response... 200 OK
Length: 115957 (113K) [application/octet-stream]
Saving to: 'flag.php'

flag.php                    100%[=========================>] 113.24K  --.-KB/s    in 0s

2023-10-22 20:17:09 (716 MB/s) - 'flag.php' saved [115957/115957]


┌──(kali㉿kali)-[~/Desktop/Example]
└─$ file flag.php
flag.php: JPEG image data, JFIF standard 1.01, resolution (DPI), density 300×300, segment length 16, baseline, prec
ision 8, 1170×715, components 3

┌──(kali㉿kali)-[~/Desktop/Example]
└─$ mv flag.php flag.jpg

┌──(kali㉿kali)-[~/Desktop/Example]
└─$ steghide extract -sf flag.jpg
Enter passphrase:
wrote extracted data to "secret.txt".

┌──(kali㉿kali)-[~/Desktop/Example]
└─$ head secret.txt
x"yT}%x~8"ðx~|,V
```

Рис. 2. Получение флага

В ходе выполнения действий по получению флага, представленных на рис. 2, студент получил файл flag.php. Далее с помощью инструмента file он выявил информацию, что это изображение. После этого с помощью утилиты stegsite студент произвел попытку обнаружить скрытую информацию внутри этого изображения. Он выяснил, что там спрятан секретный текстовый документ. В этом документе был найден ключ в зашифрованном виде, поэтому в дальнейшем была произведена дешифровка.

Одной из важных особенностей созданных нами разнообразных образовательных виртуальных машин является возможность выбора уровней сложности, так как у студентов разный уровень знаний и навыков, и их потребности в обучении также могут значительно отличаться. Поэтому наши виртуальные машины создаются с учетом нескольких уровней сложности:

– Начальный уровень: эти виртуальные машины предоставляют базовые сценарии и задачи, которые помогают студентам, только начинающим изучать кибербезопасность, понять основы безопасности и практические аспекты шифрования.

– Средний уровень: сценарии на среднем уровне сложности предлагают более продвинутые вызовы, которые требуют более глубокого понимания методов шифрования и уязвимостей. Эти варианты могут быть подходящими для студентов, имеющих определенный опыт.

– Продвинутый уровень: виртуальные машины на продвинутом уровне сложности разработаны для студентов, которые уже обладают высокими навыками в области кибербезопасности. Они предлагают сложные сценарии и задания, требующие глубокого анализа и креативных решений.

Этот индивидуальный подход позволяет студентам выбирать уровень сложности, который соответствует их текущим знаниям и уровню подготовки. Он также способствует их постепенному росту и развитию, поскольку они могут сначала освоить более простые сценарии, а затем переходить к более сложным по мере увеличения своих навыков.

Вдобавок для усиления сложности обучающей среды, мы интегрировали различные алгоритмы шифрования, такие как Base85, ROT47 и другими, с целью поднять уровень осведомленности о разнообразии криптографических методов и их значении. Это позволяет студентам оценить важность безопасности данных и разработать навыки в области шифрования и дешифрования, что является неотъемлемой частью современных методов обеспечения конфиденциальности информации [3, 4].

Одним из ключевых аспектов в обучении в области кибербезопасности является возможность для студентов изучать инструменты, которые могли бы использоваться злоумышленниками для атак. Это имеет неоспоримую важность по нескольким причинам:

1. Понимание того, как работают инструменты взлома, позволяет студентам разрабатывать более эффективные методы защиты. Они могут вы-

являть уязвимости, предотвращать атаки и разрабатывать стратегии обороны, применяя свои знания в реальных сценариях.

2. Этическое использование: изучение инструментов в контролируемой среде обучения способствует формированию этичного подхода к их использованию. Студенты учатся применять свои знания для благих целей, законных аудитов безопасности и защиты информации.

3. Реальный мир: современная кибербезопасность требует знания о том, какие инструменты и методы используются злоумышленниками. Это помогает студентам адаптироваться к современным вызовам и разрабатывать стратегии противодействия.

4. Обучение деталям: изучение механизмов работы инструментов взлома углубляет понимание студентов о принципах кибератак и механизмах уязвимостей. Это позволяет им быть более компетентными в анализе и реагировании на разнообразные угрозы [5].

Также важными аспектами этого проекта является возможность интегрировать виртуальные машины в виртуальную среду EVE-NG (Emulated Virtual Environment – Next Generation), которая открывает широкие возможности для создания и моделирования сложных сетевых топологий и сценариев для обучения и практики в области информационной безопасности.

В дальнейшем планируется развертывание нескольких взаимосвязанных виртуальных машин, которые будут работать в единой сетевой среде для обеспечения более реалистичных сценариев. Взаимосвязь между этими машинами будет способствовать практике обнаружения и реагирования на атаки в сетевой среде, изучению принципов работы средств мониторинга, а также улучшению навыков восстановления после инцидентов.

Дополнительно планируется создание платформы для сдачи полученных флагов, которая основана на уникальном идентификаторе для каждого студента. Платформа не исключает факта передачи хода выполнения работы, поэтому оптимальным вариантом является диалог с преподавателем, так как целью созданных виртуальных машин служит обучение работы с утилитами, принципами и логикой атак. Пока что только в разговоре преподаватель сможет оценить глубину знаний и выявить слабые места.

Кроме того, рассматривается возможность создания системы компоновщика, которая будет собирать виртуальную машину по заданным уязвимостям. На данный момент эта задача требует значительных усилий, так как не всегда вручную получается установить компоненты, которые будут взаимосвязаны между собой.

Создание образовательных виртуальных сред, представленных в данной статье, предоставляет площадку для студентов и нового поколения специалистов в сфере информационной безопасности, чтобы они могли развивать свои навыки в области кибербезопасности и информационных технологий. Этот подход к обучению помогает им приобрести практиче-

ские знания и навыки, необходимые для успешной карьеры в сфере информационной безопасности, и способствует повышению уровня информационной грамотности в целом. Наша работа доказывает, что обучение, комбинирующее теорию и практику, является ключевым фактором для подготовки будущих специалистов в области кибербезопасности.

Библиографический список

1. Баранкова И.И., Михайлова У.В., Быкова Т.В. Сложности, возникающие при проведении аудита информационной безопасности на предприятии // Вестник УрФО. Безопасность в информационной сфере. 2019. С. 53–56.

2. Кузьмина У.В., Абзалутдинов Д.Р., Бараков К.Я. Создание модуля киберполигона, имитирующего компьютерные атаки // Актуальные проблемы современной науки, техники и образования. 2023. С. 54–57.

3. Кузьмина У.В., Мирская С.Д., Корнещук Р.К. Подход red team как способ обеспечения безопасности информации// Научный аспект. 2023. С. 1638–1644.

4. Тестирование на проникновение [Электронный ресурс] // Режим доступа: https://www.ptsecurity.com/ru-ru/services/pentest (дата обращения: 19.10.23).

5. Афанасьева М.В., Абзалутдинов Д.Р., Бараков К.Я. Создание модуля киберполигона, полноценно имитирующего компьютерные атаки // Актуальные проблемы современной науки, техники и образования. Тезисы докладов 81-й международной научно-технической конференции. 2023. С. 406.

УДК 004.056.53

XSS УЯЗВИМОСТИ И МЕТОДЫ ИХ ПРЕДОТВРАЩЕНИЯ

К.И. Головин
Научный руководитель: ассистент кафедры ИиИБ Л.А. Григоренко
Магнитогорский государственный технический университет имени Г.И. Носова, г. Магнитогорск

На сегодняшний день, вследствие развития веб-технологий, безопасность в сети стала одной из наиболее важных проблем. Одной из самых распространенных уязвимостей, связанных с веб-системами, является XSS. В данной статье приведены практические примеры её эксплуатации, а также предложены рекомендации по защите веб-сайтов от межсайтового скриптинга.

Ключевые слова: XSS-уязвимость, безопасность, веб-приложение, межсайтовый скриптинг.

В настоящее время, с развитием информационных технологий и распространением веб-приложений, безопасность в сети стала одной из

наиболее актуальных проблем [1]. Согласно статистике компании «Positive Technologies», число успешных атак на веб-ресурсы стремительно растет с каждым годом (рис. 1).

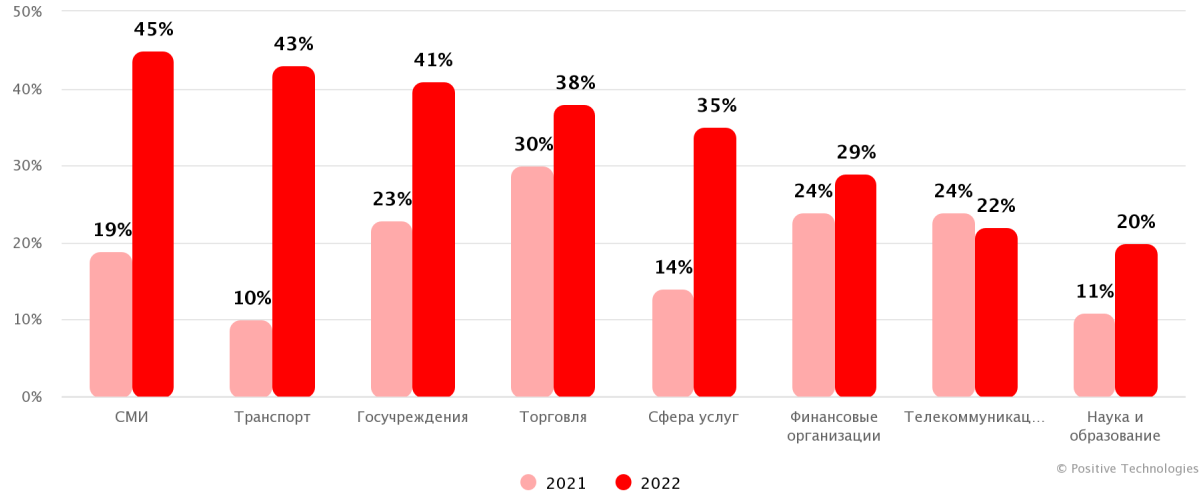


Рис. 1. Рост числа атак на веб-ресурсы

Одной из наиболее распространенных уязвимостей, связанных с веб-системами, является XSS. По оценкам компании «Positive Technologies», данная уязвимость возглавляет топ актуальных угроз, связанных с атаками на пользователей в веб-приложениях (рис. 2).

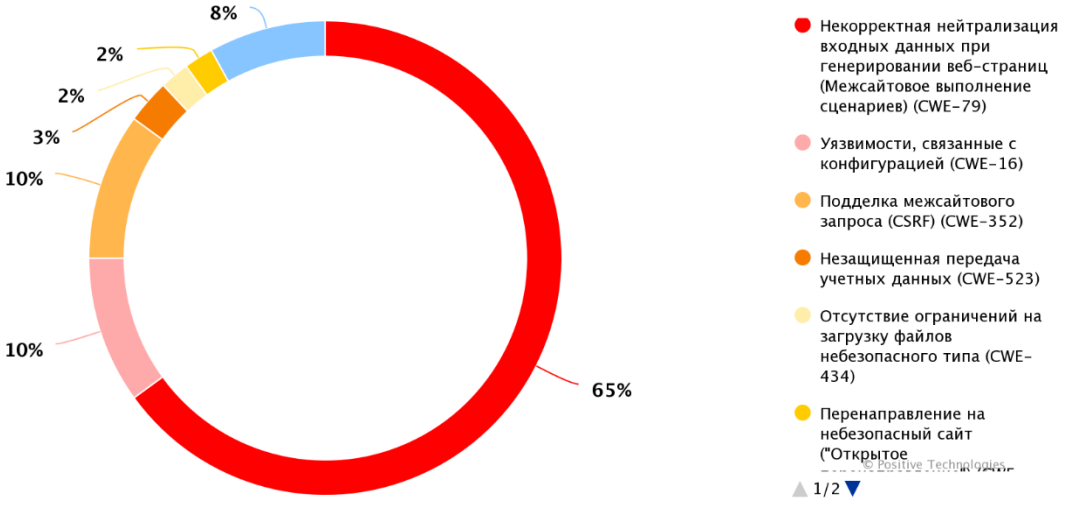


Рис. 2. Актуальные угрозы веб-приложений

XSS (Cross-Site Scripting) – уязвимость веб-приложения, являющаяся разновидностью CSRF (Cross-Site Request Forgery). Её суть заключается во внедрении в веб-страницу вредоносного JavaScript кода и его дальнейшем взаимодействии с сервером хакера [2]. Эксплуатация XSS-уязвимости позволяет злоумышленнику перенаправлять пользователей приложения на

фишинговый сайт, выдавать себя за других пользователей, устанавливать на устройства жертв вредоносное программное обеспечение.

Выделяют три основных вида XSS-уязвимостей [3]:

1. Хранимый XSS – особенность данного вида заключается во внедрении вредоносного кода на сервер. Сохраненный код обработается в браузере пользователя, когда тот откроет зараженную страницу.

2. Отраженный XSS – данная уязвимость зачастую эксплуатируется через URI. Вредоносный скрипт внедряется в параметры адреса. Сформированная ссылка доставляется жертве. Перейдя по ней, пользователь отправит запрос на сервер, который в ответ сформирует страницу с уже внедренным вредоносным кодом.

3. XSS на основе DOM (Document Object Model) – это тип уязвимости, связанный с веб-приложениями, при котором злоумышленник может внедрить вредоносный код на страницу для создания или изменения DOM-элементов.

Хранимый XSS считается наиболее опасным. Эксплуатация происходит путем выполнения вредоносного скрипта, загруженного злоумышленником на сервер веб-приложения.

В качестве примера можно привести ситуацию, связанную с кражей cookie у пользователя с целью получения доступа к ресурсу от его лица.

Имеется форум, на котором пользователи имеют возможность писать посты и оставлять к ним комментарии (рис. 3).

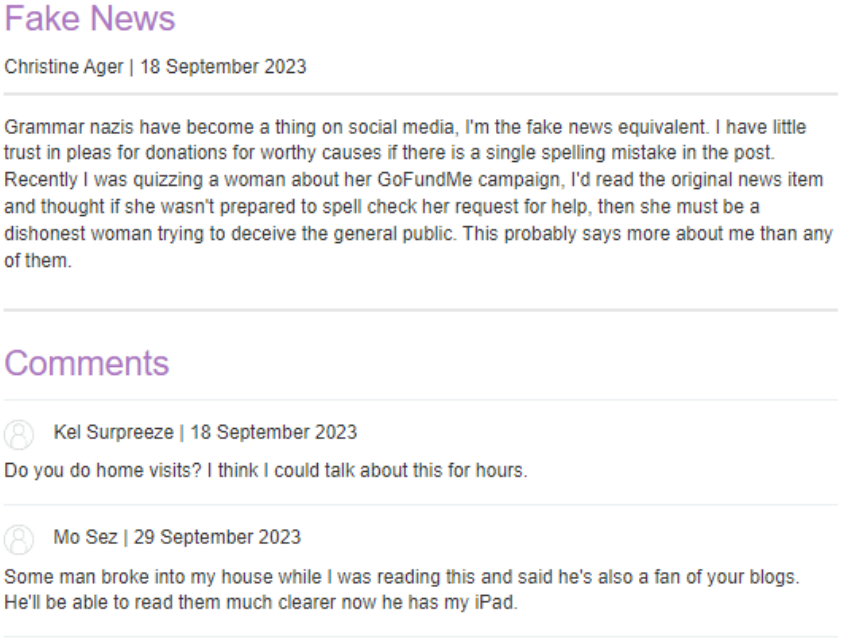


Рис. 3. Форум

Для того, чтобы оставить комментарий, используется форма ввода, в которую можно внедрить вредоносный код (рис. 4).

Рис. 4. Форма ввода

Данный скрипт работает так, что, когда пользователь заходит прочитать комментарии на зараженной странице, его браузер отправляет HTTP-запрос на сервер злоумышленника, содержащий в себе cookie жертвы. Теперь злоумышленник может авторизоваться в системе под видом данного пользователя.

Отражённый XSS отличается тем, что вредоносный код вписывается прямо в URI-адрес страницы, перейдя по которому браузер жертвы выполнит скрипт злоумышленника.

К примеру, имеется сайт с поисковой строкой, содержимое которой сохраняется в URI (рис. 5).

Рис. 5. Сайт с поисковой строкой

В данном случае получен URI «https://example.com /?search=%3Cscript%3Ealert%28%22123%22%29%3C%2Fscript%3E», в котором содержится вредоносный скрипт. Теперь, когда жертва перейдет по данной ссылке, на её странице отобразится уведомление (рис. 6).

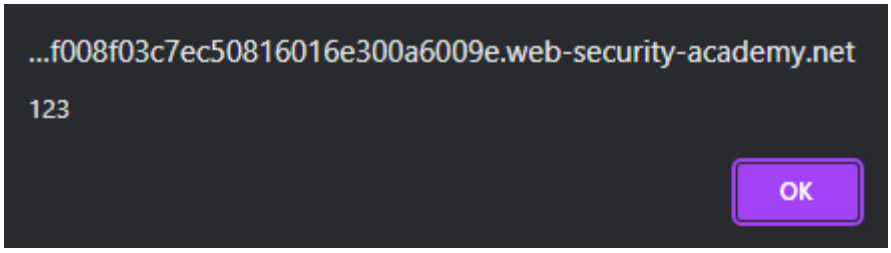


Рис. 6. Уведомление

Данный пример лишь отражает проверку концепции, и не несет в себе весомого ущерба для системы, однако он дает понять, что злоумышленник способен реализовать наихудший сценарий для веб-сайта.

XSS на основе DOM похож на отраженный, он также зачастую реализуется через переход жертвы по вредоносной ссылке. Отличается же он тем, что злоумышленник изменяет структуру DOM дерева страницы, основываясь на введенных пользовательских данных.

Примером может послужить уже продемонстрированный веб-сайт с поисковой строкой, но на этот раз XSS будет реализован через создание нового элемента на странице.

Изучив код страницы, можно заметить, что пользовательский ввод встраивается в атрибут тега «img» (рис. 7).

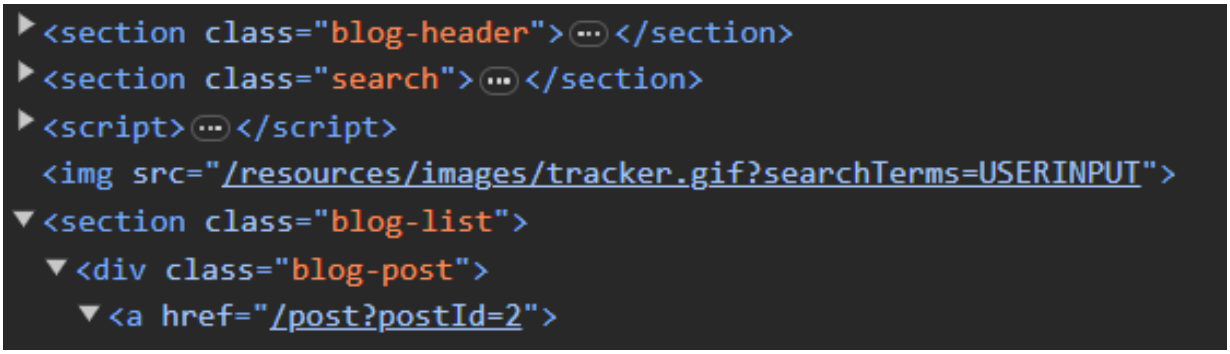


Рис. 7. Код страницы

Таким образом, есть возможность закрыть данный тег введенным кодом, а далее добавить свой элемент, который будет реализовывать необходимый функционал (рис. 8).

Рис. 8. Создание своего тега

В данном случае на странице будет создан тег картинки. В качестве ссылки на изображение используется заведомо ошибочная строка. Исходя из этого, когда пользователь перейдет по вредоносной ссылке, заданный атрибут «onerror» отработает при загрузке страницы и покажет пользователю уведомление.

Учитывая опасность приведённых видов XSS-уязвимостей, следует выделить несколько способов предотвращения XSS-атак, а именно, кодирование и валидация вводимых пользовательских данных, политику безопасности контента.

Кодирование вводимых пользовательских данных является одним из способов защиты от XSS-уязвимости. Под кодированием подразумевается процесс, в ходе которого специальные символы заменяются на их закодированный вариант. В данном случае используется кодирование в контексте HTML (табл. 1) и в контексте JavaScript (табл. 2).

Кодирование необходимо осуществлять перед записью пользовательских данных, так как место, в которое будут записаны данные, определяет необходимую кодировку. В результате кодирование позволяет браузеру интерпретировать пользовательский ввод именно как текст, а не как код.

Валидацию входных данных можно рассматривать как ещё одну возможность предотвращения XSS-уязвимости. Валидация – это процесс, в ходе которого происходит проверка полученных от пользователя данных в рамках заданных разработчиком критериев безопасности. Осуществляться она может несколькими способами.

1. Проверка на наличие специальных символов. Если в переданной пользователем строке присутствуют специальные символы, можно обработать данные таким образом, что эти символы будут удалены из исходного текста.

2. Проверка протокола в отправленном запросе. Если запрос пользователя основан на небезопасном протоколе, таком как «javascript», система может отбросить этот запрос.

3. Проверка контекста. Необходимо проверять соответствие введенных пользователем данных, с одной стороны, и данных, которые система ожидает от него – с другой. К примеру, если система на вход ожидает число, то она должна получить именно число, а не строковый литерал.

Таблица 1

Кодирование в контексте HTML

Изначальный текст	Закодированный текст
<	<
>	>
<script>alert()</script>	< script > alert()< /script >

Таблица 2

Кодирование в контексте JavaScript

Изначальный текст	Закодированный текст
<	\u003c
>	\u003e

Результат валидации может быть разным в зависимости от выбранного подхода. Он может представлять из себя как полное отклонение пользовательских данных, так и их преобразование к безопасному виду. Кроме того, валидация может происходить, используя в своей проверке как черные, так и белые списки. Подход с использованием черного списка является менее безопасным относительно белого. Суть использования черного списка заключается в том, что разработчик составляет базу запрещенного ввода. Таким образом, приложение будет пропускать любые данные, за исключением тех, что содержатся в базе. Используя белый список, разработчик составляет базу разрешенного ввода. Как результат, приложение будет отбрасывать любые входящие данные, за исключением тех, что содержатся в базе. Вариант с использованием белого списка намного безопаснее и проще в реализации. У разработчика нет необходимости заносить в базу множество не валидной информации, достаточно указать лишь необходимые для функционирования системы данные. Безопасность же заключается в том, что веб-технологии стремительно развиваются и дополняются новым функционалом, вследствие чего могут появиться новые вредоносные протоколы и/или функции, и, если вовремя не внести их в черный список, злоумышленник сможет ими воспользоваться для компрометации системы.

Политика безопасности контента (Content Security Policy – CSP) представляет собой средство для ограничения пользовательского ввода. Такая политика позволяет настроить контент, используемый на странице таким образом, что разрешается применение данных только из доверенных источников [4]. Таким образом, даже если хакеру удастся произвести внедрение вредоносного кода, используемый им скрипт будет недействителен на данной странице.

Для работы CSP страница веб-сайта должна содержать в себе заголовок «Content-Security-Policy». Содержание же этого заголовка будет отвечать за используемый на странице контент. Пример CSP может выглядеть следующим образом: «Content-Security-Policy: default-src 'self'; script-src 'self'; media-src 'none'; img-src 'self';». В данной политике содержатся следующие требования: аудио и видео файлы запрещены для загрузки, скрипты и изображения могут загружаться только из источника, на котором находится веб-сайт. Весь остальной контент может загружаться только из источника, на котором находится страница.

В данной статье был проведен анализ XSS-уязвимости с основным акцентом на практические примеры её эксплуатации. Были рассмотрены три типа этой уязвимости, их отличия и особенности. В данной работе также рассматривается актуальное влияние угрозы межсайтового выполнения сценариев на безопасность веб-систем. В завершении работы приводятся рекомендации по защите веб-сайта от данной уязвимости.

Библиографический список

1. Чусавитина Г.Н. Управление рисками в проекте web-приложения "Магту.Антиплагиат" / Г.Н. Чусавитина, А.В. Киселев // Актуальные проблемы современной науки, техники и образования: Тезисы 80-й международной научно-технической конференции, Магнитогорск, 18–22 апреля 2022 года. Том 1. – Магнитогорск: Магнитогорский государственный технический университет им. Г.И. Носова, 2022. – С. 413. – EDN CTEJCJ.

2. Меркулов А.С., Лапонина О.Р. Тестирование уязвимостей межсайтового выполнения сценариев (XSS) в веб-приложении онлайн-платежей // International Journal of Open Information Technologies. 2019. №10. URL: https://cyberleninka.ru/article/n/testirovanie-uyazvimostey-mezhsaytovogo-vypolneniya-stsenariev-xss-v-veb-prilozhenii-onlayn-platezhey (дата обращения: 16.10.2023).

3. Лужнова Е.Е. Инструменты защиты от межсайтового скриптинга // Шаг в науку. 2022. №4. URL: https://cyberleninka.ru/article/n/instrumenty-zaschity-ot-mezhsaytovogo-skriptinga (дата обращения: 16.10.2023).

4. Крылов И.Д. Эффективные способы обнаружения и предотвращения XSS-уязвимостей сайтов // StudNet. 2021. №2. URL: https://cyberleninka.ru/article/n/effektivnye-sposoby-obnaruzheniya-i-predotvrascheniya-xss-uyazvimostey-saytov (дата обращения: 16.10.2023).

УДК 004.056.57

АНАЛИЗ ОБХОДА АНТИВИРУСНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ МЕТОДАМИ ОБФУСКАЦИИ

Д.Н. Неклюдов, А.С. Лебедь, У.В. Кузьмина
Научный руководитель: канд. техн. наук, доцент кафедры ИиИБ
У.В. Кузьмина
Магнитогорский государственный технический университет
имени Г.И. Носова, г. Магнитогорск

В данной статье приводится исследование, посвященное анализу различных методов обфускации на возможность обхода антивирусного программного обеспечения. Приведены различные техники, направленные на сокрытие кода вредоносных программ. Проводится оценивание эффективности обнаружения обфусцированных файлов различными антивирусными программами. Рассматриваются тезисы на проблему разработки антивирусного ПО способного детектировать запутанный файл.

Ключевые слова: антивирус, виртуализация кода, вирус, вредоносное программное обеспечение, информационная безопасность, обфускатор, обфускация, угроза безопасности информации, упаковка кода.

Многие специалисты в сфере информационной безопасности утверждают, что методы обфускации вредоносных программ не являются ультимативным способом обойти проверку на обнаружение антивирусными средствами защиты.

В основе антивирусного программного обеспечения лежат три подхода:

1. Статический анализ сигнатур

Построен на сравнении программы с имеющейся базой данных вредоносов. Сигнатура может быть основана на конкретном коде, на его части или же на первых исполняемых байтах вредоносного файла. Главная проблема статистического анализа заключается в необходимости постоянного обновления сигнатур, что в свою очередь может затруднить обнаружение нового malware.

2. Статический эвристический анализ

Заключается в поиске определенных паттернов кода, которые характерны для вредоносных программ. Преимуществом этого подхода является возможное обнаружения нового вредоноса, который еще не был зафиксирован в сигнатурах. Правила определения паттернов держатся в секрете из-за чего может быть не очевидным почему антивирус считает программу вредоносной. Из этого следует, что недостатком является генерация методом ложных срабатываний. Путем обхода является сокрытие вредоносного кода.

3. Динамический анализ

Основан на изучении поведения файла в виртуальной среде. В наши дни всё больше антивирусов полагаются на динамический подход. В то же время большинство malware имеют проверку на виртуализацию, по результатам которой она может себя не проявить.

Исходя из этого злоумышленники используют следующие методы сокрытия:

- Упаковка
- Инъекция кода
- Обфускация

Упаковка заключается в сжатии исполняемого файла, а затем упаковку файла в код, необходимый для распаковки во время выполнения.

Инъекция кода включает в себя группу техник: process injection, подразумевающий загрузку кода в память проверенного процесса, что позволяет работать от имени этого приложения; process ghosting, позволяющий записывать на диск malware таким образом, чтобы его было тяжело обнаружить, после чего злоумышленник может реализовать удаленный запуск; process hollowing, дающий возможность загрузки доверенного процесса, который будет выступать оболочкой для вредоносного кода.

Обфускация – способ, позволяющий затруднить чтение и понимание кода, не влияя на его функциональность и работоспособность. Этот метод является наиболее часто используемым за счет его дешевизны и малой ресурсоемкости [1].

Процесс обфускации осуществляется на двух уровнях представления кода – низшем (операции над ассемблерным кодом программы или непосредственно над двоичным файлом программы) и высшем (операции над исходным кодом программы, реализованном на языке высокого уровня).

Разделяют несколько способов модификации кода:

- лексическая обфускация – форматирование кода программы и модификации его структуры таким образом, чтобы код стал нечитаемым, а в идеале неинформативным (например, дезинформирующий комментарий, замена названий переменных, добавление ненужных операций, пробелов и так далее);
- обфускацию структур данных – подразделяется на 3 вида: хранения (создание и использование непривычных типов данных), соединения (объединение переменных, реструктурирование массива) и переупорядочивания (изменение очередности объявления переменных);
- обфускацию потока управления – изменение последовательности выполнения кода (добавление: мертвого, избыточного, недостижимого кода; клонирование функций; устранение ссылок на библиотеки);
- обфускация преобразованием – перевод кода программы в двоичное, шестнадцатеричное представление и так далее;

• обфускация шифрованием – наложение ключа на биты кода программы используя различные алгоритмы (например, XOR) [2];

Стоит отметить отдельным пунктом возможность обфусцировать код при помощи виртуализации. Этот метод подразумевает написание собственного байт-кода. Для того чтобы восстановить поток управления в программе, которая подверглась виртуализации, необходимо проанализировать каждый опкод.

Большинство этих методов реализуют специальные утилиты, именуемые обфускаторами.

Для анализа возможности обхода антивирусного ПО было выбрано несколько инструментов для обфускации. Все инструменты были отобраны на площадке github исходя из их функциональных возможностей. А также был написан тестовый keylogger на языке Python. На основе программы был создан исполняемый файл. Предварительная проверка на VirusTotal показала следующие результаты (рис. 1).

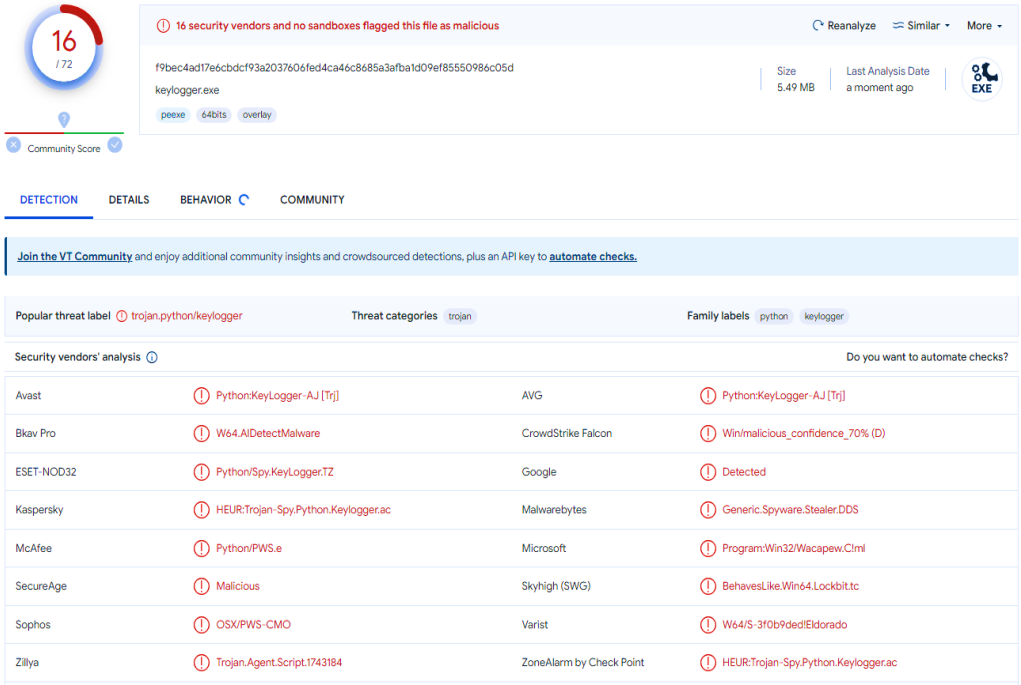


Рис. 1. Анализ исполняемого файла при помощи VirusTotal

Тестовый файл без обфускации был обнаружен 16 вендорами из 72.

Первая утилита, при помощи которой была проведена обфускация keylogger - Intensio-Obfuscator. Данная утилита представляет собой программу, написанную на языке python для .py файлов. Методы обфускации, реализуемые этой программой достаточно просты. Были использованы флаги –rth (replace string hex – преобразование кода программы в hex представление) –rts (replace string to string mixed – замена названий переменных, функций, классов на случайные строковые значения). Также утилита удалила все комментарии, пояснения и пробелы в программе и привела код к одной исполняемой строке.

Результаты сканирования приведены на рис. 2. Стоит отметить, что, используя такие простые методы удалось обойти на 5 антивирусов больше. Так же, если при проверке исходного файла антивирусы точно идентифицировали вредоносное ПО (вероятнее всего использовался эвристический анализ), в этом случае сработал динамический метод анализа поведения malware.

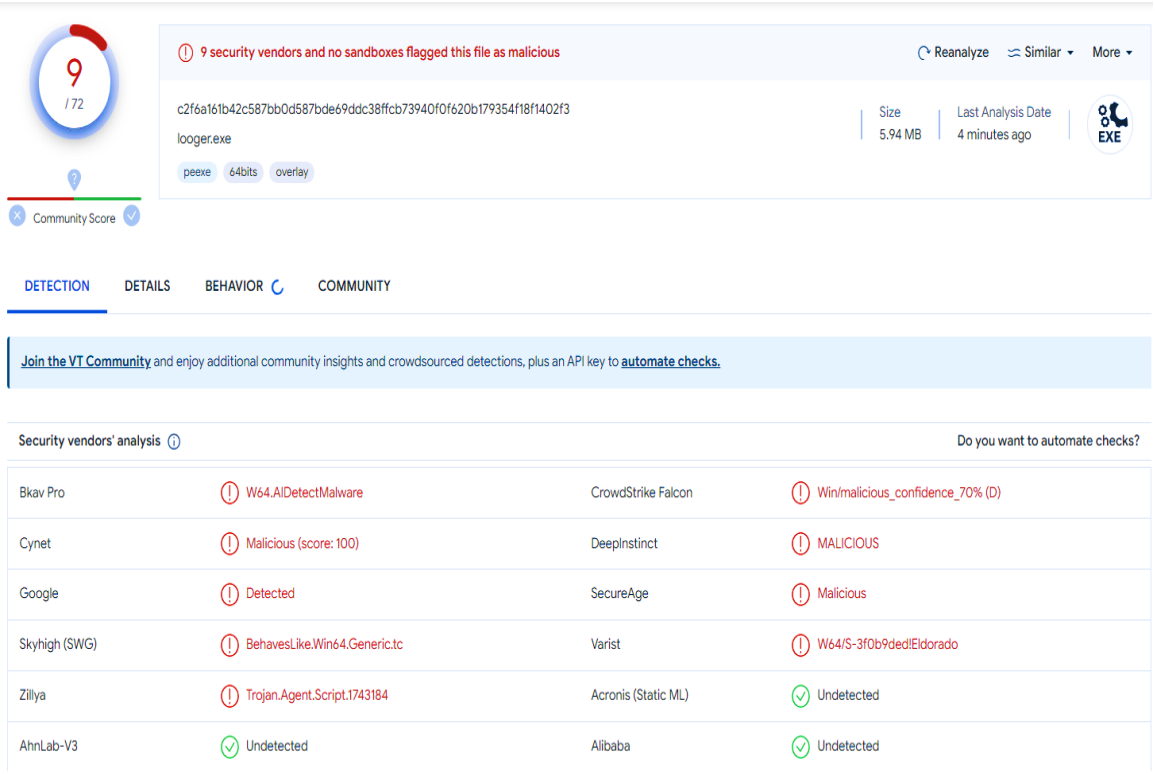


Рис. 2. Проверка VirusTotal после обфускации Intensio-Obfuscator

Следующей утилитой была выбрана Vare-Obfuscator. Исходя из анализа открытого кода, программа выполняет следующие действия:

- Исходный код скрипта компилируется в байт-код с помощью функции compile();
- Байт-код сериализуется с использованием marshal;
- Данные кодируются в формат base64 и сжимаются с использованием zlib;
- При помощи библиотеки Fernet генерирует случайный ключ и шифрует сжатый байт-код;
- Данные разбиваются на несколько частей и разбавляются мусорным кодом;
- Так же существует возможность добавления цифровой подписи при помощи SigThief (генерирует устаревшие сертификаты – неактуально)

После обфускации исходного кода была проведена очередная проверка в VirusTotal результаты которой оказались неутешительны (рис. 3). Вредоносное ПО было детектировано лишь пятью антивирусами.

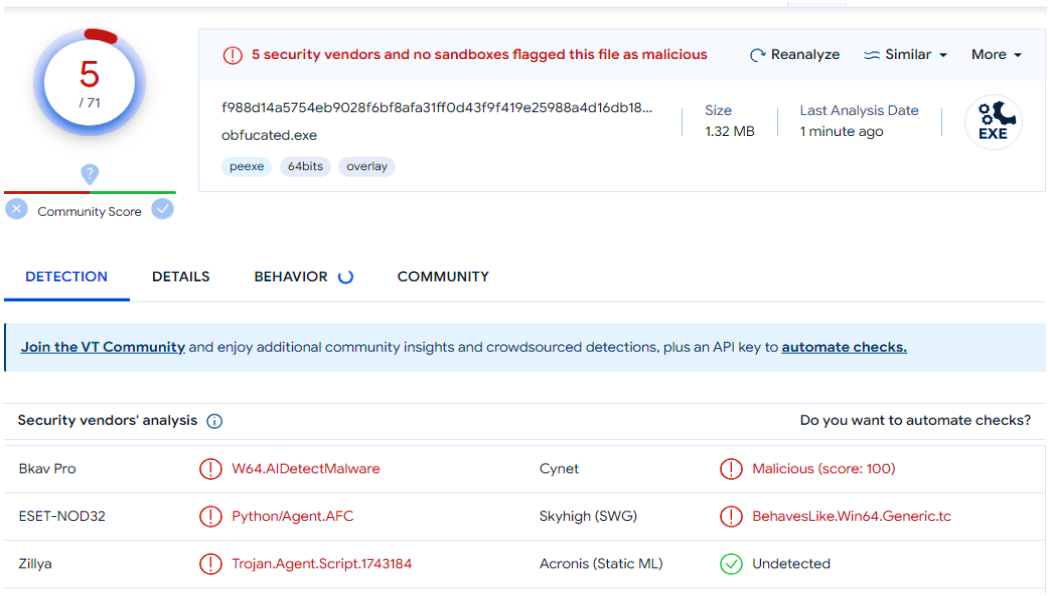


Рис. 3. Проверка VirusTotal после обфускации Vare-Obfuscator

В ходе исследования была выявлена корреляция обнаружения вредоносных исполняемых файлов, что натолкнуло на попытку замены .exe файлов .dll библиотеками. Было решено переделать обфусцированный скрипт под библиотеку формата .pyd. По результатам VirusTotal файл не прошел лишь одну проверку (рис. 4).

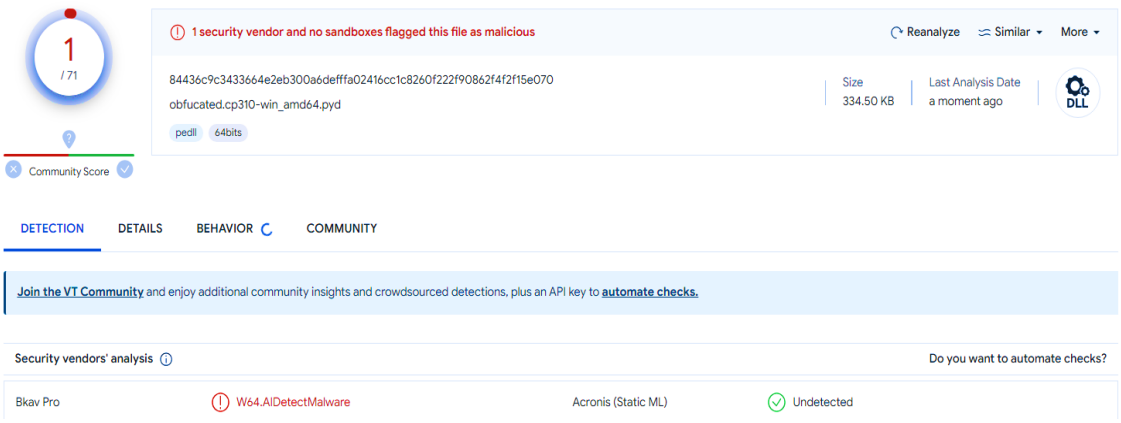


Рис. 4. Проверка VirusTotal обфусцированного файл формата .pyd

Несмотря на то, что с каждым шагом исследования обнаружение антивирусами обфусцированных файлов ухудшалось, метод множественного наложения разных типов обфускации не привело к улучшению обхода и файл распознавался одним антивирусом. Слишком глубокая обфускация может сделать программу неэффективной и неустойчивой, что также может помочь обнаружению антивирусными программами.

Был проведен ряд последующих тестирований других инструментов по обфускации, результаты которых приведены на рис. 5. Для выборки были отобраны антивирусы, которые детектировали файл хотя бы раз на протяжении всего тестирования.

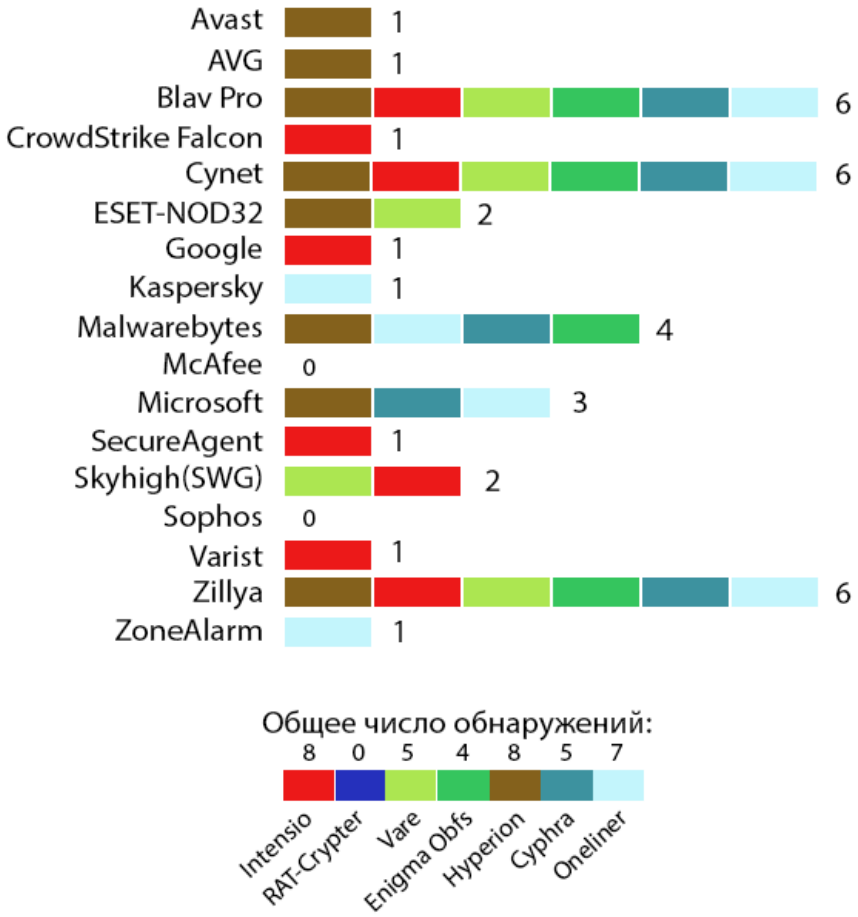


Рис. 5. Результаты тестирования на обнаружение обфусцированного файла

Лучше всего себя показали три антивируса: Bkav Pro, Zillya, Cynet. Однако Cynet не детектировал исходный файл (вероятно, есть проверка на обфускацию).

Среди тестируемых инструментов для обфускации хорошо показали себя: Vare, EnigmaObfs. Cyphra. Лучший результат RAT-Crypter, так как обфусцированный им файл не был опознан ни одним из антивирусов. Он работает по следующему алгоритму:

- Генерирует ключ AES из пароля, используя функцию PBKDF2HMAC для укрепления пароля и создания криптографически стойкого ключа.
- Генерирует пару ключей RSA (приватный и публичный).
- Зашифровывает вредоносный файл с использованием ключа AES и записывает зашифрованные данные.
- Зашифровывает ключ AES с использованием публичного ключа RSA.
- Кодирует зашифрованные данные и ключи в формате Base64.

В ходе исследования было доказано, что даже при использовании исключительно простых Open-Source решений по обфускации, злоумышленник способен полностью обойти антивирусную защиту, что ставит под вопрос их эффективность как таковых. Несмотря на постоянные обновления сигнатур, разработку новых алгоритмов анализа поведения процессов в системе, предпринятых мер оказывается недостаточно, чтобы обезопаситься в полной мере.

Исходя из данного вывода, следует обратиться к более глубоким методам анализа файлов. Например, рассмотрение метода обнаружение запутанных программ путем рекурсивного дизассемблер обхода, который извлекает граф потоков бинарных файлов.

Библиографический список

1. Калугина О.Б., Кириченко Н.С. Практика применения обфускации программного кода / Сбродова Е.А., М.А. Загребин // Безопасность информационного пространства. 2019. Т.1. С. 218–222.

2. Пушнов Ю.А. Анализ методов обфускации вредоносного программного обеспечения / Всяких М.В. // Научные вести. 2019. Т.2. С. 166–171.

УДК 004.056

ПОРЯДОК ОФОРМЛЕНИЯ И ВЫДАЧИ ЭЛЕКТРОННОЙ ПОДПИСИ ПОЛЬЗОВАТЕЛЯМ В УДОСТОВЕРЯЮЩЕМ ЦЕНТРЕ

М.Н. Дмитриев
Научный руководитель: ст. преподаватель кафедры ИТиЗИ
Е.А. Гузенкова
Уральский государственный университет путей сообщения,
г. Екатеринбург

Использование электронной подписи становится все более распространенным в различных сферах жизни, ведь она позволяет подписывать и передавать документы онлайн, без необходимости физического присутствия, обеспечивает целостность данных и неотказуемость. В статье рассмотрены шаги, которые принимает оператор удостоверяющего центра для выдачи электронной подписи пользователям, какие токены используются, какие считаются наиболее надежными и как ведется учет средств криптографической защиты информации.

Ключевые слова: электронная подпись, аккредитованный удостоверяющий центр.

В современном цифровом мире, где электронная коммерция и прочие онлайн-сервисы стали неотъемлемой частью жизни каждого человека, во-

просы безопасности и подлинности данных становятся все более актуальными. Особенно важным является обеспечение конфиденциальности и достоверности информации при использовании электронной подписи.

Для оформления электронной подписи пользователь обращается в удостоверяющий центр и предоставляет необходимые документы, подтверждающие его личность и право на получение электронной подписи. В рамках данного процесса производится проверка и аутентификация пользовательской информации, а также проводятся дополнительные меры для обеспечения безопасности во время самого процесса выдачи.

Оформление и выдача электронной подписи пользователям в удостоверяющем центре является процедурой, направленной на создание безопасного и надежного цифрового окружения для всех пользователей компании и обеспечения их электронной безопасности и доверия к электронным сделкам и операциям.

Федеральный закон от 06.04.2011 № 63-ФЗ «Об электронной подписи» [1] вступил в силу с 8 апреля 2011 года. Закон предусматривает три вида электронных подписей: «простая электронная подпись; усиленная неквалифицированная электронная подпись; усиленная квалифицированная электронная подпись». Одним из условий действительности квалифицированной электронной подписи в российском законодательстве является факт выдачи такой подписи аккредитованным удостоверяющим центром, аккредитация которого действительна на день выдачи указанного сертификата. Аккредитация удостоверяющего центра обозначает признание уполномоченным федеральным органом соответствия данного центра требованиям Федерального закона «Об электронной подписи». Она предполагает выполнение удостоверяющим центром определенных организационно-технических и экономических требований [2].

Перед началом работ оператором удостоверяющего центра (далее – УЦ) необходимо обеспечить выполнение следующих мероприятий:

1. Программное обеспечение «КриптоПро УЦ 2.0» [3] должно быть установлено и готово к эксплуатации в штатном режиме функционирования.

2. Оператор УЦ должен иметь ключевой носитель, предоставленный пользователем. Пользователь должен предоставить оператору УЦ ключевой носитель, соответствующий требованиям информационной безопасности и сертифицированный во ФСТЭК России. Ключевой носитель учитывается в журнале установленной формы.

На автоматизированном рабочем месте (далее – АРМ) оператора УЦ должны быть установлены и настроены средства защиты информации (антивирус, средство контроля доступа, резервное копирование и т.д.), а также настроены параметры подключения к Центру регистрации.

Для запуска консоли управления центра регистрации (далее консоль управления ЦР) необходимо запустить приложение «Консоль управления

ЦР» в меню «Пуск», в центральном окне нажать «Подключиться» (рис. 1) и убедиться, что подключение к Центру регистрации успешно установлено (рис. 2).

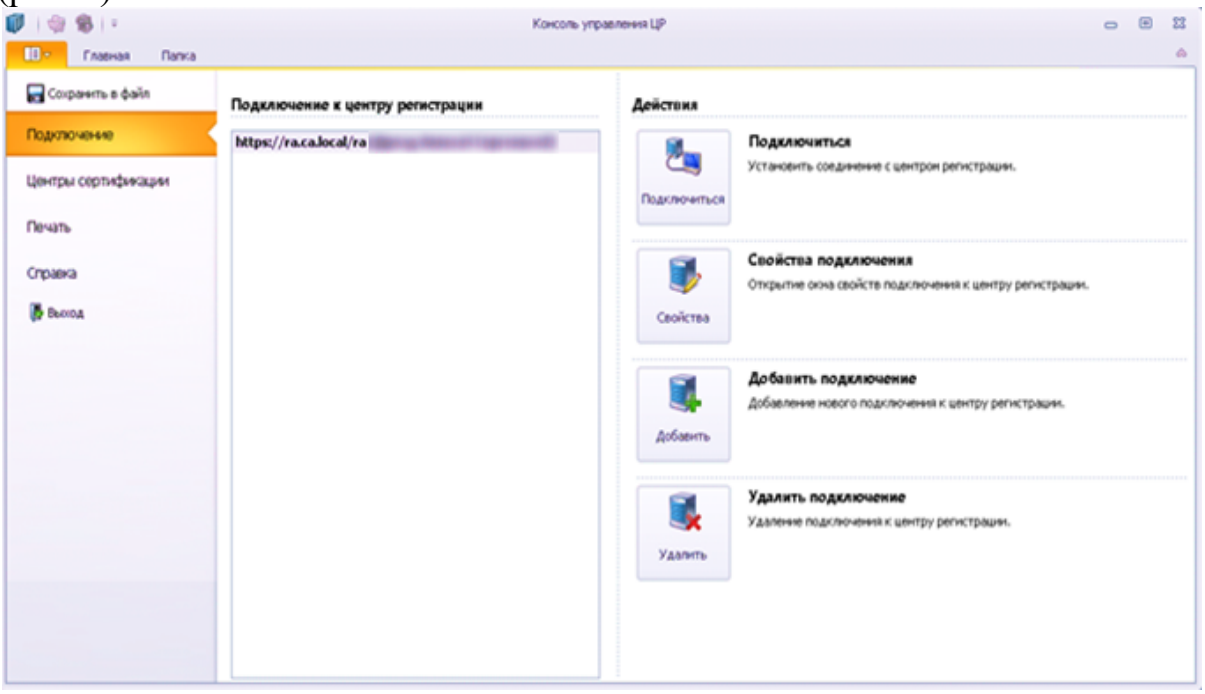


Рис. 1. Стартовое окно Консоли управления ЦР

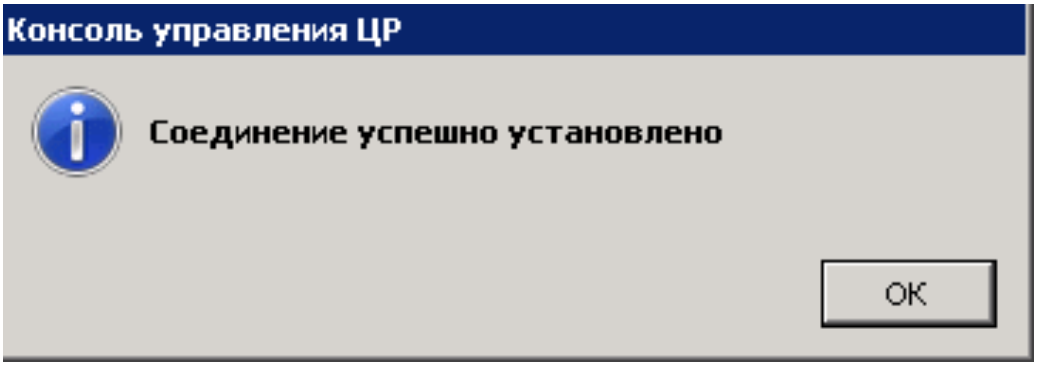


Рис. 2. Соединение успешно установлено

Для регистрации пользователя необходимо выполнить следующие действия:

– Запустить Консоль управления ЦР и перейти в раздел «Пользователи» (рис. 3).

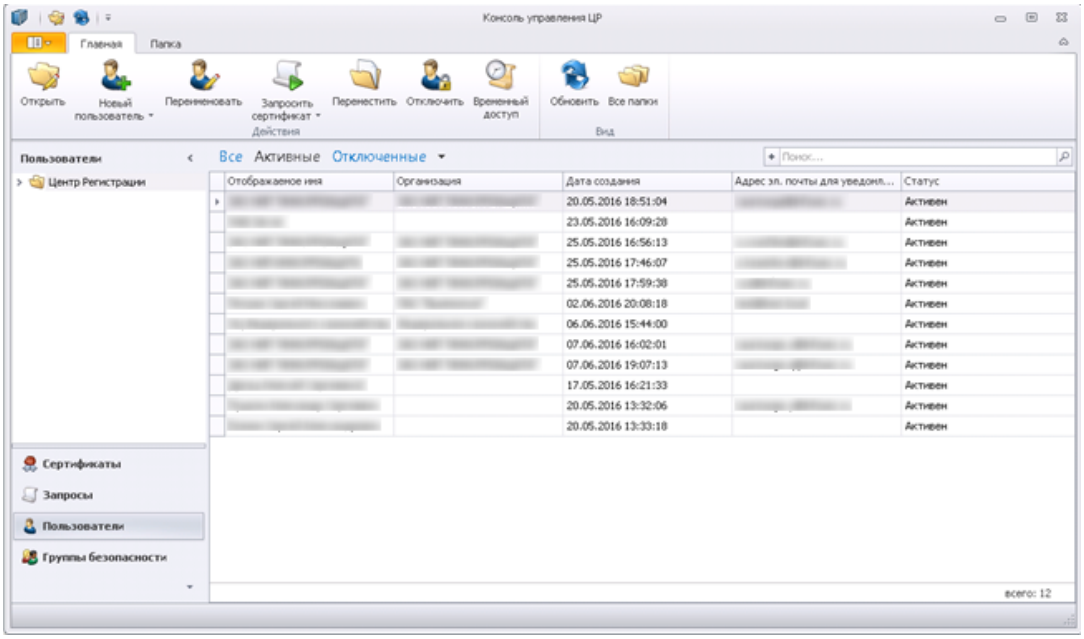


Рис. 3. Вкладка «Пользователи»

– Для создания нового пользователя необходимо в верхней части консоли нажать кнопку «Новый пользователь» (рис. 4).

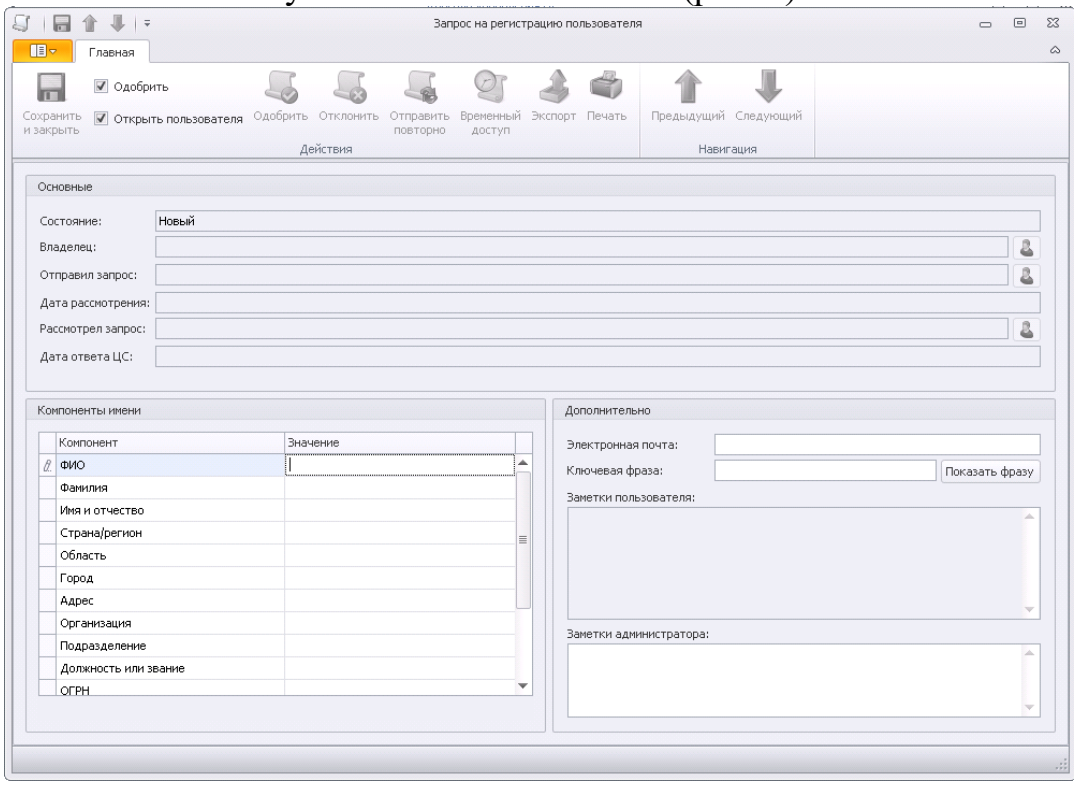


Рис. 4. Создание запроса на регистрацию пользователя

– Ввести данные о пользователе, отметить чекбокс «Одобрить», нажать кнопку «Сохранить и закрыть» и убедиться, что созданная учётная запись появилась в списке пользователей.

Для выпуска сертификата пользователю УЦ необходимо выполнить следующие действия:

– Запустить Консоль управления ЦР и перейти в раздел «Пользователи» (рис. 4).

– Выбрать пользователя, которому необходимо выпустить сертификат, и щелчком правой кнопкой мыши по выбранному пользователю нажать «Запросить сертификат → Запросить сертификат» (рис. 5).

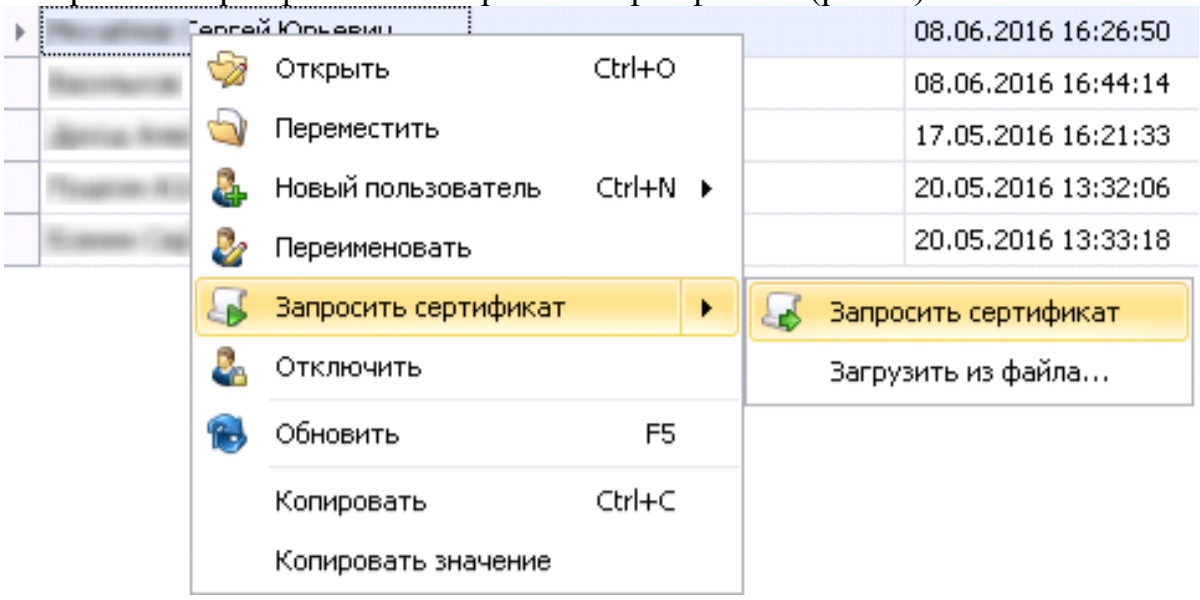


Рис. 5. Переименование пользователя

– Подключить ключевой носитель пользователя к АРМ Оператора УЦ, выбрать шаблон сертификата для создания (выделен красным), установить чекбокс «Одобрить» и нажать «Создать и закрыть» (рис. 6).

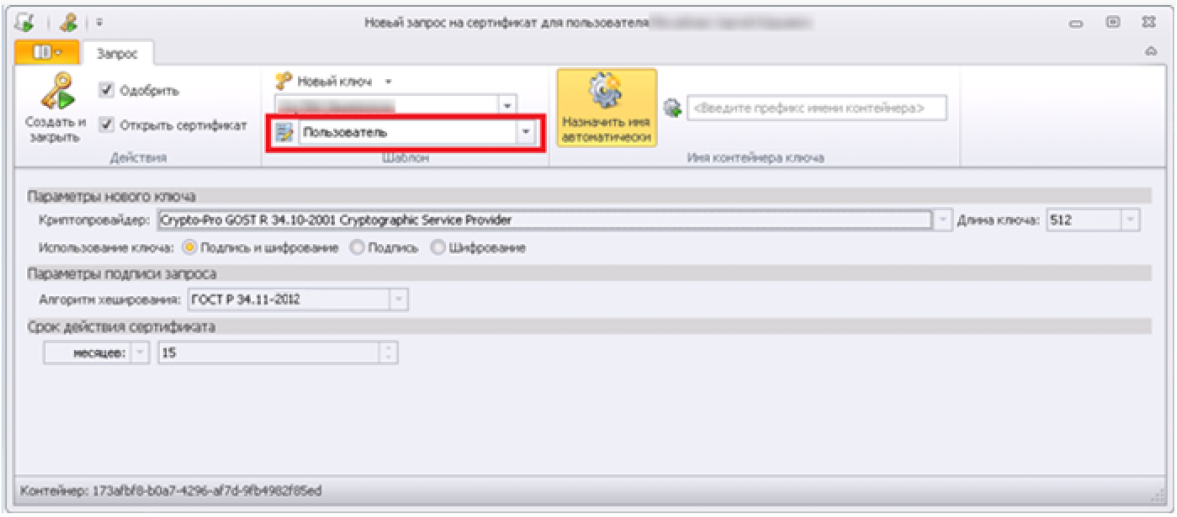


Рис. 6. Создание запроса на сертификат

– Перейти на вкладку «Сертификаты» и убедиться, что пользователю был выпущен сертификат (рис. 7).

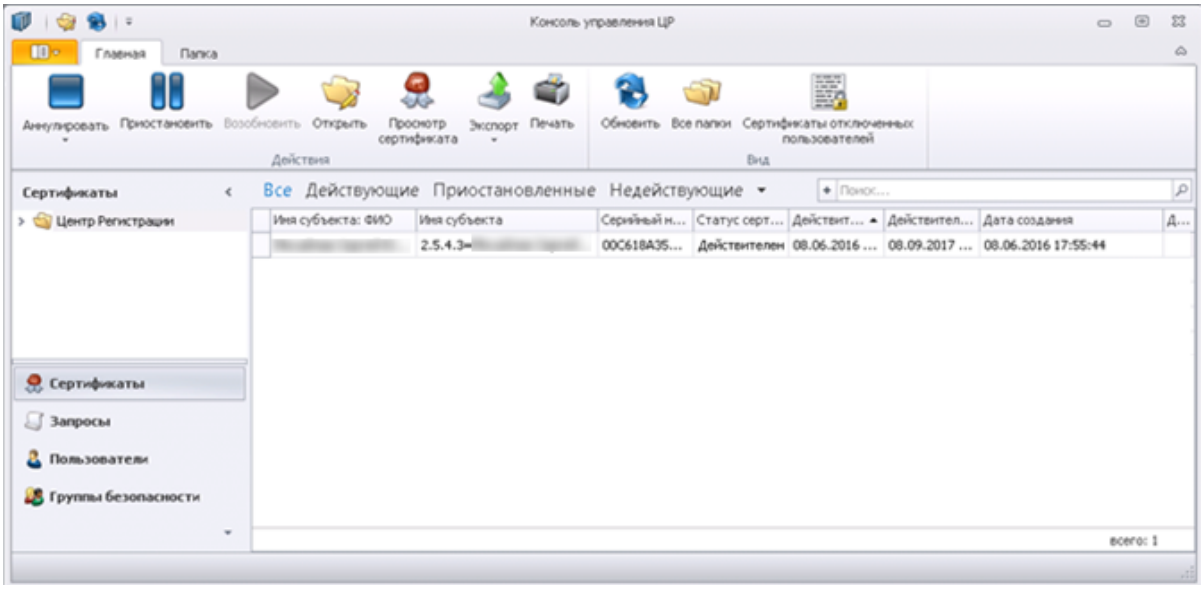


Рис. 7. Сертификаты пользователей

После того, как сертификат пользователя успешно выпущен, необходимо распечатать его в количестве 2 шт. На каждом сертификате пользователя указаны его персональные данные [4], а также на одном из них будет указан номер ключа (его также присваивает оператор) и сгенерированный ПИН-код пользователя. Затем производится заполнение журнала учета СКЗИ и выдача пользователю под роспись одного из 2-х экземпляров сертификатов. Далее пользователь расписывается в журнале с заполненными данными. Один экземпляр заявления остается в УЦ и убирается в архив (рис. 8).

ЖУРНАЛ
учета СКЗИ для выдачи сертификатов ЭП в подразделении
(ведется уполномоченным представителем, пример заполнения)

N п.п.	Наименование криптосредства, эксплуатационной и технической документации к ним, ключевых документов	Регистрационные номера СКЗИ, эксплуатационной и технической документации к ним, номера серий ключевых документов	Номера экземпляров (криптографические номера) ключевых документов	Отметка о получении		Отметка о подключении (установке) СКЗИ		
				От кого получены или Ф.И.О. сотрудника органа криптографической защиты, изготовившего ключевые документы	Дата и номер сопр. пись-ма	Ф.И.О. пользователя криптосредств, производившего подключение (установку)	Дата подключения (установки) и подписи лиц, произведших подключение (установку)	Номера аппаратных средств, в которые установлены или к которым подключены крипто-средства
1	2	3	4	5	6	7	8	9
	Тип носителя	Серийный номер носителя (+номер, наклеенный на ключе)	Номер сертификата	ФИО уполномоченного представителя, подпись				
	ruToken/eToken/feitian / JaCarta (если Вы не знаете какой у Вас тип носителя – обратитесь за консультацией к операторам УдЦ)	Серийный номер ключа указан производителем на корпусе ключа. Если его нет, то пишется только номер с наклейки. Если серийный номер есть, то пишется он, и рядом в скобках номер с наклейки.	Серийный номер сертификата указан в бумажном экземпляре, выданном в УдЦ. Состоит из 20 символов.	ФИО уполномоченного представителя, передающего документы пользователю				
пример	ruToken	916B028086 (1234)	65AD 00R6 0000 0000 1234	Иванов И.И.				

Рис. 8. Журнал учета СКЗИ

Конкретный выбор между токенами при покупке и оформлении электронной подписи (далее ЭП) зависит от следующих факторов:

1. Специфика применения (необходима ли дополнительная Flash-память, будет ли использоваться токен в системах контроля доступа);

2. Личные предпочтения в плане внешнего вида и бренда.

Одновременно для обеспечения информационной безопасности, предотвращения потери данных важно учитывать следующие моменты:

1. Приобретайте токен и ЭП в авторизованных удостоверяющих центрах;

2. Поддерживайте на должном уровне грамотность персонала, который будет работать с токеном;

3. Соблюдайте перечень простейших правил хранения ключей для предотвращения их хищения, компрометации паролей.

3.1. Не делитесь своими ключами или паролями с посторонними лицами.

3.2. Используйте сложные и уникальные пароли.

3.3. Избегайте хранения ключей и паролей в открытом виде.

Типы возможных ключевых носителей представлены в табл. 1.

Таблица 1

Типы ключей

Тип носителя	Сертифицирован по требованиям ФСБ и ФСТЭК	Цвет корпуса	Название устройства в системе	Внешний вид
Rutoken	Да	Красный	rutoken magistra/magistra	
Feitian	Да	Черный или серебристый	feitian scr301/magistra	
Etoken	Да	Фиолетовый	AKS ifdh 0	

В заключение, важно отметить, что одним из основных преимуществ электронной подписи является защита от подделки и изменения информа-

ции. Подпись позволяет получателю убедиться в том, что документ не был изменен после его создания, а отправитель не может отказаться от авторства своего сообщения. Это особенно важно в сфере финансов, правовой и медицинской документации, где целостность и подлинность информации имеют критическое значение.

Кроме того, электронная подпись повышает эффективность и удобство процессов, которые ранее требовали подписи на бумажных документах. Она позволяет заключать сделки и соглашения удаленно, без необходимости физического присутствия сторон или курьерской доставки документов. Это экономит время и ресурсы, способствует более быстрой и удобной коммуникации.

Библиографический список

1. Федеральный закон "Об электронной подписи" от 06.04.2011 N 63-ФЗ // URL: https://www.consultant.ru/document/cons_doc_LAW_112701/ (дата обращения: 16.10.2023).

2. Журнал «Вестник УрФО. Безопасность в информационной сфере» // URL: https://www.info-secur.ru/index.php/ojs/issue/archive (дата обращения: 17.10.2023).

3. Программное обеспечение КриптоПро УЦ 2.0 // URL: https://www.cryptopro.ru/products/ca/2.0 (дата обращения: 17.10.2023).

4. Федеральный закон "О персональных данных" от 27.07.2006 N 152-ФЗ // URL: https://www.consultant.ru/document/cons_doc_LAW_61801/ (дата обращения: 21.10.2023).

УДК 004.728.8

СЛОЖНОСТИ РАЗВЕРТЫВАНИЯ КОНТЕНТ-ФИЛЬТРОВ

Г.А. Новиков, А.А. Кузьмин, У.В. Кузьмина
Научный руководитель: канд. техн. наук У.В. Кузьмина
Магнитогорский государственный технический университет имени Г.И. Носова, г. Магнитогорск

Статья рассматривает контент-фильтры как устройства или ПО для блокирования доступа к определенным сайтам или услугам в Интернете на основе их содержимого. Она фокусируется на технологии Deep Packet Inspection (DPI) для решения проблем контентной фильтрации. В статье приведен анализ архитектуры DPI и его компоненты, такие как компоненты захвата пакетов, анализа, фильтрации и управления. Рассмотрены различные техники и алгоритмы DPI, такие как сигнатурное сопоставление, глубокий анализ содержимого, машинное обучение и поведенческий анализ. В статье также освещаются сложности настройки DPI системы, включая масштабируемость и производительность,

конфиденциальность данных пользователей, ложные срабатывания и пропуски, анализ зашифрованного трафика через протокол HTTPS. В настоящее время DPI-технологии актуальны в связи с развитием этой технологии в масштабе страны Роскомнадзором.

Ключевые слова: deep packet inspection, анализ зашифрованного трафика, архитектура dpi, блокировка сайтов, глубокий анализ содержимого, компоненты dpi, контент-фильтр, конфиденциальность, ложные срабатывания, масштабируемость, производительность, сигнатурное сопоставление.

Введение

Контент-фильтр – это устройство или программное обеспечение для фильтрации сайтов по их содержимому. Такие системы не позволяют получить доступ к определённым сайтам или услугам сети Интернет. Система проверяет содержимое страницы на отношение к запрещённому или потенциально вредоносному контенту. По логике работы таких систем контент должен не отображаться, производить редирект всей страницы на безопасную часть контента или выдавать ошибку. Особенностью таких систем в отличии от блокирования по IP или URL заключается в более гибкой и адаптивной фильтрации контента, так как контент блокируется не на определённый адрес, а на содержание. Для решения проблем контентной фильтрации используется технология Deep Packet Inspection (DPI) [1].

1. Технология DPI

1.1. Архитектура DPI

Архитектура DPI состоит из нескольких ключевых компонентов, которые работают в совместной сети для обеспечения анализа и фильтрации пакетов данных. Основные компоненты архитектуры DPI включают в себя:

- Компонент захвата пакетов: отвечает за сбор и захват пакетов данных из сетевого трафика.
- Компонент анализа: производит глубокий анализ содержимого пакетов, используя различные техники и алгоритмы DPI.
- Компонент фильтрации: осуществляет фильтрацию пакетов на основе заданных правил и политик безопасности.
- Компонент управления: обеспечивает управление и конфигурацию DPI системы, включая настройку правил фильтрации и мониторинг сетевого трафика.

1.2. Техники и алгоритмы DPI

В DPI применяются различные техники и алгоритмы для анализа и фильтрации пакетов данных. Некоторые из них включают:

- Сигнатурное сопоставление: использует заранее определенные сигнатуры для идентификации конкретных приложений или протоколов.

- Глубокий анализ содержимого: анализирует содержимое пакетов на более высоком уровне, позволяя идентифицировать конкретные типы контента или поведение приложений.
- Машинное обучение: применение алгоритмов машинного обучения для обнаружения и классификации новых или неизвестных типов трафика.
- Поведенческий анализ: анализирует поведение сетевого трафика и идентифицирует аномалии или подозрительные активности.

2. Сложности настройки

2.1. Масштабируемость и производительность

Настройка DPI системы может столкнуться с проблемами масштабируемости и производительности. Обработка большого объема сетевого трафика требует высокой производительности оборудования и оптимизации алгоритмов DPI. Кроме того, при увеличении количества пользователей и приложений в сети могут возникать сложности с обеспечением достаточной пропускной способности и быстрой обработкой пакетов данных.

2.2. Конфиденциальность

Настройка DPI системы может вызывать вопросы о конфиденциальности данных пользователей. Глубокий анализ содержимого пакетов может потенциально нарушать приватность пользователей и вызывать опасения относительно сбора и использования личной информации. Поэтому необходимо учитывать соответствие DPI системы нормам и законодательству в области защиты данных и конфиденциальности.

2.3. Ложные срабатывания и пропуски

Настройка DPI системы может столкнуться с проблемами ложных срабатываний и пропусков. Ложные срабатывания возникают, когда система неправильно идентифицирует или блокирует легитимный трафик, что может привести к недоступности определенных приложений или сервисов. Пропуски возникают, когда система не обнаруживает или не блокирует нежелательный трафик. Решение этих проблем требует тщательной настройки правил фильтрации и постоянного обновления сигнатур и алгоритмов DPI.

2.4. Анализ зашифрованного трафика

Анализ зашифрованного HTTPS-трафика представляет сложности для DPI системы из-за использования шифрования SSL/TLS. При передаче данных через HTTPS, содержимое пакетов зашифровано и не доступно для прямого анализа DPI системой. Это означает, что DPI не может прочитать и проанализировать содержимое пакетов, так как оно остается зашифрованным. Это создает ограничения для DPI в обнаружении и фильтрации нежелательного контента или вредоносных действий внутри зашифрованного трафика. Например, если внутри зашифрованного HTTPS-трафика находится вредоносный код или запрещенный контент, DPI система не сможет обнаружить его, так как не имеет доступа к содержимому пакетов.

Однако, хотя DPI не может прямо анализировать зашифрованный трафик, существуют некоторые методы, которые могут помочь в анализе HTTPS-трафика. Один из таких методов – это расшифровка и инспекция SSL/TLS. При использовании этого метода DPI система может расшифровать зашифрованный трафик, чтобы получить доступ к содержимому пакетов и проанализировать его. Однако, для успешной расшифровки и инспекции SSL/TLS, DPI система должна иметь доступ к соответствующим сертификатам и ключам, которые используются для шифрования и аутентификации HTTPS-соединения [2].

3. Исследование принципов блокировки HTTPS трафика

Для анализа и блокировки HTTPS трафика нам потребовался прокси-сервер, который может выполнять функции DPI. Примеры таких прокси-серверов включают Squid, mitmproxy, и другие.

Эти инструменты работают путем выполнения "man-in-the-middle" (MitM) атаки: они декодируют HTTPS трафик, анализируют его, а затем снова шифруют перед отправкой к конечному получателю. Это требует установки специального сертификата на всех клиентских машинах, чтобы они доверяли прокси-серверу [3].

Простая схема для настройки DPI состоит из двух компонентов (рис. 1).

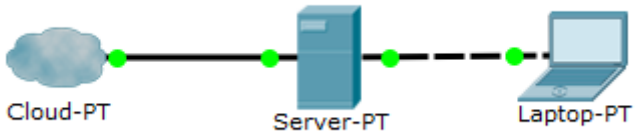


Рис. 1. Логическая схема сети

В этом случае Server является прокси сервером с настроенным squid, а Laptop конечным устройством пользователя, с которого будут идти запросы.

3.1. Правила *iptables*

Интерфейс *iptables* анализирует пакет текстов с помощью модуля *string*. Поэтому можно сделать некий примитивный DPI для блокировки определенного текста в пакетах. Вот пример команды, которую мы использовали для блокировки трафика, содержащего слово "blabla" (рис. 2).

```
┌──(root㉿kali)-[~]
└─# iptables -A INPUT -p tcp -m string --string "blabla" --algo bm -j DROP
```

Рис. 2. Фильтрация с помощью *iptables*

Эта команда добавляет новое правило в цепочку *INPUT*. Она проверяет входящий TCP-трафик на наличие строки "blabla". Если строка найдена, пакет отбрасывается. Обратите внимание, что это очень примитивный метод и может привести к ложным срабатываниям. Например, если слово

"blabla" встречается в нормальном веб-трафике, этот трафик также будет заблокирован.

Также стоит учесть, что этот метод не будет работать с зашифрованным трафиком, таким как HTTPS, поскольку *iptables* не может анализировать зашифрованный контент.

3.2. Правила *SquidGuard*

Команда установки *SquidGuard* (рис. 3).

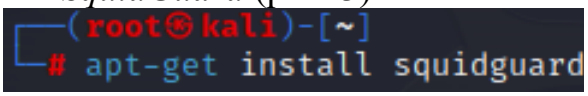


Рис. 3. Команда установки *squidguard*

SquidGuard считывает свою конфигурацию из файла /etc/squidguard/squidGuard.conf. Этот файл представляет собой текстовый файл, содержащий правила для *SquidGuard* [4].

Первым шагом в настройке *SquidGuard* является определение категорий. Категории используются для группировки URL-адресов на основе их содержания. Чтобы определить категории, нам нужно создать файл с именем /etc/squidguard/blacklists/categories. Этот файл должен содержать список категорий, по одной в строке (рис. 4).

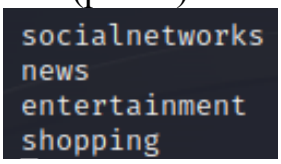


Рис. 4. Список категорий в файле

Определив категории, мы можем создать правила контента. Правила контента используются для определения того, какой контент следует блокировать или разрешить на основе различных критериев, таких как URL-адрес, имя домена, время суток и группа пользователей.

Правила оформления содержимого определены в файле /etc/squidguard/squidGuard.conf (рис. 5).

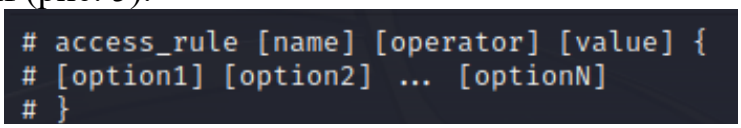


Рис. 5. Правила оформления squidGuard.conf

Например, чтобы заблокировать доступ к Facebook, мы могли бы определить правило контента следующим образом (рис. 6).

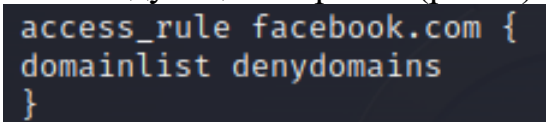


Рис. 6. Заблокированный доступ к Facebook

Чтобы включить фильтрацию контента, нам нужно указать Squid использовать *SquidGuard*. Мы можем сделать это, добавив следующую строку в наш файл конфигурации *Squid* который находиться по пути /etc/squid/squid.conf (рис. 7).

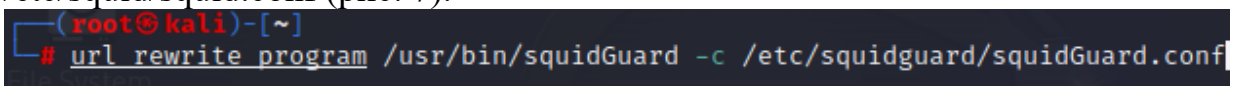


Рис. 7. Включить фильтрацию контента

После настройки правил контента нам нужно включить их (рис. 8).

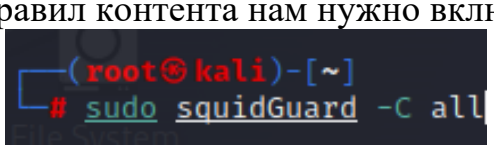


Рис. 8. Включить правила SquidGuard

Эта команда скомпилирует наш файл конфигурации *SquidGuard* и включит наши правила контента.

Заключение. Deep packet inspection – это мощная технология, позволяющая фильтровать контент и обеспечивать безопасность сети. Однако ее настройка и развертывание представляют несколько сложностей. Анализ HTTPS-трафика добавляет дополнительный уровень сложности из-за шифрования. Несмотря на эти проблемы, DPI продолжает развиваться, и ведутся работы по усовершенствованию техник расшифровки SSL/TLS для обеспечения эффективной фильтрации контента в зашифрованном трафике.

В ходе исследования было доказано, что современная спецификация контент-фильтра тяжело осуществима технологиями DPI. Несмотря на это данная технология сейчас развивается в нашей стране и уже сейчас является неплохим фундаментом для суверенного интернета, которому помогает Роскомнадзор, который занимается данной технологией с 2017 года. Конечно, систему фильтрации в национальном масштабе реализовали не так много стран, из удачного примера можно отметить Китай, однако и в Китае система DPI обходится, но со временем все перечисленные недостатки будут решены ввиду развития технологий.

Библиографический список

1. Ёлкин Т. Контент-фильтры и продукты их обитания. / Ёлкин Т. [Электронный ресурс] // Хабр: [сайт]. – URL: https://habr.com/ru/articles/699578/ (дата обращения: 21.10.2023).

2. Deep packet inspection processing market / [Электронный ресурс] // Markets and markets: [сайт]. – URL: https://www.marketsandmarkets.com/Market-Reports/deep-packet-inspection-processing-market-252816977.html (дата обращения: 21.10.2023).

3. Баранкова И.И., Михайлова У.В., Лукьянов Г.И. Подход к проектированию сети предприятия в защищенном исполнении [Текст] / И.И. Баранкова,

У.В. Михайлова, Г.И. Лукьянов // Вестник УРФО. Безопасность в информационной сфере – 2018. – № 1(27). – С. 24–28.

4. Настройка SquidGuard / [Электронный ресурс] // Linux-console: [сайт]. – URL: https://ru.linux-console.net/?p=22530 (дата обращения: 21.11.2023).

УДК 004.056.53

ИНТЕРНЕТ ВЕЩЕЙ В ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Н.А. Байтяков, С.В. Мухачев
Научный руководитель: канд. физ.-мат. наук, доц. С.В. Мухачев
Уральский государственный университет путей сообщения, г. Екатеринбург

Рассматриваются виды кибератак и их влияние на IoT. Приводятся примеры IoT, их характеристики, способы защиты. Анализируются возможные методы защиты, правовая ответственность.

Ключевые слова: защита информации, информационная безопасность, кибератака, неправомерный доступ к информации, угроза информационной безопасности, IoT.

IoT (Internet of Things) – это широко распространенное явление в нашей жизни, которое представляет все большую угрозу безопасности. С появлением разнообразных устройств, способных подключаться к интернету и обмениваться данными, стало сложнее защититься от потенциальных угроз. Внедрение IoT в различные повседневные предметы приводит к инновациям и изменяет наше взаимодействие с окружающим миром. Однако, это также повышает риски информационной безопасности. В статье будет рассмотрено взаимодействие IoT и информационной безопасности, а также возможности и вызовы, связанные с этим.

Безопасность IoT стала важной задачей, так как она охватывает защиту устройств, сети и данных, а также обеспечивает непрерывное функционирование всей системы IoT. Область IoT расширила периметр безопасности, вводя множество подключаемых устройств, которые могут содержать пользовательские данные. Управление безопасностью IoT позволяет анализировать и управлять рисками, обеспечивая общую надежность системы.

Расширение использования устройств Интернета вещей (IoT) в корпоративной среде имеет свои риски для безопасности [1]. Компании сталкиваются с потенциальными уязвимостями данных из-за подключённых устройств, которые находятся за пределами знания большинства ИТ-специалистов. Более того, такие устройства могут вызвать физическое по-

вреждение или повысить риск сбоя в службах, что может иметь серьезные последствия, особенно для медицинского оборудования.

Более того, IoT может использовать технологию блокчейна для дополнительного слоя безопасности. Как распределенный реестр, блокчейн предлагает надежные механизмы для прозрачных, неизменяемых и безопасных транзакций, что значительно повышает доверие к межсетевому взаимодействию внутри сетей IoT.

В основном существует несколько основных признаков Интернет вещей. Первый и самый важный, что любая система IoT необходима считывать данные с окружающей среды с помощью датчиков. Такие устройства измеряют различные изменения среды, а также явления. Наглядный пример такого датчика находится в фитнес браслете, который каждую секунду считывает показания человека или расчет расстояния до ближайшего объекта у автономного транспортного средства.

Еще один важный аспект IoT систем, что некоторые из них могут функционировать без участия человека, для этого они используют вышеприведенные датчики и соединением с сетью. Умный дом может начать выполнять команды, даже если хозяин находится далеко от него. Клапан способен открываться или закрываться в зависимости от показателей, которые сняли датчики [2].

С развитием интернет вещей, производителе часто переходят на более новую версию программного обеспечения, оставляю старую систему без поддержки. Подобные системы имеют много уязвимостей, делая их желанной целью киберпреступников. Злоумышленники могут воспользоваться данными уязвимостями для взлома устройства или сети и получения конфиденциальной информации.

На данный момент не существует единого стандарта безопасности интернет вещей. Это происходит так, как разные производители используют разные методы защиты своих устройств, что является преградой для единого уровня защиты IoT и созданию стандарта.

Важную роль в развитии угроз, связанных с IoT, сыграла социальная инженерия. Злоумышленник использует мошеннические методы, чтобы убедить владельцев сообщит персональную информацию.

Все вышеперечисленные проблемы и угрозы делают информационную безопасность в области IoT актуальной и требующей постоянного внимания и дальнейшего развития безопасности устройств и стандартов. сетевые соединения.

Организация атак на IoT может быть осуществлена следующим образом: для начала киберпреступник попытается перебрать пароли к сервисам, которые используют протокол Telenet, данный протокол часто используется в IoT средах, он передаёт информацию в нешифрованном виде. Если злоумышленнику удалось подобрать пароль, он получает полный контроль над устройством и способен запускать любое вредоносное про-

граммное обеспечение на нем. В случае если устройство использует протокол, который шифрует данные, то потребуется больше вычислительных ресурсов, чем на Telenet [3].

Ботнет IoT – сеть подключенных к интернету устройств (маршрутизаторов), зараженных вредоносным программным обеспечением. Обычно такие устройства применяются для организации атак типа DDoS на интересуемые объекты.

В системе интернет вещей, периодически появляются разные кодовые базы. Одной из первых таких база стала Mirai, которая использовалась для создания ботнетов IoT. Mirai была обнаружена в 2016 году и концентрировала свое внимание на “умном доме”. В основном это были камеры и маршрутизаторы. Данная кодовая база использовала уязвимости в устройствах по умолчанию, таких как слабые пароли и открытые порты, для заражения и добавления их в ботнет [4].

IoT_reaper, обнаруженный в 2017 году, является одной из самых последних баз кода. Он использовал более изощренные методы заражения устройств, включая использование уязвимостей программного обеспечения и использование слабых паролей. Отличительной особенностью IoT_reaper является его способность обновлять и расширять свою функциональность, что делает его более гибким и трудным для защиты.

Persirai специализируется на заражении и использовании уязвимостей в IP-камерах. Он был обнаружен в 2017 году и быстро стал одной из наиболее распространенных кодовых баз для ботнетов Интернета вещей. Persirai использует уязвимости в программном обеспечении IP-камер, чтобы получить к ним доступ и добавить их в свою ботнет-сеть.

Все эти базы кода представляют серьезную угрозу для устройств Интернета вещей и могут быть использованы для DDoS-атак, фишинга, шпионажа и других киберпреступных действий. Регулярные обновления программного обеспечения и меры безопасности, такие как использование сложных паролей и фильтров для сетевых служб, могут помочь предотвратить атаки.

Эти базы кода демонстрируют общие характеристики, такие как использование слабых паролей и открытых портов, а также использование уязвимостей в программном обеспечении устройств. Они также могут расширять свою функциональность и обновляться, чтобы оставаться активными и преодолевать защиту.

Однако, важно понимать, что безопасность Интернета вещей – это непрерывный процесс. Угрозы могут меняться, появляются новые уязвимости, поэтому системы безопасности Интернета вещей должны быть гибкими и обновляемыми. Кроме того, важно, чтобы пользователи сами принимали меры для обеспечения безопасности своих IoT-устройств, такие как смена steam, регулярные обновления программного обеспечения и другие. Только при соблюдении всех этих условий Интернет вещей сможет полно-

стью реализовать свой потенциал и внести значительный вклад в повышение информационной безопасности. С появлением Mirai в 2016 году вредоносное ПО для Интернета вещей (IoT) становилось все более популярным. После публикации исходного кода Mirai были созданы различные модификации Mirai с различными методами атаки и способами получения учетных данных. Киберпреступники используют вредоносное ПО для нейтрализации конкурентов и предотвращения повторного заражения устройств. Одним из методов является блокирование доступа к устройству через брандмауэр. Конфликты между программами разрешаются путем обнаружения и анализа процессов, связанных с портом и памятью. Злоумышленники прерывают объединяющие процессы и удаляют файлы.

В Российской Федерации взлом IoT (Интернета вещей) рассматривается как преступление, и за такие действия применяются нормы уголовного и административного законодательства. Вот некоторые из возможных правовых последствий при взломе IoT:

Уголовные последствия [5]:

Статья 272 УК РФ (Незаконный доступ к компьютерной информации): Лица, взламывающие IoT-устройства с целью несанкционированного доступа к информации, могут быть привлечены к уголовной ответственности по данной статье. Это включает в себя штраф или лишение свободы на срок до двух лет.

Статья 273 УК РФ (Создание, использование и распространение вредоносных программ для ЭВМ и компьютерных баз данных): если в результате взлома IoT-устройств создаются и распространяются вредоносные программы, то виновные также могут быть наказаны уголовно, включая лишение свободы.

Административные последствия [6]:

Статья 6.27 КоАП РФ (Незаконное вмешательство в работу информационных систем): за незаконное вмешательство в работу информационных систем (включая IoT-устройства) наступает административная ответственность, которая может включать в себя штрафы.

Статья 6.31 КоАП РФ (Создание и использование вредоносных программ): По этой статье также предусмотрена административная ответственность за создание и использование вредоносных программ.

Статья 7.27 КоАП РФ (Нарушение законодательства о защите персональных данных): если в результате взлома IoT-устройств были нарушены законы о защите персональных данных, то нарушители могут быть привлечены к административной ответственности.

Эти статьи законодательства применяются в Российской Федерации для борьбы с взломом IoT-устройств. Правовые последствия могут включать в себя уголовное или административное наказание, в зависимости от характера и тяжести преступления, а также его последствий.

Взаимодействие IoT и информационной безопасности представляет собой меч. В то время как он открывает перспективу инновационных способов улучшения безопасности систем, он вводит и новые уязвимости. Важно противодействовать этим проблемам с помощью всеобъемлющего подхода к безопасности, который будет развивать преимущества и сокращать потенциальные риски. Я считаю, что мир IoT движется к упрощению и модификации защиты устройств, ведь с каждым годом все больше производителей разрешают ставить свои ПО для защиты информации, но под ответственность владельца. Простой пример, когда самый обычный пользователь может установить своё ПО на видеокамеру.

По мере того, как мы продолжаем двигаться к полностью связанному миру, роль IoT в формировании траектории информационной безопасности становится все более значимой. Поэтому технологи, регуляторы и пользователи должны пересмотреть и адаптировать свои механизмы цифрового взаимодействия. Поддержание баланса между инновациями и безопасностью в сфере IoT будет неотъемлемым для безопасного и выгодного развития нашего все более взаимосвязанного будущего.

Библиографический список

1. Что такое интернет вещей? Определение и описание // Kaspersky.ru URL: https://www.kaspersky.ru/resource-center/definitions/what-is-iot (дата обращения: 10.10.2023).

2. Обзор угроз для IoT-устройств в 2023 году // securelist.ru URL: https://securelist.ru/iot-threat-report-2023/108088/ (дата обращения: 10.10.2023).

3. Internet of Things (IoT) // trendmicro.com URL: https://www.trendmicro.com /vinfo/us/security/definition/internet-of-things (дата обращения: 10.10.2023).

4. Бухарев Д.А., Вагин С.В., Соколов А.Н. Защита устройств интернет вещей от MIRAI-подобных вирусных программ-червей. // Вестник УрФО. 2018. №4(30). С. 11–19.

5. Уголовный кодекс Российской Федерации от 13.06.1996 № 63-ФЗ // URL: http://www.consultant.ru/ (дата обращения 10.10.2023).

6. "Кодекс Российской Федерации об административных правонарушениях" от 30.12.2001 N 195-ФЗ (ред. от 04.08.2023) СПС КонсультантПлюс (дата обращения 10.10.2023).

УДК 004.056

КЛАССИФИКАЦИЯ УЯЗВИМОСТЕЙ СМАРТ-КОНТРАКТОВ

А.Д. Алексеев, Л.А. Чернятин, И.Р. Зулькарнеев
Научный руководитель: доц. кафедры информационной безопасности И.Р. Зулькарнеев
Тюменский государственный университет, г. Тюмень

При внедрении цифрового рубля могут возникнуть проблемы, связанные с использованием централизованных и децентрализованных технологий, таких как системы на распределенных реестрах. В данной статье рассмотрены основные этапы функционирования и элементы смарт-контрактов. При реализации цифрового рубля были определены два вида проблем использования децентрализованных технологий. Авторами была предложена классификация уязвимостей смарт-контрактов, их логики, языка реализации и используемой инфраструктуры.

Ключевые слова: Ethereum, безопасность смарт-контрактов, блокчейн, уязвимости, цифровой рубль.

Архитектура систем на распределенных реестрах получает все большее распространение как по миру, так и в России, что сокращает административные расходы и повышает общую работоспособность системы. Узлы распределенного реестра располагаются в кредитных организациях и ФНС России, а блокчейн позволяет связать все узлы в одну общую защищенную и одновременно доступную всем реестровую запись [1]. Бизнесу это дает прозрачный и простой механизм помощи при любых финансовых операциях, государству контроль эффективности грантов и иной государственной поддержки, исключения возможности ошибок в выплатах и снижение затрат на содержание государственного аппарата, банкам же предоставляет дополнительные возможности независимой квалификации заемщика. Постепенно, в странах по всему миру вместе с технологиями распределенных реестров происходит активная разработка CBDC (Central bank digital currency), или же собственной цифровой валюты банка страны [2]. Это нужно для более удобного регулирования экономики, прозрачной системы грантов от государства, введения систем для регулярных платежей между юридическими и физическими лицами. В Китае, например, цифровой юань уже давно укрепляет экономику, используется в различных переводах с огромным общим банком транзакций в сумме, тем самым укрепляя себя на мировом рынке [3]. ЦБ РФ также делает шаги по внедрению цифрового рубля, где планируется использовать как централизованные, так и децентрализованные элементы, что в совокупности представляет гибридную мо-

дель цифрового рубля. Его введение в жизнь компаний, банков и обычных граждан будет огромным шагом в финансовой и технологических сферах. Цифровой рубль не является криптовалютой, хоть и использует некоторые технологии из сферы децентрализованных финансов, на которых будет строиться взаимодействие пользователей. Одной из основ цифрового рубля выступают смарт-контракты.

В процессе заключения и функционирования смарт-контрактов выделяют следующие этапы и элементы, представленные ниже (рис. 1).

Рис. 1. Этапы функционирования и элементы смарт контрактов

1. Контрагенты передают условия, которые фиксируются с помощью программного кода в смарт-контракте.

2. При необходимости смарт-контракт может вызывать другие смарт-контракты для выполнения определенных ранее в коде функций и методов.

3. Смарт-контракт передает свой программный код в битовом виде для обработки и исполнения Ethereum Virtual Machine (далее – EVM), являющийся вычислительным механизмом в сети Ethereum, который отвечает за исполнение и развертывание смарт-контрактов и формирует блоки для интеграции в общую сеть блокчейна и внешнего взаимодействия.

4. EVM формирует блоки информации для встраивания в общий реестр.

5. Блок информации встраивается в блокчейн и связывается с другими блоками.

6. После завершения процесса интеграции отдельного блока в общую сеть транзакция подтверждается.

В случае реализации цифрового рубля с помощью этих технологий мы выделяем две глобальных категории проблем:

1. Проблемы безопасности смарт-контрактов и языка.
2. Проблемы реализуемой инфраструктуры.

Для понимания первой проблемы необходимо понять, на каких технологиях распределенных реестров реализуются смарт-контракты.

В соответствии с исследованием [4] результаты которого представлены на рис. 2, мы видим, что самыми популярными платформами являются BNB Chain, Ethereum и Polygon.

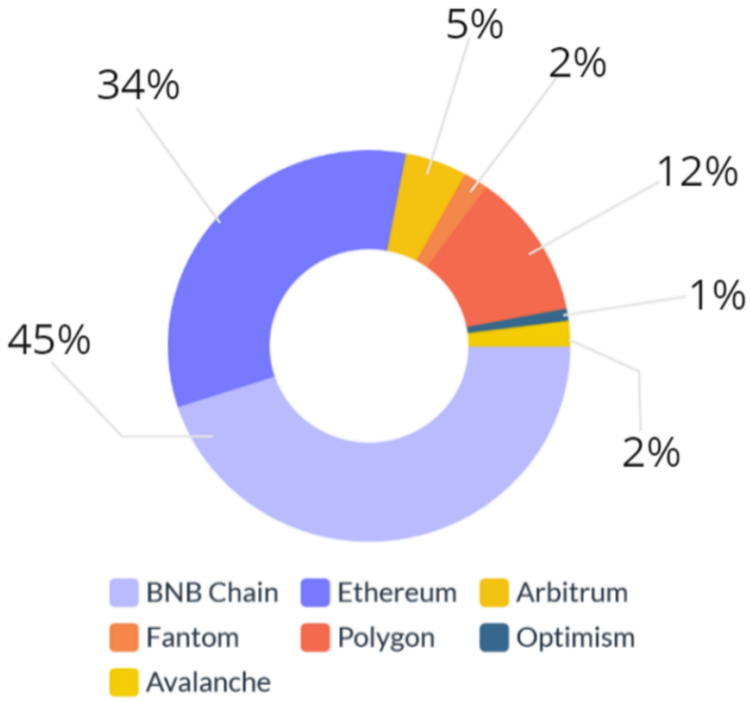


Рис. 2. Статистика популярных блокчейн-платформ

Все указанные выше блокчейн-платформы реализованы на сети Ethereum, либо полностью с ней совместимы, поэтому далее мы будем говорить только о смарт-контрактах реализованных на Ethereum.

В рамках данной проблемы, уязвимости, по области происхождения можно поделить на две категории:

1. Уязвимости логики смарт-контракта.
2. Уязвимости кода и языка реализации.

Уязвимости логики смарт-контракта – это проблемы, связанные с неправильной реализацией логики в коде смарт-контракта, которые могут приводить к ошибкам или нарушениям безопасности, описанным в Decentralized Application Security Project [5]:

1. Reentrancy – уязвимость, которая возникает в случае возможности совершения нового вызова функции смарт-контракта до завершения предыдущей, что может привести к финансовым потерям.

2. Access Control – проблема небезопасных настроек доступа, которые позволяют злоумышленнику получить прямой доступ к частным данным или логике контракта.

3. Insecure Arithmetic – неправильное или без должной проверки выполнение арифметической операции в смарт-контракте.

4. Denial Of Service – искусственное увеличение взимаемой с кошелька комиссии при проведении транзакции, приводящее к неработоспособности смарт-контракта.

5. Bad Randomness – уязвимость, возникающая при генерации случайного значения, или принятия решения на его основе, при котором создание этого значения не является истинно случайным и может быть предсказан злоумышленником.

6. Front-Running – ошибка, допускаемая в разработке смарт-контрактов, при которой злоумышленник использует информацию о транзакциях, чтобы опередить и получить выгоду.

7. Time Manipulation – атака подделки или изменения времени взаимодействия смарт-контракта с блокчейном для нанесения вреда участникам смарт-контракта.

Однако, помимо известных уязвимостей, описанных выше, проблемы с безопасностью могут вызвать и уязвимые конструкции языка, созданные самим разработчиком. К примеру, MultiSig – кошелек, который за счет некорректной логики не выполняет правильно условия обработки проверки подписей [6]. Его реализация представлена ниже, на рис. 3.

```
function changeRequirement(uint _required) public
  ownerExists(msg.sender)
  validRequirement(owners.length, _required)
{
  required = _required;
  RequirementChange(_required, now)
}
```

Рис. 3. Реализация некорректной логики MultiSig

Для подписи транзакции и перевода средств, необходимо количество подписей владельцев кошелька, равное переменной *required*, но для того, чтобы поменять эту переменную, к примеру на единицу, чтобы единолично подписывать транзакции, достаточно подписи только одного из владельцев. Неудачная логика программы, когда разработчики пытаются имплементировать в пределах одного контракта мультисигнатурность – ме-

ханизм, требующий несколько подписей для подтверждения определенных операций, таких как перевод средств или подписание транзакций, – является одной из причин почему возникают проблемы с безопасностью.

Далее рассматриваются проблемы, требующие внесения изменений в исполняемые файлы на языке программирования Solidity – доминирующем языке программирования в области разработки смарт-контрактов [7]. Одной из особенностей языка, являются низкоуровневые функции – методы, предоставляющие прямое взаимодействие с блокчейном или другими контрактами: *call()*, *callcode()*, *delegatecall()* и *send()*. Они эксплуатируют уязвимость Unchecked Return Values For Low Level Calls, возникающую при отсутствии проверки смарт-контрактом возвращаемого значения при взаимодействии с другими контрактами на низком уровне.

Для защиты от уязвимостей, описанных в классификации выше, можно применять следующие меры для обеспечения безопасности [8]:

1. Управление ресурсами смарт-контракта.
2. Проверка уязвимости функционала и бизнес-логики.
3. Контроль взаимодействия с внешним миром.
4. Обработка исключений - обработка ошибок, которые могут возникнуть в процессе работы смарт-контракта.
5. Проведение аудитов смарт-контрактов - детальный анализ исходного кода контракта для выявления в нем уязвимостей.

Проблемы безопасности реализации и внедрения смарт-контрактов могут лежать не только в области кода и логики смарт-контракта, но и в области инфраструктуры, которые можно разделить исходя из объекта атаки. Данная классификация представлена ниже.

1. Атаки на EVM. В сети Ethereum важнейшей частью инфраструктуры является EVM [9]. Она является связующим звеном между самим смарт-контрактом и блокчейном. У неё есть фундаментальный недостаток: нет никаких нативных библиотек и функций, к которым можно было бы обратиться. Следовательно, всю логику, которая используется в каждом смарт-контракте, приходится проектировать с нуля. При усложнении конструкции языка, логики и архитектуры увеличивается вероятность совершения ошибки при реализации EVM. Это усугубляется тем, что смарт-контракты взаимодействуют между собой, используя мультисигнатурность, которая описана выше. Примером эксплуатации такой уязвимости можно назвать атаку Furucombo – из-за некорректной инициализации определенных функций можно переопределить логику контракта. Далее, при вызове определенных функций, например *delegatecall()*, при обработке кода в EVM происходит сбой, позволяющий получить доступ к кошелькам контрагентов [10].

2. Атаки на механизмы консенсуса. Это нарушение логики принятия решений в сетях без единого центра управления. Поскольку сутью децентрализации является отсутствие главного звена, присутствуют существен-

ные трудности с принятием общих решений, без которых сеть не смогла бы функционировать как единое целое. Существуют четыре самых популярных механизма консенсуса для корректного функционирования системы:

A. Proof-of-Stake – валидаторы (пользователи, несущие ответственность за различные вычислительные операции) выбираются на основе их валютного вклада в общую сеть блокчейна.

B. Proof-of-Work – валидаторы сети выбираются на основе их вычислительной мощности: кто первый решает сложные математические вычисления, связанные с транзакцией, тот и несёт ответственность за нее.

C. Proof-of-Authority – валидаторы выбираются на основе их уровня репутации в сети, который присваивается системой на основе различных параметров: чем выше авторитет, тем выше шанс стать валидатором определенной транзакции.

D. Hydrid model – может совмещать в себе различные перечисленные выше механизмы консенсуса одновременно.

Из-за сложных моделей принятия решений существуют различные уязвимости, позволяющие нанести ущерб инфраструктуре. Одной из самых известных является «Атака 51%» – злоумышленники получают более 51% вычислительной мощности одной блокчейн платформы и за счет этого могут обходить механизмы консенсуса, манипулируя мощностями и в дальнейшем могут полностью скомпрометировать сеть.

3. Атаки на оракул. Оракулы в контексте децентрализованных финансов – приложения, которые получают, проверяют и передают внешнюю информацию (хранящуюся вне цепочки) в смарт-контракты, работающие в блокчейне. Как и любая другая технология, оракулы могут быть подвержены различным атакам, к примеру: атаки на конфиденциальность, атаки на доступность, атаки на целостность, манипуляции рынком.

4. Атака на децентрализованные приложения. Вид атаки на приложения, созданные и функционирующие на базе блокчейна, с использованием смарт-контрактов и пользовательского веб-интерфейса. Такие приложения уязвимы для тех же атак, что и смарт-контракты.

В связи с тем, что реализация цифрового рубля в Российской Федерации будет использовать как централизованные, так и децентрализованные технологии, то необходимо обратить внимание на возможные проблемы, связанные с использованием этих технологий. В процессе исследования авторы определили основные этапы функционирования и элементы смарт-контрактов, а также связи между ними. На основе этих этапов были выделены две основные категории проблем, которые могут возникнуть при использовании децентрализованных технологий. В результате была представлена классификация уязвимостей смарт-контрактов, языка их реализации и используемой инфраструктуры. В силу слабой изученности данного вопроса в дальнейшем необходимо более детально рассмотреть представ-

ленные выше уязвимости, их влияние на описанные технологии и возможные меры противодействия.

Библиографический список

1. Финансовая Налоговая Служба: Машиночитаемая доверенность, URL: https://m4d.nalog.gov.ru/ (дата обращения: 14.10.2023).

2. Эндрю Стенли, Восхождение ЦВЦБ, URL: https://www.imf.org/ru/Publications/fandd/issues/2022/09/Picture-this-The-ascent-of-CBDCs (дата обращения: 15.10.2023).

3. Kevin Warsh, The U.S. Needs a Better Digital Dollar, URL: https://www.wsj.com/articles/the-chinese-cryptocurrency-threat-e-cny-digital-dollar-yuan-reserve-privacy-wholesale-transfer-fed-reform-inflation-11668954794 (дата обращения: 16.10.2023).

4. BNB Chain: Web3 In Numbers: Verified Smart Contracts Surged in Q2 2023 Despite Bear Market, URL: https://www.bnbchain.org/en/blog/web3-in-numbers-verified-smart-contracts-surged-in-q2-2023-despite-bear-market (дата обращения: 17.10.2023).

5. NCC Group, Decentralized Application Security Project, URL: https://dasp.co/ (дата обращения: 18.10.2023).

6. Blockchain Media, Что такое мультисиг-кошелек (multisig) и как он работает?, URL: https://blockchain-media.org/what-is-multisig-wallet-and-how-it-works (дата обращения: 19.10.2023).

7. Solidity Documentation, Solidity Developer Survey 2022 Results, URL: https://soliditylang.org/blog/2023/03/10/solidity-developer-survey-2022-results/ (дата обращения 20.10.2023).

8. ConsenSys, Ethereum Smart Contract Best Practices, URL: https://consensys.github.io/smart-contract-best-practices/ (дата обращения: 21.10.2023).

9. Алиев И.А. Уязвимости смарт-контрактов блокчейн-платформы Ethereum / И.А. Алиев // Научные записки молодых исследователей. 2019. №3. С. 47–57.

10. Coinspect, Learn Evm Attacks, URL: https://github.com/coinspect/learn-evm-attacks/tree/master/test/Business_Logic/Furucombo (дата обращения: 23.10.2023).

УДК 004.056

АНАЛИЗ УЯЗВИМОСТЕЙ СИСТЕМ АУТЕНТИФИКАЦИИ И АВТОРИЗАЦИИ НА ОСНОВЕ JSON WEB TOKENS

К.А. Басалай, И.Р. Зулькарнеев
Научный руководитель: доц. кафедры информационной безопасности И.Р. Зулькарнеев
Тюменский государственный университет, г. Тюмень

В статье рассмотрена проблема обеспечения безопасности аутентификации и авторизации на основе JSON Web Tokens. Проведен анализ 119-ти уязвимостей базы данных CVE, связанных с JWT. В результате обработки полученных данных сделаны выводы о темпах ежегодного прироста уязвимостей JSON Web Tokens и тенденциях роста числа критических уязвимостей. Была составлена классификация уязвимостей JWT, включающая в себя 6 различных категорий, обнаружено появление новых типов уязвимостей за последние 3 года. Выявлен феномен комбинированных уязвимостей и проведен их анализ. По результатам исследования сделан вывод о необходимости пересмотра существующих классификаций уязвимостей JWT и подходов к обеспечению безопасности их применения.

Ключевые слова: JSON Web Tokens, JWT, аутентификация, авторизация, уязвимость, CVE, CVSS.

Число кибератак на веб-приложения увеличивается с каждым годом. Согласно исследованию компании Positive Technologies [1], веб-ресурсы становились объектами атак злоумышленников в 22% случаев, а число успешных атак на веб-приложения увеличилось на 56%. В то же время инциденты, связанные с атаками на веб-ресурсы в 53% случаев, приводили к нарушению деятельности организации.

По данным отчета за 2020-2021 гг. [2] наиболее распространенными являются уязвимости, связанные с нарушением контроля доступа, они были обнаружены во всех исследованных приложениях (рис. 1).

В данном случае под контролем доступа следует понимать политику, при которой пользователи действуют исключительно в пределах своих полномочий [3]. Эксплуатация уязвимостей данного типа может повлечь за собой компрометацию пользовательских конфиденциальных данных, а также несанкционированный доступ к сторонним личным кабинетам и ресурсам компании. Одними из самых опасных уязвимостей данной категории являются некорректные авторизация и аутентификация пользователей. Следовательно, при проектировании информационной системы особое

внимание требуется уделять именно системам аутентификации и авторизации.

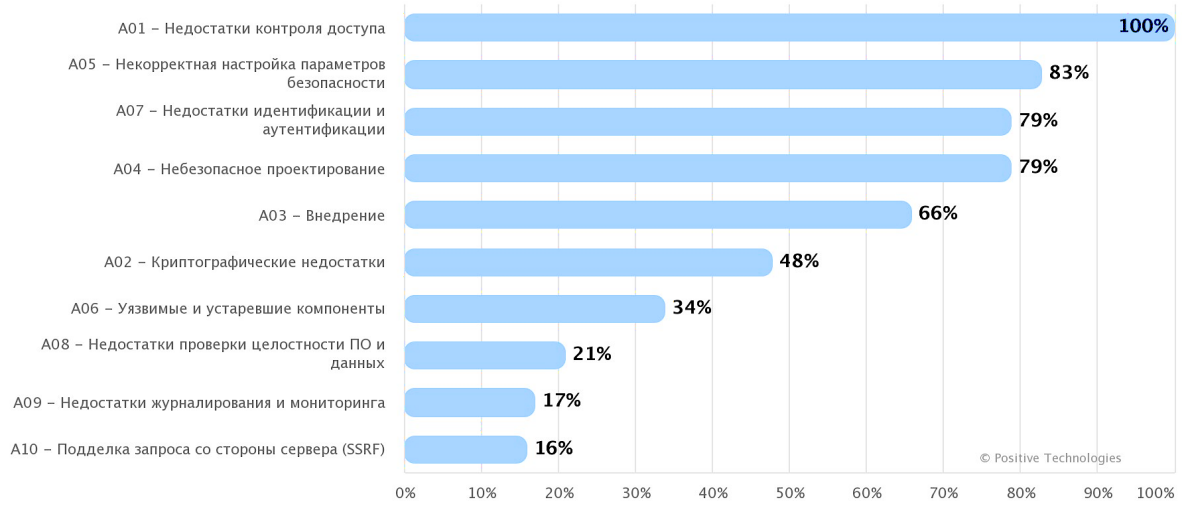


Рис. 1. Распределение уязвимостей по категориям OWASP Top 10

Крупные организации и учреждения зачастую создают собственные информационные экосистемы. Такой подход очень удобен для пользователя, потому что можно один раз ввести свои учетные данные и пользоваться всеми сервисами в рамках экосистемы компании. Для реализации такого подхода нужна особенная система аутентификации и авторизации.

Одним из распространенных решений данной задачи является использование JSON Web Tokens (далее JWT) – средства аутентификации, авторизации и передачи информации между двумя сторонами в формате JSON. Среди преимуществ использования JWT можно выделить:

1. Удобство – благодаря аутентификации на основе токенов, клиенту нет необходимости вводить учетные данные для пользования различными сервисами в рамках одной экосистемы.

2. Универсальность – JSON Web Tokens можно использовать в приложениях, написанных на разных языках программирования на различных платформах.

3. Повышение производительности веб-приложения – JWT хранятся на стороне клиента, а не на сервере, поэтому время ответа на пользовательские запросы сокращается [4].

Несмотря на перечисленные преимущества, JWT имеют потенциальные уязвимости в своей структуре. Проблемы с безопасностью связаны с тем, что технология JSON Web Tokens появилась не так давно и нет единого стандарта их создания, существуют лишь рекомендации, одни разработчики их придерживаются, а другие – нет. Как следствие – появление большого числа уязвимостей при использовании JWT.

В ходе работы был проведен анализ базы данных уязвимостей CVE, которые напрямую связаны с процессами аутентификации и авторизации на

основе JSON Web Tokens, было выявлено 119 уязвимостей различного уровня критичности.

На рис. 2 представлена динамика ежегодного роста уязвимостей, связанных с JWT в период с 2015 по 2023 гг.

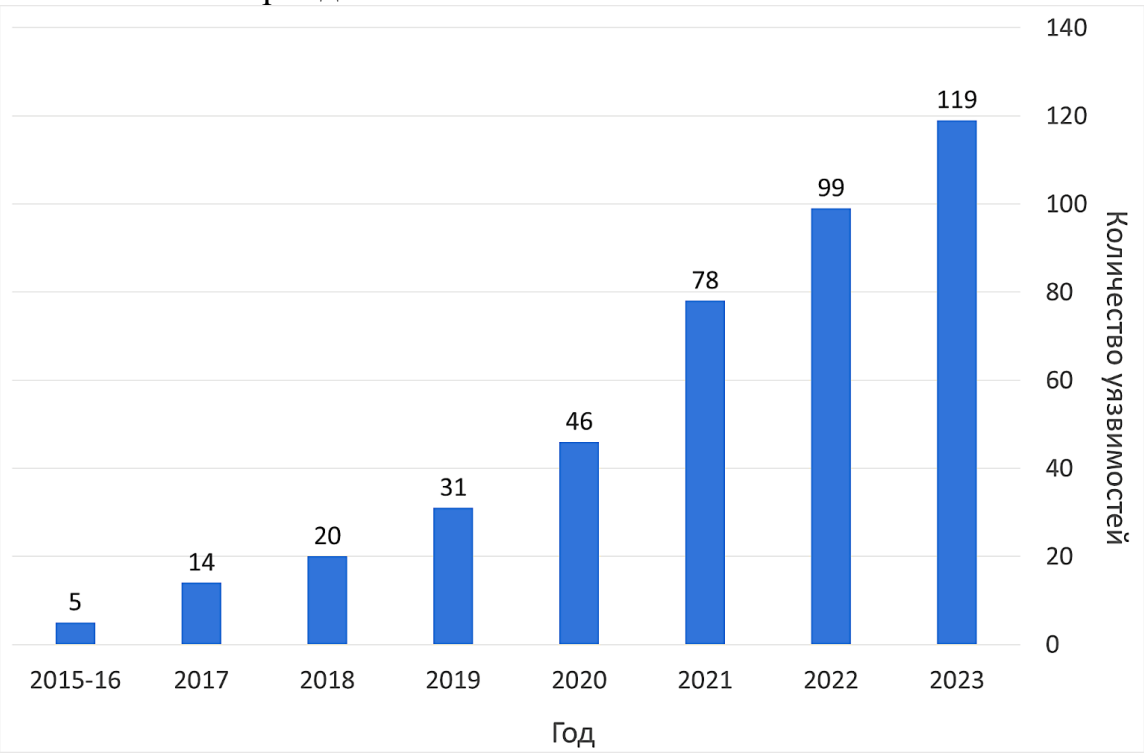


Рис. 2. Диаграмма роста уязвимостей JWT

Ежегодный прирост уязвимостей в период с 2015 по 2020 годы составляет в среднем 9,2 уязвимостей в год. Как можно заметить, начиная с 2021 года темпы появления новых уязвимостей увеличились и составляют в среднем 24,3 уязвимости в год. Данный факт может означать, что растет количество приложений, использующих аутентификацию и авторизацию на основе JWT. Также полученные данные могут говорить о росте количества уязвимостей в библиотеках создания и обработки JSON Web Tokens. Из приведенного графика видны стремительные темпы роста числа уязвимостей, что свидетельствует о массовости применения JWT и необходимости обеспечения мер по безопасности их использования.

Также был проведен анализ уязвимостей в соответствии со стандартом расчета количественных оценок уязвимостей CVSS v3.0. Полученные результаты представлены на рис. 3.

Из полученных данных можно сделать несколько выводов. Во-первых, за период с 2015 по 2023 гг. обнаружена только одна уязвимость уровня Low по шкале CVSS v3.0 с рейтингом 3.3. Во-вторых, можно отметить, что после 2021 года имеется, с одной стороны, тенденция к снижению прироста уязвимостей уровня Medium и High с рейтингами 4.0-6.9 и 7.0-8.9 соответственно по шкале CVSS v3.0, а с другой - ежегодное увеличение числа

критических уязвимостей с рейтингом 9.0-10.0. Оценивая темпы роста и тенденции появления критических уязвимостей, можно вывод о том, что уровень их опасности будет продолжать расти, а эксплуатация данных уязвимостей может привести к серьезному ущербу информационной системе.

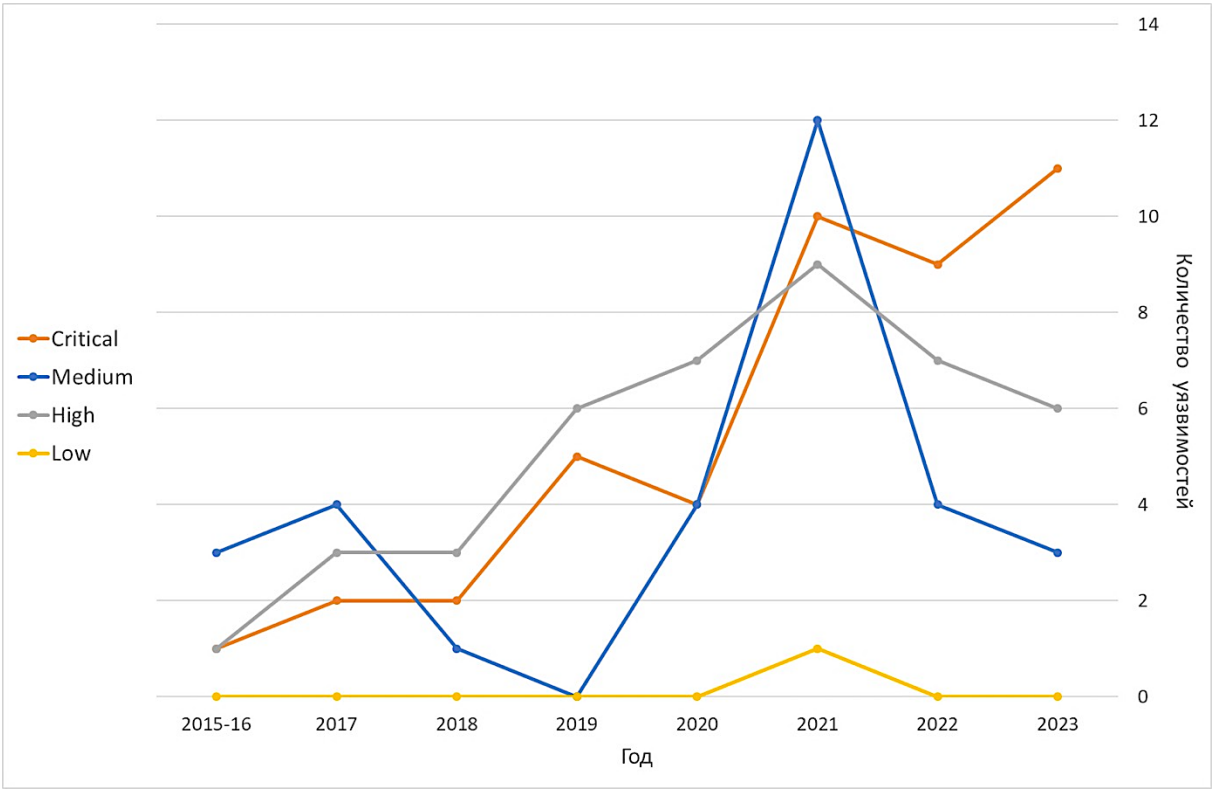


Рис. 3. Анализ уязвимостей в соответствии с их уровнем CVSS v3.0

В ходе исследования была предпринята попытка классификации найденных уязвимостей, в результате чего было определено 6 категорий.

1. Уязвимости в структуре токена включают в себя различные инъекции в поля токенов, а также изменение существующих полей полезной нагрузки.

2. Атаки на криптографию включают уязвимости, связанные с алгоритмами формирования подписи JWT, возможностями использования непредусмотренных криптографических алгоритмов.

3. Уязвимость hard-coded JWT keys возникает в том случае, если разработчики веб-приложения оставили в его исходном коде секретный ключ для формирования подписи JSON Web Tokens.

4. Уязвимости в конфигурации сервера включают в себя возможность вызвать отказ в обслуживании (DoS) вследствие неправильной обработки токенов с некорректной подписью или случаи, когда сервер в ответ на неверно подписанный токен в http-ответе возвращал значение верного ключа подписи.

5. Уязвимости передачи токена связаны с возможностью их перехвата вследствие атаки типа «человек посередине» (MITM) или же при реализации атак на клиента типа XSS и CSRF.

6. К уязвимостям окружения токена были отнесены некорректная обработка JWT в рамках микросервисной архитектуры, уязвимости в API-запросах и возможность скомпрометировать секретный ключ на стороне сервера

На рис. 4 приведен график распределения уязвимостей различных категорий. Сплошная заливка соответствует количеству уязвимостей в каждой категории, пунктир – показывает долю критических уязвимостей от числа всех уязвимостей данной категории.

Как видно из графика, с 2020 года начинают появляться принципиально новые уязвимости, а именно hard-coded JWT keys и уязвимости передачи токена. Рост первых особенно заметен в 2022–2023 годах, их количество увеличилось в 6 раз по сравнению с 2021 и 2022 годами вместе взятыми, а доля критических уязвимостей среди них составляет 83%. Уязвимости в окружении токена впервые появляются в 2021 году. Из полученной классификации видно, что со временем появляются всё новые категории уязвимостей, связанные с JSON Web Tokens. Это говорит о том, что веб-приложения в целом становятся более сложными, а уязвимости системы аутентификации могут располагаться на различных уровнях: клиентском, серверном или локально, внутри токена. Полученные данные подтверждают необходимость комплексного подхода при проектировании системы аутентификации и анализе ее защищенности.

В ходе исследования было замечено, что некоторые системы аутентификации и авторизации на основе JWT содержат в себе уязвимости, которые можно отнести сразу к нескольким категориям представленной выше классификации, в рамках данной статьи будем называть такие уязвимости комбинированными. Данный тип уязвимостей стал появляться с 2020 года, половина из них имеет уровень опасности Critical по шкале CVSS v3.0 со средним рейтингом 8.5.

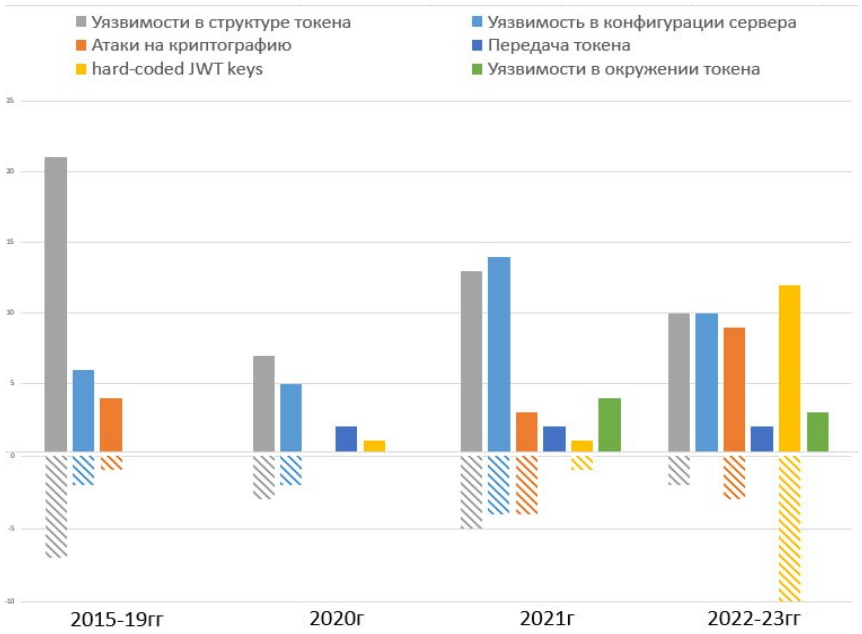


Рис. 4. Соотношение количества критических уязвимостей JWT с общим числом уязвимостей каждой категории

Также был проведен анализ зависимости между критическими и комбинированными уязвимостями, результаты которого представлены на рис. 5.

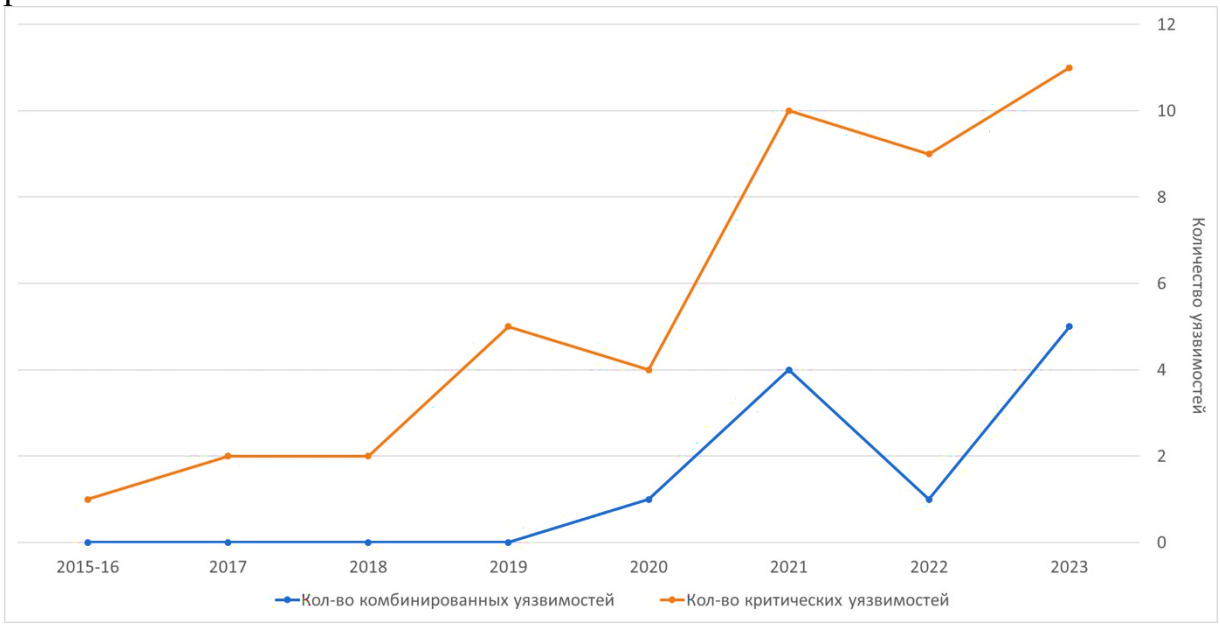


Рис. 5. Анализ зависимости критических и комбинированных уязвимостей

Как видно из графика, между данными уязвимостями существует прямая зависимость: с увеличением числа комбинированных уязвимостей, растет и число критических. Ввиду того, что комбинированные уязвимости относятся сразу к нескольким категориям представленной классификации, то возрастает количество способов их эксплуатации. Это повышает риск нанесения серьезного ущерба информационной системе, поэтому комбинированные уязвимости имеют высокий рейтинг по CVSS v3.0.

В результате исследования был проведен анализ 119 существующих на данный момент уязвимостей базы CVE, связанных с процессами аутентификации и авторизации на основе JWT. Была выявлена положительная динамика ежегодного прироста уязвимостей, что свидетельствует об актуальности использования JSON Web Tokens в современных веб-приложениях. Был проведен анализ найденных уязвимостей в соответствии с системой количественных оценок CVSS v3.0, на основании которого выявлена тенденция роста критических уязвимостей с одновременным снижением числа уязвимостей типа High и Medium. Была составлена классификация полученных уязвимостей, включающая в себя 6 различных категорий. Из классификации видно, что за последние несколько лет в JWT появилось несколько новых типов уязвимостей, часть из которых имеют высокую степень критичности. Также был обнаружен феномен комбинированных уязвимостей в JWT и выявлена зависимость между ними и критическими уязвимостями.

Высокие темпы роста уязвимостей, появление уязвимостей новых типов и уровень их критичности говорят о необходимости пересмотра существующих классификаций и подходов к обеспечению безопасности систем аутентификации и авторизации на основе JWT с учетом новых тенденций.

Библиографический список

1. Актуальные киберугрозы: итоги 2022 года, // URL: https://www.ptsecurity.com/ru-ru/research/analytics/cybersecurity-threatscape-2022/ (дата обращения 11.10.2023).

2. Уязвимости и угрозы веб-приложений в 2020–2021 гг., // URL: https://www.ptsecurity.com/ru-ru/research/analytics/web-vulnerabilities-2020-2021/ (дата обращения 11.10.2023).

3. OWASP Top Ten, // URL: https://owasp.org/www-project-top-ten/ (дата обращения 11.10.2023).

4. Феоктистов И.В. Сравнительное исследование методов аутентификации в информационных системах / И.В. Феоктистов // Инновации и инвестиции. – 2023. № 7. – С. 193–198.

УДК 004.056

ОБЕСПЕЧЕНИЕ БЕЗОПАСНОСТИ СЕТИ АСУ ТП ПРИ ИСПОЛЬЗОВАНИИ КОМБИНИРОВАННОГО МЕТОДА АКТИВНОГО СКАНИРОВАНИЯ[1]

А.В. Быкасов, А.М. Богер, А.Н. Соколов
Научный руководитель: канд. техн. наук, доц. А.Н. Соколов
Южно-Уральский государственный университет, г. Челябинск

Современные системы, обеспечивающие безопасность сетей автоматизированных систем управления технологических процессов (АСУ ТП), например системы обнаружения вторжений, все чаще комплектуются модулем, обеспечивающим периодический опрос имеющихся узлов сети – активным сканером. Несмотря на то, что процесс активного сканирования является легитимным, неконтролируемый поток данных запросов и ответов от узлов может оказаться небезопасным для рабочего процесса АСУ ТП, поскольку узлы сети затрачивают свои ресурсы на оформление ответа. Эта задержка может оказаться критичной при обеспечении непрерывности производственного процесса в реальном времени работы АСУ ТП. Поэтому процесс активного сканирования не должен создавать нагрузку на узлы сети АСУ ТП, приводящую к замедлению или недоступности передаваемых сигналов. В этой связи актуальной является разработка безопасных методов активного сканирования сетей. В работе представлен комбинированный метод активного сканирования, не нарушающий безопасность функционирования сети АСУ ТП, и описаны параметры его настройки.

Ключевые слова: автоматизированная система управления технологическим процессом (АСУ ТП), активное сканирование, контроль сетевых устройств, программируемый логический контроллер (ПЛК), сетевое взаимодействие.

В настоящее время сети автоматизированных систем управления технологическими процессами (АСУ ТП) зачастую имеют значительную протяженность и обеспечение их информационной безопасности становится задачей дополнительного программного обеспечения, такого как системы обнаружения или предотвращения вторжений (СОВ и СПВ). Их функционал направлен на контроль целостности трафика, сети и исполняемого на узлах ПО.

[1] Исследование поддержано грантом Российского научного фонда (проект № 22-71-10095).

При этом описанный функционал может быть полностью реализован с помощью методов пассивного анализа сети (сниффинг, пассивный анализ трафика). Но такой подход является излишне затратным по времени и в некоторых ситуациях может иметь недостаточную точность. Поэтому СОВ зачастую дополняются методами активного сканирования [1]. Их суть заключается в формировании СОВ специально сформированного запроса, посылаемого целевому хосту, всем хостам, периферийному оборудований, либо всем типам устройств и дальнейшая интерпретация ответов от них [2].

Большую оперативность в получении информации об используемых устройствах, установленном ПО и конфигурациях в сети АСУ ТП даёт способ, заключающийся в интеграции СОВ с установленным на рабочие станции программном обеспечением. Такая интеграция может проводиться посредством установленного на рабочие станции специализированного ПО – агента инвентаризации, который передаёт информацию, либо напрямую в СОВ, либо через SIEM систему. Однако, данный способ может быть применим только к устройствам, на которые возможно свободно установить прикладное ПО огромное количество встраиваемых систем в АСУ ТП (программируемые логические контроллеры (ПЛК), распределённый ввод-вывод, сетевое оборудование и т.д.) как правило не имеют возможности установки дополнительного ПО [3–5].

Таким образом, необходимо применять активное сканирование, запущенное средствами самой СОВ, либо сторонними ресурсами. Однако, такие подходы нарушают принцип невмешательства в рабочий процесс, то есть принцип невмешательства СОВ в сетевой процесс для получения информации. Следовательно, для обеспечения информационной безопасности сетей АСУ ТП и одновременной минимизации вмешательства в их рабочий процесс необходимы специальные методы активного сканирования [6, 7]. Однако стоит понимать, что применение методов активного сканирования может нести риски для сети АСУ ТП, поскольку не существует индустриально принятых процедур и методов, которые могут формально подтвердить, что использование активного сканирования не приведёт к нарушению функционирования АСУ ТП.

Цель данной статьи состоит в создании комбинированного метода активного сканирования, который будет обеспечивать безопасность сети АСУ ТП при его использовании. Под безопасностью в данном случае подразумевается обеспечение целостности и доступности сети и подключенных устройств.

Комбинированный метод активного сканирования состоит из 4-х этапов сканирования:

1. ARP-сканирование. На этом этапе отправляется широковещательный запрос в сеть с целью обнаружения подключенных хостов, а также нахождения их IP- и MAC-адресов. В качестве входных данных используется под-

сеть или IP-адрес устройства. В результате получается список устройств, содержащий IP- и MAC-адреса.

2. ICMP-сканирование. В качестве входных данных используется список IP-адресов, полученный на предыдущем этапе. Далее, на каждое устройство из списка отправляется ping-запрос. Если ответ приходит, мы считаем это устройство «живым». В результате получаем список «живых» IP-адресов.

3. SNMP-сканирование. На данном этапе проводится опрос устройств по протоколу SNMP с целью получения информации об устройстве: марка, модель, имя в сети и т.д. В качестве входных данных используется результат ICMP-сканирования.

4. Сканирование портов. На данном этапе проводится поиск открытых портов устройств посредством отправки TCP-пакетов. Сканирование может проводиться как по определенному заранее списку «популярных» портов, так и по заданному вручную. В качестве входных данных также используется список IP-адресов, полученный на этапе ICMP-сканирования.

Для апробирования нашего комбинированного метода активного сканирования в работе был использован лабораторный стенд, осуществляющий сетевое взаимодействие двух ПЛК. Состав оборудования:

– ПЛК-1, Siemens-1512;
– ПЛК-2, Siemens-1510;
– Коммутатор Scalance XC208;
– АРМ инженера, содержащая ПО для программирования ПЛК;
– АРМ оператора системы безопасности;
– HMI- панель для визуализации и управления.

На ПЛК загружена программа, эмулирующая металлонарезной станок. Для проверки методов сканирования к стенду подключен ноутбук, исполняющий роль внешнего сканирующего устройства.

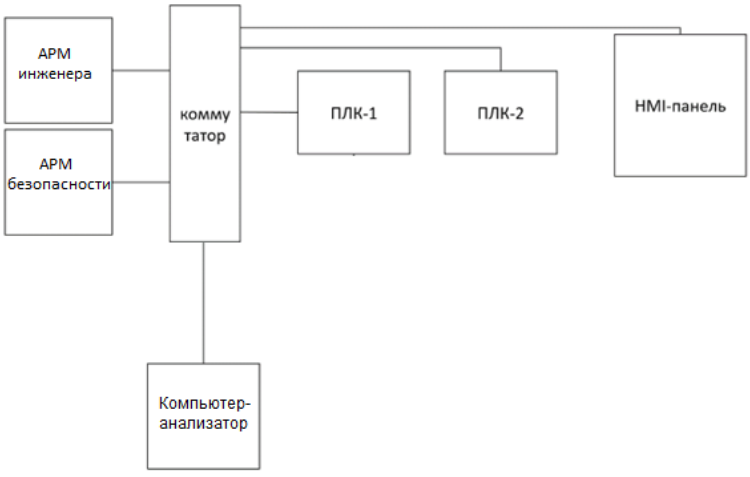


Рис. 1. Схема сетевых соединений первого сегмента лабораторного стенда

В результате экспериментов были получены следующие результаты:

1. ARP-сканирование:

Были обнаружены все устройства сети, их IP- и MAC-адреса, включая точку сканирования (рис. 2). Среднее время отклика составило 491 мс. Согласно анализатору траффика Wireshark при сканировании создается всплеск сетевой активности, хорошо видный на графике (рис. 3).

2. ICMP-сканирование:

Все устройства, обнаруженные на предыдущем этапе, ответили на запрос и считаются «живыми» (рис. 2). Среднее время отклика составило 75 мс. Согласно анализатору траффика Wireshark при сканировании создается всплеск сетевой активности, который сливается с основным трафиком сети (рис. 3).

3. SNMP-сканирование:

Из всех устройств сети ответило только одно, так как на остальных устройствах протокол SNMP не поддерживается, либо был отключен (рис. 2). Среднее время отклика составило 396 мс. Согласно анализатору траффика Wireshark при сканировании создается всплеск сетевой активности, который сливается с основным трафиком сети (рис. 3).

4. Сканирование портов:

Для найденных устройств были обнаружены все открытые порты (рис. 2). Среднее время отклика составило 456 мс. Согласно анализатору траффика Wireshark при сканировании создается всплеск сетевой активности, который сливается с основным трафиком сети (рис. 3).

```
IP                    MAC Address
-----------------------------------------------------------
192.168.150.103         18:31:bf:af:65:ed
192.168.150.101         ac:64:17:2d:10:c0
192.168.150.102         ac:64:17:2b:5b:32
192.168.150.110         00:d0:c9:fa:eb:b8
192.168.150.100         20:87:56:9d:62:5c
192.168.150.111         d8:c0:a6:81:64:99
Number of devices by ARP: 6
Avarage responce time for ARP: 0.49147140979766846 s

Live hosts discovered using ICMP (Ping): {'192.168.150.103', '192.168.150.101', '192.168.150.102', '192.168.150.110', '192.168.150.100', '192.168.150.111'}
Number of devices by ICMP: 6
Avarage responce time for ICMP: 0.0755537748336792 s

Error for 192.168.150.103
Error for 192.168.150.101
Error for 192.168.150.102
Error for 192.168.150.110
SNMP Info for 192.168.150.100:
1.3.6.1.2.1.1.1.0 : Siemens, SIMATIC NET, SCALANCE XC208, 6GK5 208-0BA00-2AC2, HW: Version 1, FW: Version V03.00.02, SVPK8155812
1.3.6.1.2.1.1.5.0 : sysName Not Set
Error for 192.168.150.111
Avarage responce time for SNMP: 0.39629435539245605 s

Open ports for host 192.168.150.103: [445, 3389, 8080]
Open ports for host 192.168.150.101: [80, 503, 443]
Open ports for host 192.168.150.102: [80, 443, 5900]
Open ports for host 192.168.150.110: [21, 22, 23, 80, 443]
Open ports for host 192.168.150.100: [22, 23, 80, 443]
Open ports for host 192.168.150.111: [445]
Avarage responce time for Scanning Ports: 0.4557889382044474 s

Scan Time:  165.19584465026855 s
```

Рис. 2. Ответ от устройств лабораторного стенда при активном сканировании

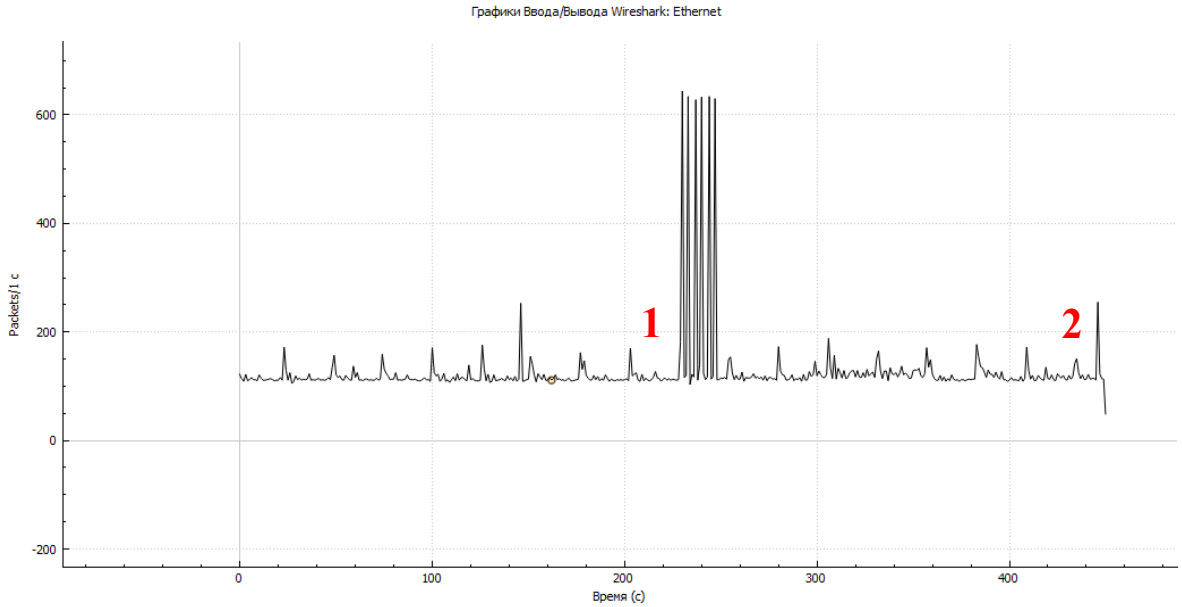


Рис. 3. Сетевой трафик лабораторного стенда во время сканирования:
1 – начало сканирования, 2 – конец сканирования

Из полученных результатов хорошо видно, что трафик, генерируемый при использовании данного метода хорошо заметен на фоне основного трафика сети. Следовательно, его применение может оказывать негативное влияние на пропускную способность сети, ее целостность, а также на доступность устройств сети. Для того, чтобы обеспечить безопасное сканирование, необходимо предусмотреть возможность настройки параметров сканирования:

1. При ARP-сканировании необходимо выбрать подходящее значение timeout для получения более точного результата. Также нужно подобрать значение inter, которое определяет интервал отправки запросов в сеть. С помощью этого параметра можно контролировать скорость отправки пакетов и гарантировать, что сеть будет функционировать в штатном режиме.

2. При SNMP-сканировании необходимо правильно определить параметры OID для получения корректных ответов от устройств.

3. При сканировании портов необходимо определить параметр timeout для получения более точного результата при наименьшем времени сканирования.

После применения корректных параметров были проведены те же эксперименты и получены следующие результаты:

1. ARP-сканирование:

Были обнаружены все устройства сети, их IP- и MAC-адреса, включая точку сканирования (рис. 4). Среднее время отклика составило 3,75 с. Согласно анализатору траффика Wireshark при сканировании создается всплеск сетевой активности, хорошо видный на графике (рис. 5).

2. ICMP-сканирование:

Все устройства, обнаруженные на предыдущем этапе, ответили на запрос и считаются «живыми» (рис. 4). Среднее время отклика составило 59 мс. Согласно анализатору траффика Wireshark при сканировании создается всплеск сетевой активности, который сливается с основным трафиком сети (рис. 4).

3. SNMP-сканирование:

Из всех устройств сети ответило только одно, так как на остальных устройствах протокол SNMP не поддерживается, либо был отключен (рис. 4). Среднее время отклика составило 678 мс. Согласно анализатору траффика Wireshark при сканировании создается всплеск сетевой активности, который сливается с основным трафиком сети (рис. 5).

4. Сканирование портов:

Для найденных устройств были обнаружены все открытые порты (рис.4). Среднее время отклика составило 320 мс. Согласно анализатору траффика Wireshark при сканировании создается всплеск сетевой активности, который сливается с основным трафиком сети (рис. 5).

```
You didn't provide any target. Default target is 192.168.150.1/24
You didn't provide any method of scanning. Default method is auto
IP                    MAC Address
-----------------------------------------------------------
192.168.150.103       18:31:bf:af:65:ed
192.168.150.111       d8:c0:a6:81:64:99
192.168.150.110       00:d0:c9:fa:eb:b8
192.168.150.100       20:87:56:9d:62:5c
192.168.150.102       ac:64:17:2b:5b:32
192.168.150.101       ac:64:17:2d:10:c0
Number of devices by ARP: 6
Avarage responce time for ARP: 3.752784252166748 s

Live hosts discovered using ICMP (Ping): {'192.168.150.103', '192.168.150.111', '192.168.150.110', '192.168.150.100', '192.168.150.101', '192.168.150.102'}
Number of devices by ICMP: 6
Avarage responce time for ICMP: 0.05848113695780436 s

Error for 192.168.150.103
Error for 192.168.150.111
Error for 192.168.150.110
SNMP Info for 192.168.150.100:
1.3.6.1.2.1.1.1.0 : Siemens, SIMATIC NET, SCALANCE XC208, 6GK5 208-0BA00-2AC2, HW: Version 1, FW: Version V03.00.02, SVPK8155812
1.3.6.1.2.1.1.5.0 : sysName Not Set
Error for 192.168.150.101
Error for 192.168.150.102
Avarage responce time for SNMP: 0.6776385307312012 s

Open ports for host 192.168.150.103: [445, 3389, 8080]
Open ports for host 192.168.150.111: [445]
Open ports for host 192.168.150.110: [21, 22, 23, 80, 443]
Open ports for host 192.168.150.100: [22, 23, 80, 443]
Open ports for host 192.168.150.101: [80, 503, 443]
Open ports for host 192.168.150.102: [80, 443, 5900]
Avarage responce time for Scanning Ports: 0.3200418478912777 s

Scan Time:  166.92304706573486 s
```

Рис. 4. Ответ от устройств лабораторного стенда при активном сканировании после применения настроек: 1 – начало сканирования, 2 – конец сканирования

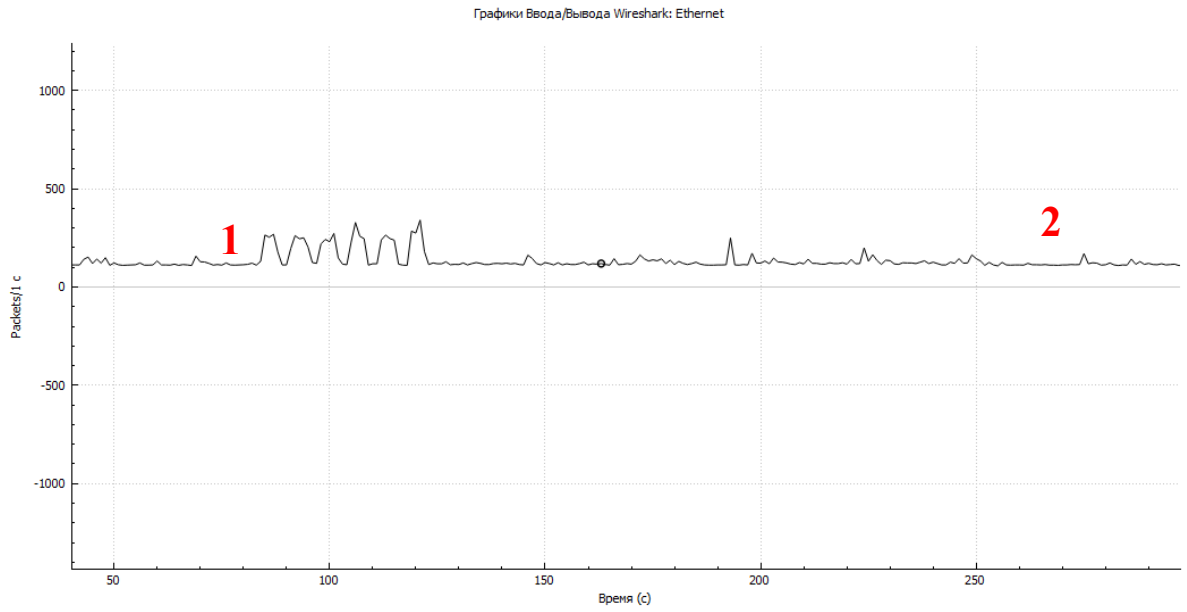


Рис. 5. Сетевой трафик лабораторного стенда во время сканирования после применения настроек

В результате работы был спроектирован комбинированный метод активного сканирования сетей АСУ ТП, который обеспечивает безопасное взаимодействие с сетью и ее устройствами при сканировании. Метод предусматривает применение настроек, которые позволяют получить более точные результаты при наименьшем воздействии на сеть.

После применения настроек мы получили более сглаженный по времени трафик, который генерируется при использовании комбинированного метода активного сканирования. Общее время сканирования увеличилось на 1 секунду. Результаты сканирования не ухудшились. Для определения большего количества параметров комбинированного метода активного сканирования необходимы дополнительные эксперименты, в том числе с другим оборудованием.

Библиографический список

1. Павленко А. Сканирование на наличие уязвимостей. / А. Павленко // Отус онлайн-образование. – 2022. – URL: https://otus.ru/nest/post/2468/ (дата обращения: 23.10.2023).

2. Проведение активных опросов устройств с помощью Kaspersky Industrial CyberSecurity for Networks. // URL: https://support.kaspersky.com/KICSforNetworks/4.0/ru-RU/236044.htm (дата обращения: 21.10.2023).

3. Активное сканирование с помощью Nozomi Networks Guardian. // URL: https://www.nozominetworks.com/downloads/US/Nozomi-Networks-Smart-Polling-Data-Sheet.pdf (дата обращения: 21.10.2023).

4. Hansson A. Analyzing Internet-connected industrial equipment. / A. Hansson, M. Khodari, A. Gurtov. // 2018 International Conference on Signals and Systems (ICSigSys). 2018. С. 29–35. – DOI: 10.1109/ICSIGSYS.2018.8372775 – URL:

https://www.researchgate.net/publication/325635836_Analyzing_Internet-connected_industrial_equipment (дата обращения: 23.10.2023).

5. Исследование: более 4 000 устройств АСУ ТП уязвимы для удаленных атак / InfoWatch. – 2021. – URL: https://www.infowatch.ru/resources/blog/issledovanie-bolee-4-000-ustroystv-asu-tp-uyazvimy-dlya-udalennykh-atak (дата обращения: 24.10.2023).

6. Функциональная безопасность систем электрических, электронных, программируемых электронных, связанных с безопасностью. Часть 1. Общие требования [Текст]: ГОСТ Р МЭК 61508-1-2012. – Введ.2013-08-01. – М.: Федеральное агентство по техническому регулированию и метрологии, 2012. – 586 с.

7. Функциональная безопасность систем электрических, электронных, программируемых электронных, связанных с безопасностью. Часть 3. Требования к программному обеспечению [Текст]: ГОСТ Р МЭК 61508-3-2012. – Введ. 2013-08-01. – М.: Федеральное агентство по техническому регулированию и метрологии, 2012. – 588 с.

УДК 004.056.53

ОСОБЕННОСТИ ВНЕДРЕНИЯ ЗАЩИЩЕННОГО КЛАСТЕРА KUBERNETES С УЧЕТОМ ТРЕБОВАНИЙ ДОКУМЕНТОВ ФСТЭК РОССИИ НА ОСНОВЕ ОТЕЧЕСТВЕННЫХ ОПЕРАЦИОННЫХ СИСТЕМ СЕМЕЙСТВА «АЛЬТ»

Б.В. Ружанович, С.А. Сабельников
Научный руководитель: ст. преподаватель С.А. Сабельников
Южно-Уральский государственный университет,
г. Челябинск

В статье сформулированы особенности процесса внедрения защищенного кластера «Kubernetes», и привидение его в соответствие требования документов ФСТЭК на базе отечественных операционных систем семейства «Альт».

Ключевые слова: защищенный кластер «Kubernetes», отечественные операционные системы.

В современном мире информационных технологий и цифровой трансформации, все больше российских организаций стремятся использовать технологии с открытым исходным кодом, такие как «Kubernetes», для управления своими контйнеризованными приложениями. «Kubernetes» – это открытое программное обеспечение *для управления контейнеризованными приложениями* [1]. Оно предоставляет автоматизацию развертывания, масштабирования и управления контейнерами в кластерах серверов. Kubernetes позволяет разработчикам и операторам проще и быстрее развертывать, обновлять и масштабировать приложения в облаке или локально. Контейнер-

ные приложения – это приложения, которые упакованы в небольшие, легкие и переносимые контейнеры. Эти контейнеры содержат все зависимости и библиотеки, необходимые для запуска приложения, что упрощает развертывание и обновление программного обеспечения [2]. Благодаря Kubernetes, эти приложения можно легко масштабировать и управлять ими в облачных средах, а также быстро вносить изменения и обновлять их.

При использовании «Kubernetes» возникает вопрос соответствия требованиям ФСТЭК России. Основным требованием, утвержденным ФСТЭК России, которое необходимо соблюсти является «Требование по безопасности информации к средствам контейнеризации» № 118, утвержденный приказом ФСТЭК России от 4 июля 2022 года [3].

Цель данной статьи – раскрыть особенности процесса внедрения защищенного кластера «Kubernetes», и привидение его в соответствие требования документов ФСТЭК. В качестве основы для создания кластера использованы отечественные операционные системы семейства «Альт», разработанные российской компанией «Базальт СПО».

Основным преимуществом ОС «Альт» является то, что это единственная ОС, основанная на отечественном репозитории СИЗИФ (один из крупнейших в мире репозиториев СПО, наряду с Debian, Red Hat и SUSE. Он основан в России и развивается международным сообществом разработчиков ALT Linux Team), что делает из нее лучшим импортозамещением на рынке ОС в России.

Для раскрытия особенностей процесса внедрения «Kubernetes» рассмотрим принципы его работы, а также возможности операционных систем «Альт» и требования ФСТЭК. Основными принципами работы «Kubernetes» являются:

- Масштабируемость: «Kubernetes» позволяет легко масштабировать приложение вверх или вниз, в зависимости от требований к нагрузке;
- Автоматическое развертывание: «Kubernetes» автоматически развертывает, обновляет и удаляет контейнеры на основе заданных конфигураций;
- Самовосстановление: если какой-либо из контейнеров перестает работать, «Kubernetes» автоматически перезапускает его;
- Разделение ответственности: «Kubernetes» разделяет развертывание, масштабирование и управление контейнерами, что упрощает процесс разработки и эксплуатации приложений;
- Гибкость: «Kubernetes» поддерживает различные среды, такие как публичные и частные облака, а также физические серверы;
- Мониторинг и оповещения: «Kubernetes» предоставляет инструменты для мониторинга состояния кластера и приложений, а также отправки оповещений при возникновении проблем;

- Безопасность: «Kubernetes» обеспечивает защиту от уязвимостей и атак, таких как изоляция контейнеров и ограничение доступа к ресурсам [4].
- «Kubernetes» не привязан к аппаратной инфраструктуре и представляет весь центр хранения и обработки данных как единый вычислительный ресурс. [5]

Операционная система «Альт» имеет множество возможностей, которые стали решающими в выборе ОС для разворачивания кластера «Kubernetes». Рассмотрим, основные возможности ОС «Альт»:

- Безопасность: ОС «Альт» обеспечивает высокий уровень безопасности, защищая систему от вредоносных программ, вирусов и других угроз;
- Простота использования: ОС «Альт» имеет простой и интуитивно понятный интерфейс, который облегчает работу с системой;
- Гибкость: ОС «Альт» поддерживает различные архитектуры и конфигурации, что позволяет использовать ее на разных типах устройств;
- Надежность: ОС «Альт» разработана с учетом надежности и стабильности, что гарантирует бесперебойную работу системы;
- Производительность: ОС «Альт» оптимизирована для обеспечения высокой производительности, позволяя пользователям работать с большим количеством задач одновременно;
- Обновления: ОС «Альт» регулярно обновляется с учетом новых технологий и требований, что обеспечивает пользователям актуальную и безопасную систему [6].

ФСТЭК России предъявляет следующие требования к контейнеризации:

- Использование сертифицированных гипервизоров: ФСТЭК требует использования гипервизоров, имеющих сертификаты соответствия требованиям информационной безопасности. Эти сертификаты подтверждают, что гипервизор соответствует определенным стандартам безопасности и прошел необходимые тесты;
- Изоляция контейнеров: контейнеры должны быть изолированы друг от друга для предотвращения несанкционированного доступа к ресурсам других контейнеров. Это включает в себя изоляцию на уровне операционной системы, сети и файловой системы.
- Контроль доступа к контейнерам: должны быть реализованы механизмы контроля доступа к контейнерам, такие как аутентификация пользователей и авторизация на основе ролей. Это позволяет ограничить доступ к контейнерам только авторизованным пользователям [3].

Основные принципы работы «Kubernetes», такие как масштабируемость, автоматическое развертывание, самовосстановление, разделение ответ-ственности, гибкость и мониторинг и оповещения, соответствуют требованиям ФСТЭК и могут быть реализованы с

использованием отечественных операционных систем семейства Альт. Альт предлагает ряд преимуществ, включая поддержку российских криптографических алго- ритмов, соответствие требованиям регуляторов и наличие сертификатов соответствия ФСБ России и ФСТЭК России. Однако, для полного соответствия требованиям ФСТЭК может потребоваться дополнительная настройка и адаптация Kubernetes и Альт.

В результате анализа требований Федеральной службы по техническому и экспортному контролю (ФСТЭК) России к информационным системам можно сделать вывод о том, что использование технологии контейнеризации, и в частности, оркестратора контейнеров «Kubernetes», не противоречит данным требованиям.

Руководствуясь вышесказанным, можно определить этапы создания защищенного кластера, начиная с выбора подходящего оборудования и заканчивая настройкой безопасности и проверкой соответствия требованиям ФСТЭК.

1. Планирование: на данном этапе необходимо определить цели и задачи проекта, выбрать подходящий дистрибутив «Kubernetes» и определить требования к инфраструктуре.

2. Подготовка инфраструктуры: на этом этапе необходимо обеспечить соответствие инфраструктуры требованиям выбранного дистрибутива «Kubernetes».

3. Установка и настройка «Kubernetes»: включает в себя установку и настройку компонентов оркестратора, таких как kube-apiserver, kube-controller-manager, kubelet и др.

4. Разработка и развертывание приложений: на данном этапе происходит разработка и тестирование приложений, а затем их развертывание в кластере «Kubernetes».

5. Мониторинг и оптимизация: после внедрения необходимо осуществлять мониторинг работы кластера, выявлять и устранять возможные проблемы, а также оптимизировать работу приложений и «Kubernetes».

В будущем на основании результатов данного исследования, будет разработана методика внедрения защищенного кластера «Kubernetes», удовлетворяющий требованиям документов ФСТЭК России на базе отечественных операционных систем семейства «Альт».

Библиографический список

1. Kubernetes // kubernetes.io URL: https://kubernetes.io/ (дата обращения: 21.10.2023).

2. libraries // kubernetes.io URL: https://kubernetes.io/docs/reference/using-api/client-libraries/ (дата обращения: 21.10.2023).

3. Требования по безопасности информации к средствам контейнеризации // fstec.ru URL: https://fstec.ru/dokumenty/vse-dokumenty/spetsialnye-normativnye-

dokumenty/trebovaniya-po-bezopasnosti-informatsii-utverzhdeny-prikazom-fstek-rossii-ot-4-iyulya-2022-g-n-118 (дата обращения: 21.10.2023).

4. Основы Kubernetes // kubernetes.io URL: https://kubernetes.io/ru/docs/tutorials/kubernetes-basics/ (дата обращения: 21.10.2023).

5. Реализация отказоустойчивости в системе оркестрации микросервисной архитектуры Kubernetes / И.Р. Зулькарнеев, Д.П. Белов, Р.В. Крамаренко, И.А. Пелевин // Вестник УрФО. Безопасность в информационной сфере. – 2019. – №2(32). – С. 5–11. – DOI 10.14529/secur190201. – EDN HZXRFJ.

6. ОС «Альт СП» // basealt.ru URL: https://www.basealt.ru/alt-8-sp-sertifikat-fstehk/description (дата обращения: 21.10.2023).

УДК 004.056.2

РЕАЛИЗАЦИЯ АКТИВНОГО СКАНИРОВАНИЯ УСТРОЙСТВ НА БАЗЕ UCI[1]

А.Е. Баринов, Д.Д. Варапанова, В.П. Мартынов, Э.И. Филиппова
Научный руководитель: спец. по ЗИ А.Е. Баринов
Южно-Уральский государственный университет,
г. Челябинск

Рассмотрены подходы к активной инвентаризации сетевых устройств на базе извлечения конфигураций через удалённую консоль. Подробно описана система конфигурирования UCI. Разработан программный модуль, извлекающий информацию посредством UCI и сигнатуры, позволяющие обнаружить потенциальные угрозы информационной безопасности.

Ключевые слова: активное сканирование, контроль сетевых устройств, Telnet, SSH, управление конфигурациями.

Одним из ключевых направлений в обеспечении информационной безопасности вычислительной сети является контроль целостности сетевых активов и их взаимодействий. Наиболее критичной в этом направлении является задача инвентаризации. Эта задача может быть реализована, как на основе пассивного анализа трафика [1], так и дополняться методами активного сканирования, предполагающими отклик от устройств сети при сборе и анализе сетевой информации [2]. При пассивном сканировании могут быть пропущены или обработаны с недостаточной степенью достоверности, устройства, которые редко осуществляют сетевое взаимодействие, или такое взаимодействие может не содержать значимых сведений о версиях используемого ПО или конфигурациях [3]. Поэтому эти устройства становятся

[1] Исследование поддержано грантом Российского научного фонда (проект № 22-71-10095).

одними из наименее контролируемых сетевых активов. Единственным решением для оперативного сбора информации о таких устройствах становится активное сканирование сети. Таким образом, активное сканирование осуществляет постоянное тестирование системы с целью своевременного обнаружения аномалий в конфигурациях сетевых активов и состоянии их программного обеспечения. Так можно заблаговременно предположить развития обстановки, заметить неполадки системы на различных уровнях.

Преимущества активного мониторинга:

1. Полнота и точность данных: предоставляют данные о параметрах устройств: IP-адреса, MAC-адреса, модели, производителя, версии программного обеспечения и другие характеристики.

2. Отслеживание изменений: обнаружить новые устройства, которые были добавлены в сеть, или устройства, которые были удалены или заменены в реальном времени.

3. Оптимизация ресурсов: могут выявить неиспользуемые или малоиспользуемые устройства, которые можно отключить или заменить, выявить узкие места и проблемы производительности. Это может помочь снизить расходы на обслуживание сети.

4. Повышение безопасности всей системы.

В целом, проведение активных опросов устройств в задаче инвентаризации сети помогает обеспечить более полную, точную и актуальную информацию о состоянии сети, что важно для эффективного управления и обеспечения безопасности сетевой инфраструктуры.

Как правило, широко распространенным подходом по конфигурированию сетевых и встраиваемых устройств является применение удалённой консоли через SSH или Telnet. В целом, контроль конфигураций по протоколам SSH и Telnet является одним из универсальных подходов по отслеживанию состояния сетевых устройств.

Вместе все эти факторы делают активные опросы устройств важным инструментом в задаче инвентаризации сети. Они помогают обеспечить актуальную и полную информацию о сети, обогатить инвентаризационную информацию.

На рынке известны решения, которые обеспечивают активное сканирование посредством удалённой консоли [4, 5], однако в основном они работают с предопределённым списком производителей сетевого оборудования. При этом на рынке сетевых и встраиваемых систем, особенно специализированного назначения (АСУ ТП и т.д.) часто применимы решения на основе свободных консолей [6], таких как UCI (Unified Configuration Interface) [7] и Clixon [8]. В силу большей простоты решения на базе UCI более популярны, и вопросы активного сканирования таких устройств и буду рассмотрены в данной работе.

Система UCI (Unified Configuration Interface) – это система управления конфигурациями, которая используется в OpenWrt, операционной системе

для маршрутизаторов и других встроенных устройств. Принцип работы UCI основан на использовании текстовых файлов с определенным форматом для хранения конфигураций.

UCI обеспечивает удобный и единый способ управления конфигурациями в OpenWrt. Он позволяет администраторам легко проверять и изменять настройки устройств, а также отслеживать изменения и восстанавливать различные параметры системы. UCI предоставляет единый интерфейс для доступа к наиболее часто используемым конфигурационным файлам, расположенных в каталоге /etc/config/. Каждый файл относится к части системы, который он настраивает. UCI обрабатывает такие кнофигурационные файлы как: /etc/config/dhcp, /etc/config/dropbear, /etc/config/firewall, /etc/config/network, /etc/config/system, /etc/config/timeserver, /etc/config/wireless и этот список может быть расширен конкретным производителем устройства [7].

При доступе к ним через UCI информация удобно разделена на секции, параметры и значения, что позволяет удобно организовывать конфигурацию и обеспечивать простоту ее изменения. UCI также предлагает возможность проверки корректности конфигурационных файлов, что помогает избежать ошибок при внесении изменений и настройке системы.

Для инвентаризации конфигураций оборудования необходимо сначала эту конфигурацию извлечь. Для решения этой задачи был использован модуль Netmiko для языка программирования Python. Данный модуль позволяет работать с сетевым оборудованием, а также устанавливать SSH соединения с устройствами [9].

Для извлечения UCI конфигураций был использован данный модуль. Для начала было установлено соединение с виртуальной машиной с прошивкой OpenWrt, а затем отправлена команда для извлечения конфигурации UCI.

Код и результат работы программы представлен на рис. 1.

После извлечения конфигурации UCI необходимо её обработать с целью обнаружения аномалий и потенциальных угроз информационной безопасности. В извлеченной конфигурации возможно обеспечить проверку настройки отдельных компонентов, которые обеспечивают безопасную работу системы.

Примером одного из таких компонентов является аутентификация по паролю в сервисе SSH. Данный компонент находится в файле «/etc/config/dropbear», прописан строкой «option PasswordAuth» и может принимать значения ‘on’ или ‘off’.

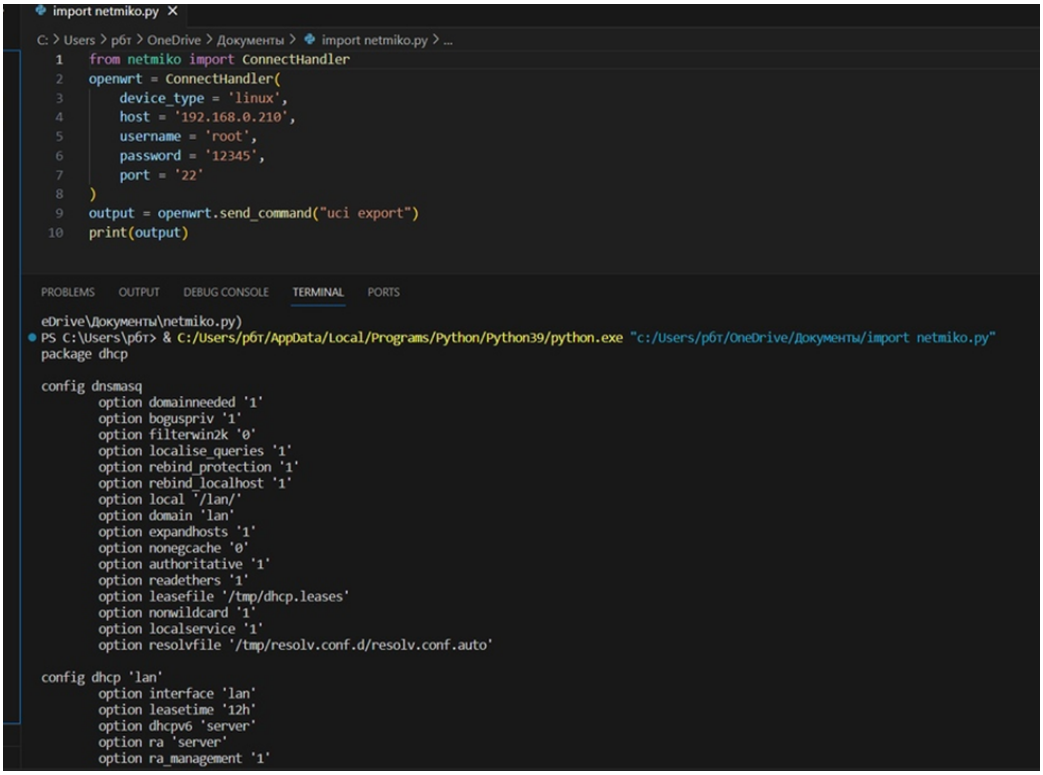


Рис. 1. Результат работы программы

Исходя из этих условий, проверка применяется ли аутентификация по паролю реализуется следующим кодом:

```
output = openwrt.send_command("uci export")
print(output)
if ("option PasswordAuth 'on'") in output :
    print('Аутентификация по паролю включена')
else:print('Аутентификация по паролю не включена')
```

Также при настройке безопасности сети стоит контролировать по каким портам можно подключаться к устройству. Данная информация хранится в файле «/etc/config/dropbear» и прописана строчкой «option Port», а значения может принимать целочисленные, например ‘22’. Поэтому реализация проверки выглядит, как вывод открытых портов. Код программы следующий:

```
output = openwrt.send_command("uci export")
print(output)
for x in output.split('\n'):
    if ("Port") in x:
        print('Подключение возможно выполнить по портам:',x[8::])
```

Таким образом, вся программа подключается к виртуальной машине по SSH, экспортирует полную конфигурацию UCI, а также выполняет проверки безопасности.

Результат работы программы приведен на рис. 2.

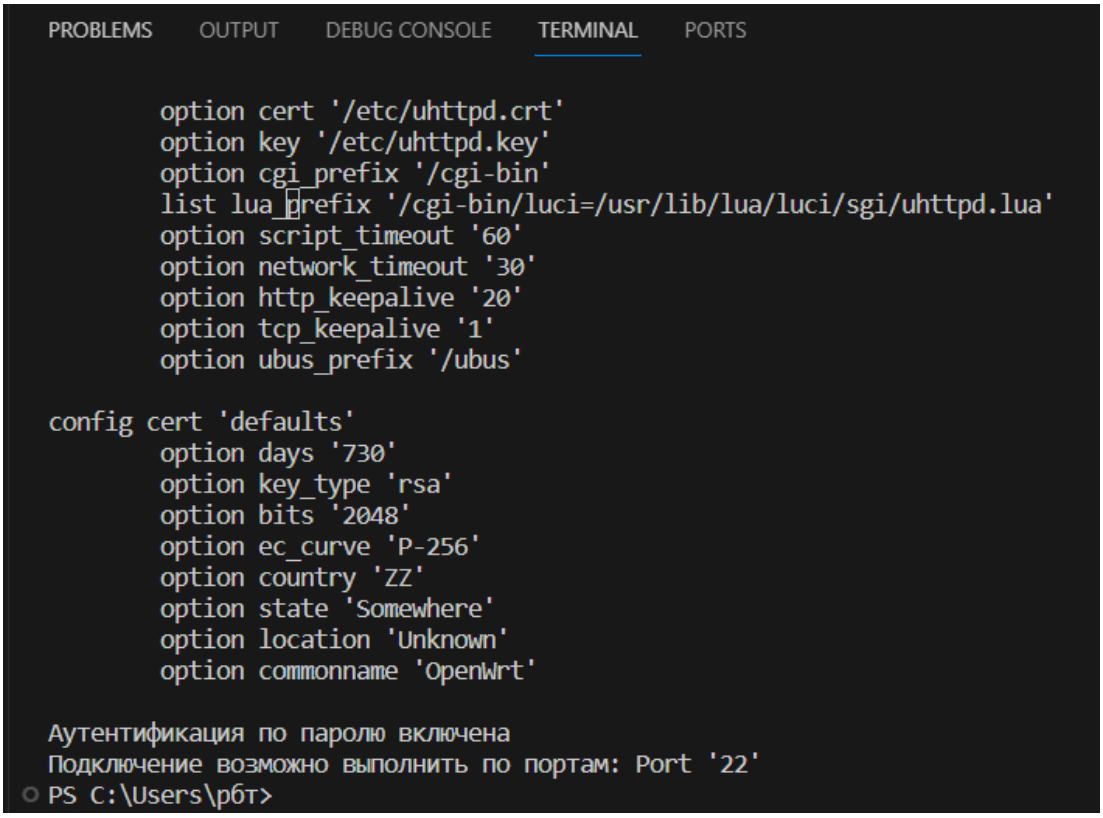


Рис. 2. Результат работы программы

Таким образом можно проверять множество компонентов сети, неправильные настройки которых могут привести к возникновению уязвимостей.

В ходе работы были изучены особенности работы системы UCI, а также разработан программный модуль активного сканирования, извлекающий информацию об запущенных сервисах и сетевых портах, а также обнаруживающий потенциально небезопасные настройки в некоторых сервисах. Дальнейшие исследования будут направлены на улучшение работы написанных программ и создание новых сигнатур для обнаружения аномалий в настройках безопасности системы.

Библиографический список

1. Montigny-Leboeuf, Annie & Massicotte, Frédéric. (2004). Passive Network Discovery for Real Time Situation Awareness. – 2004 – URL: https://archive.org/details/DTIC_ADA447338 (дата обращения: 18.10.2023).

2. Nicholson, Andrew & Janicke, Helge & Cau, Antonio. (2014). Safety and Security Monitoring in ICS/SCADA Systems. 10.14236/ewic/ics-csr2014.9. – 2014 URL: https://www.researchgate.net/publication/336585774_Position_Paper_Safety_and_Security_Monitoring_in_ICSSCADA_Systems (дата обращения: 18.10.2023).

3. Zhu, Feng & Mutka, Matt & Ni, Lionel. (2023). Classification of Service Discovery in Pervasive Computing Environments. – 2023 – URL: https://www.computer.org/csdl/magazine/pc/2005/04/b4081/13rRUxC0SBm (дата обращения: 19.10.2023).

4. Настройка и запуск активного опроса – [Электронный ресурс] – URL: https://support.kaspersky.com/KICSforNetworks/4.0/ru-RU/236152.htm (дата обращения: 21.10.2023).

5. Nozomy networks – [Электронный ресурс] – URL: http://downloads. nozomi-networks.com/bf67a4b0-7945-497c-b8d7-2caaa79e66e4/N2OS-ReleaseNotes-21.7.0. pdf (дата обращения: 20.10.2023).

6. Maiwe Communication | Connect the world, smarter the future – [Электронный ресурс] – URL: https://www.maiwe.com/ (дата обращения: 22.10.2023).

7. [OpenWrt Wiki] OpenWrt security hardening – [Электронный ресурс] – URL: https://openwrt.org/docs/guide-user/security/openwrt_security (дата обращения: 23.10.2023).

8. Clixon – [Электронный ресурс] – URL: https://www.clicon.org/ (дата обращения: 24.10.2023).

9. Модуль netmiko - Python для сетевых инженеров – [Электронный ресурс] – URL: https://pyneng.readthedocs.io/ru/latest/book/18_ssh_telnet/netmiko.html (дата обращения: 24.10.2023).

УДК 004.056.225

ИНСТРУМЕНТЫ БЕЗОПАСНОСТИ ПОДКЛЮЧЕННЫХ УСТРОЙСТВ НА БАЗЕ YOCTO-СОВМЕСТИМЫХ ДИСТРИБУТИВОВ

О.Е. Париев, А.Д. Зуев, А.Д. Сидоров, А.Е. Баринов
Научный руководитель: спец. по ЗИ А.Е. Баринов
Южно-Уральский государственный университет, г. Челябинск

Описано текущее состояние и мировые тенденции проблематики безопасности устройств интернета-вещей. Рассмотрены утилиты для обеспечения безопасности микроконтроллеров на дистрибутиве Linux «Yocto», предоставленные в слое «meta-security». Описаны основные функциональные возможности и проблемы интеграции наиболее популярных инструментов в дистрибутиве Yocto.

Ключевые слова: информационная безопасность, yocto, микроконтроллер, утилиты.

На сегодняшний день, количество устройств, подключенных к сети Интернет, оценивается в 16,7 миллиарда. Основными подходами по разработке встраиваемых устройств являются Yocto Project, Buildroot и OpenWrt. При этом разработка на Yocto становится всё более популярной за счёт гибкости и масштабируемости. Yocto Project – это проект Linux Foundation с открытым исходным кодом, целью которого является создание инстру-

ментов, позволяющих создавать дистрибутивы Linux для встроенного программного обеспечения и IoT, не зависящие от базовой архитектуры встроенного оборудования.

Однако обеспечение безопасности встраиваемых систем является сложной задачей – установка наложенных средств защиты и эффективное администрирование их конечным пользователем крайне ограничено, кроме того, встраиваемые устройства имеют длительный жизненный цикл эксплуатации, а цикл поддержки их производителями весьма ограничен.

В работе будут рассмотрены специальные инструменты и подходы для обеспечения безопасности, включенные в слой meta-security для Yocto-совместимых дистрибутивов. Слои в Yocto представляют собой наборы рецептов, конфигурационных файлов и метаданных, которые определяют, каким образом должна быть собрана и настроена операционная система. Рецепты BitBake определяют, как создается конкретный пакет. Рецепты состоят из URL источника (http, https, ftp, cvs, svn, git, локальная файловая система) пакета, зависимостей и параметров компиляции или установки. BitBake – похожий на make инструмент сборки, поддерживаемый проектами Yocto Project и OpenEmbedded. Опишем основные встроенные пакеты слоя meta-security:

1. Clam AntiVirus (ClamAV) [4] – пакет антивирусного ПО, который предназначен для защиты компьютеров и серверов от вредоносных программ. Он может работать на различных операционных системах, таких как Unix-подобные ОС, OpenVMS, Microsoft Windows и Apple Mac OS X. ClamAV является свободным ПО, лицензируется под GNU General Public License. Пример использования: *sudo clamscan -r /home.* В этом примере команда "clamscan" используется для сканирования всех файлов и папок в директории "/home" с целью обнаружения вредоносных программ.

2. Fail2Ban [5] – это программа для защиты серверов от атак методом грубой силы. Она написана на языке программирования Python и может работать на POSIX-системах, таких, как Linux, FreeBSD и других. Fail2Ban использует различные методы для блокировки IP-адресов, которые пытались несколько раз подобрать пароль или совершили другие нарушения безопасности.

Fail2ban считывает логи (например, /var/log/apache2/error.log) и блокирует IP-адреса, активность которых является подозрительной (например, большое количество попыток войти с неправильно введенным паролем, выполнение опасных или бессмысленных действий и т.д.). В случае обнаружения подобных действий программа обновляет правила брандмауэра для блокировки такого IP-адреса на определенный промежуток времени. Программа может быть настроена и для выполнения другого действия, например отправки электронного письма. Конфигурация по умолчанию содержит фильтры для Apache, Lighttpd, sshd, vsftpd, qmail, Postfix, Courier Mail Server, Asterisk и других популярных серверных приложений. В филь-

трах используются регулярные выражения, которые могут быть легко изменены и настроены в случае необходимости.

3. Firejail [6] – это программа-песочница, которая уменьшает риск безопасности, ограничивая среду запуска непроверенных приложений. Она использует функции Linux, такие, как пространства имен Linux и seccomp-bpf, чтобы создать изолированную среду выполнения для приложений. Пример использования: *firejail firefox.* Эта команда запускает веб-браузер Firefox в песочнице Firejail, обеспечивая дополнительный уровень безопасности при просмотре интернет-ресурсов.

4. eCryptfs [7, 8] – это криптографическая файловая система, разработанная для ядра Linux. Она предоставляет многоуровневое шифрование файлов и папок на уровне файловой системы, обеспечивая безопасность данных. Отличие eCryptfs от большинства других криптографических файловых систем в том, что все криптографические метаданные хранятся внутри зашифрованного файла. Это позволяет перемещать такие файлы через доверенные каналы, сохраняя возможность авторизованным лицам получить доступ к содержимому файлов.

В основе eCryptfs лежит формат файла OpenPGP, описанный в RFC2440 [1]. При этом, чтобы сохранить возможность произвольного доступа к данным в файле, разработчики отклонились от стандарта. Согласно формату OpenPGP, операции шифрования и расшифровывания должны производиться над всем содержимым файла. Это приводит к тому, что нельзя прочитать ни одного байта из файла до тех пор, пока он не расшифрован полностью. Чтобы обойти эту проблему и не ухудшить безопасность системы, eCryptfs разбивает данные на экстенты. По умолчанию, эти куски имеют размер страницы файловой системы (задается в ядре, как правило, это 4096 байт). Чтобы прочитать данные из одного куска, его нужно полностью расшифровать, а чтобы записать данные в блок, нужно шифровать весь блок. Пример форматирования раздела: *modprobe ecryptfs ecryptfs-setup-private --nopwcheck –noautomount.* Пример использования: *sudo mount -t ecryptfs /path/to/encrypted/dir /path/to/mount/point.* Эта команда монтирует зашифрованную директорию в указанную точку монтирования, позволяя доступ и работу с данными в зашифрованном виде.

5. AppArmor [9] – это программный инструмент упреждающей защиты, который определяет политики безопасности для приложений, определяя мандатный доступ приложений к системным ресурсам и привилегиям. Он обеспечивает контроль доступа и защиту от вредоносных программ. Пример использования: *sudo aa-status*. Эта команда показывает текущий статус AppArmor, показывая список активных профилей и перечень приложений, к которым они применяются.

6. PrivacyIDEA [10, 11] – это система двухфакторной аутентификации, которая поддерживает несколько клиентов и экземпляров. Она предоставляет дополнительный уровень безопасности при входе в систему, требуя

использования двух факторов для проверки подлинности. Пример использования: *privacyideaadm enroll token123 user1.* Эта команда регистрирует токен с идентификатором «token123» для пользователя «user1» в системе PrivacyIDEA.

7. nmap [12] – это мощный инструмент сканирования портов и определения состояния хоста в сети. Он позволяет обнаруживать устройства, определять открытые порты и настраивать сетевую безопасность. Пример использования: *nmap -p 80 192.168.1.1*. Эта команда сканирует хост с IP-адресом «192.168.1.1» для определения открытого порта 80.

8. pyrsa [13] – это библиотека на языке Python для работы с RSA-шифрованием. Она позволяет генерировать RSA-ключи, а также шифровать и дешифровать данные. Пример использования: *pyrsa-keygen -o private.pem.* Эта команда генерирует закрытый RSA-ключ и сохраняет его в файле «private.pem».

9. Radius [14] – это протокол аутентификации и учета, используемый для централизованного управления доступом к сети. Он позволяет контролировать доступ пользователей к сетевым ресурсам и проводить учет использования сетевых услуг. Пример использования: *radiusd –X.* Эта команда запускает сервер Radius в режиме отладки, позволяя просматривать входящие запросы и учтенные данные.

10. SoftHSM version 2 [15] – это программная реализация аппаратного модуля безопасности (HSM), используемого для защиты криптографических ключей и операций. Он предоставляет функциональность HSM на компьютере без наличия физического устройства. Пример использования: *softhsm2-util --init-token --slot 0 --label "MyToken" --pin 1234 --so-pin 5678.* Эта команда инициализирует HSM в SoftHSM version 2, создавая новый токен с именем "MyToken" и устанавливая PIN-коды для пользователей.

11. SSH [16] – это протокол удаленного доступа к компьютеру по сети, который обеспечивает защищенное соединение и шифрование данных. Он позволяет удаленно управлять компьютером, передавать файлы и выполнять команды на удаленном сервере. Пример использования: *ssh user@192.168.1.100*. Эта команда подключается по SSH к удаленному серверу с IP-адресом "192.168.1.100" от имени пользователя "user".

12. iptables [17] – это инструмент для настройки межсетевого экрана и маршрутизации в операционной системе Linux. Он позволяет управлять фильтрацией пакетов, пробросом портов и настройкой сетевой безопасности. Пример использования: *sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT*. Эта команда добавляет правило в цепочку INPUT, разрешая входящие TCP-соединения на порт 22 (используемый для SSH).

13. chkrootkit [18] – скрипт, разработанный в помощь системным администраторам для проверки системы на наличие известных руткитов. Поиск же осуществляется с помощью системных инструментов типа awk, cut, egrep и других. Утилита ищет подозрительные участки кода в программах,

сравнивает запущенные процессы с информацией из /proc, выявляя расхождения. Пример использования: *./chkrootkit*. Команда запускает полную проверку системы с использованием всех модулей, входящих в инструмент.

14. Lynis [19] – расширяемый инструмент аудита безопасности. Он проверяет все компоненты системы, выявляет слабые места и проблемы, и генерирует отчет с предупреждениями и рекомендациями для администратора. Большим плюсом является вычисление индекса защищенности, который помогает сориентироваться при выполнении рекомендаций. Пример использования: *lynis audit system*. Команда запускает сканнер и составляет подробный отчет, состоящий из настроек загрузки, настроек ядра, межсетевого экрана, параметров сети, активных портов, установленного ПО, доступных и запущенных сервисов.

15. AIDE [20] – средство проверки целостности файлов и каталогов. В настройках программы указывается расположение файлов и директорий для контроля, а также типы контролируемых параметров. AIDE отслеживает изменения контрольных сумм, типов файлов, пользователей, групп, прав доступа, inode, размеров файлов и времен. Гибкая конфигурация позволяет использовать AIDE для разных случаев. Создается база эталонных характеристик, и при проверке целостности сравниваются вычисленные значения с эталонами. В случае обнаружения изменений пользователь получит уведомление. Преимущества AIDE – легковесность и гибкость настройки, что позволяет интегрировать его в различные программные и программно-аппаратные комплексы под управлением Linux. Пример использования: *aide –init*. Команда инициализирует базу данных, в которую помещается «снимок» файлов и каталог.

16. Ncrack [21] – высокоскоростной инструмент для взлома сетевой аутентификации. Он был создан в помощь компаниям для защиты сетей путем упреждающего тестирования всех своих хостов и сетевых устройств на наличие слабых паролей. Пример использования: *ncrack -U usernames.txt -P passwords.txt ftp://10.10.0.50*. Команда запускает Brute force-атаку на ftp сервер по двум словарям, содержащим пользователей и пароли.

Заключение. В работе рассмотрены различные инструменты, способные обеспечить информационную безопасность устройств интернета вещей на основе Yocto-совместимых дистрибутивов. Описаны основные функциональные возможности всех рассмотренных инструментов.

В целом, представленные инструменты предоставляют надежный базовый набор для обеспечения информационной безопасности микроконтроллеров на базе дистрибутива «Yocto». Тем не менее, необходимо постоянно отслеживать новые угрозы и методы защиты, чтобы своевременно обеспечивать безопасность устройств.

Безопасность является ключевым аспектом развития устройств интернета вещей, и использование рассмотренных инструментов поможет нам избежать возможных проблем, связанных с безопасностью устройств.

Библиографический список

1. State of IoT Spring 2023, IoT Analytics Research Team – May 2023 – [Электронный ресурс] – URL: https://iot-analytics.com/number-connected-iot-devices/ (дата обращения 16.10.2023).
2. Alessandro Flaminio, Embedded Linux distro development with the Yocto Project – Turin, October 2018 – [Электронный ресурс] – URL: https://webthesis.biblio.polito.it/9085/1/tesi.pdf (дата обращения 17.10.2023).
3. Scott Murray, Security Hardening with OpenEmbedded / Yocto Project – 2020– [Электронный ресурс] – URL: https://www.konsulko.com/wp-content/uploads/2020/07/DD5_Security_Hardening_NA20.pdf (дата обращения 17.10.2023).
4. Cisco Systems, ClamAV Documentation – 2021 – [Электронный ресурс] – URL: https://docs.clamav.net/ (дата обращения 18.10.2023).
5. Fail2ban [Электронный ресурс]: Материал из Википедии – свободной энциклопедии: Версия 133220170, сохранённая в 18:38 UTC 23 сентября 2023 / Авторы Википедии // Википедия, свободная энциклопедия. – Электрон. дан. – Сан-Франциско: Фонд Викимедиа, 2023. – Режим доступа: https://ru.wikipedia.org/?curid=9127095&oldid=133220170 (дата обращения 19.10.2023).
6. LINUX-ORG-RU, FireJail – краткое и ознакомительное практическое руководство – 30.11.22 – [Электронный ресурс] URL: https://www.linux.org.ru/articles/desktop/17042789 (дата обращения 19.10.2023).
7. eCryptfs [Электронный ресурс]: Материал из Википедии – свободной энциклопедии: Версия, сохраненная в 10:49 5 января 2023 / Авторы Википедии энциклопедия. – Электрон. дан. – Сан-Франциско: Фонд Викимедиа, 2023. – Режим доступа: https://ru.wikipedia.org/wiki/ECryptfs (дата обращения 20.10.2023).
8. ArchWiki, eCryptfs – 1 November 2023 – [Электронный ресурс] – URL: https://wiki.archlinux.org/title/ECryptfs (дата обращения 20.10.2023).
9. RUVDS.com, Безопасный Linux вместе с AppArmor – 14 дек 2020 – [Электронный ресурс] – URL: https://habr.com/ru/companies/ruvds/articles/532988/ (дата обращения 20.10.2023).
10. Cornelius Kölbel, privacyIDEA Authentication System: Release 3.8 – Feb 20, 2023. – URL: https://buildmedia.readthedocs.org/media/pdf/privacyidea/latest/privacyidea.pdf (дата обращения 21.10.2023).
11. Wikipedia contributors. "PrivacyIDEA." Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia, 2 Jul. 2022. Web. 6 Nov. 2023. – [Электронный ресурс] – URL: https://en.wikipedia.org/wiki/PrivacyIDEA (дата обращения 22.10.2023).
12. “Fyodor” Lyon, Nmap Network Scanning: The Official Nmap Project Guide to Network Discovery and Security Scanning – 2022 – URL: https://nmap.org/book/. ISBN: 978-0-9799587-1-7. (дата обращения 23.10.2023).
13. Erik Ji, PyRSA – 2 Oct, 2021 – URL: https://github.com/Nobody912/PyRSA (дата обращения 23.10.2023).

14. RADIUS [Электронный ресурс]: Материал из Википедии – свободной энциклопедии: Версия, сохраненная в 13:43 22 сентября 2023 / Авторы Википедии энциклопедия. – Электрон. дан. – Сан-Франциско: Фонд Викимедиа, 2023. – Режим доступа: https://ru.wikipedia.org/wiki/RADIUS (дата обращения 23.10.2023).

15. Rickard Bellgrim, Francis Dupont, René Post, Roland van Rijswijk, GitHub репозиторий проекта SoftHSM version 2 – Sep 8, 2019 – [Электронный ресурс] – URL: https://github.com/opendnssec/SoftHSMv2 (дата обращения 25.10.2023).

16. Русскоязычное сообщество Ubuntu Linux, Iptables – 2018 – [Электронный ресурс] – URL: https://help.ubuntu.ru/wiki/iptables (дата обращения 25.10.2023).

17. Русскоязычное сообщество Ubuntu Linux, SSH – 2018 – [Электронный ресурс] – URL: https://help.ubuntu.ru/wiki/ssh (дата обращения 25.10.2023).

18. README файл инструмента chkrootkit – 2023 – [Электронный ресурс] – URL: https://www.chkrootkit.org/README (дата обращения 25.10.2023).

19. Lynis - Security auditing and hardening tool for Linux/Unix – [Электронный ресурс] – URL: https://cisofy.com/lynis/ (дата обращения 26.10.2023).

20. Enchanted Technology, AIDE – контроль целостности системных файлов – 2022 - [Электронный ресурс] – URL: https://wiki.enchtex.info/tools/security/aide (дата обращения 27.10.2023).

21. Nmap Software LLC, Ncrack – [Электронный ресурс] – URL: https://nmap.org/ncrack/ (дата обращения 28.10.2023).

УДК 004.056.5

АНАЛИЗ РИСКОВ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ В ПРОЦЕССЕ ЭКСПЛУАТАЦИИ ПОДКЛЮЧЕННЫХ ТРАНСПОРТНЫХ СРЕДСТВ

И.А. Шевяков, А.Н. Соколов
Научный руководитель: канд. техн. наук, доц. А.Н. Соколов
Южно-Уральский государственный университет, г. Челябинск

Развитие технологий в автомобильной и транспортной отрасли привело к необходимости обработки транспортным средством (ТС) значительных объемов информации, связанной с его функционированием и взаимодействием с устройствами дорожной инфраструктуры, участников дорожного движения и другими устройствами, подключающимися к бортовой информационной сети ТС. В связи с этим актуальной стала задача исключения рисков информационной безопасности, влияющих на безопасность дорожного движения, в процессе производства и эксплуатации ТС. В работе проанализирован процесс оценки рисков, связанных с уязвимостями в системах управления транспортными средствами, такими как возможность удаленного взлома и несанкционированного доступа.

Ключевые слова: информационная безопасность, кибербезопасность, транспортное средство, анализ рисков, безопасность дорожного движения.

Безопасность информации в транспортных средствах играет важную роль в обеспечении безопасности дорожного движения. В настоящее время многие автомобили и другие транспортные средства оснащены различными электронными системами, включая системы связи, навигации, контроля и автоматизации [1].

Если информационная безопасность не обеспечена, возникают следующие угрозы:

1. Взлом управления: возможность удаленного вмешательства в работу автомобиля может привести к авариям, кражам или другим опасным ситуациям на дороге.

2. Подделка данных: киберпреступники могут подделывать данные на приборной панели, изменять показания скорости, топлива или других важных параметров, что может привести к неправильным решениям водителя и опасным ситуациям.

3. Несанкционированный доступ к личным данным: хакеры могут получить доступ к данным о водителе или пассажирах, включая финансовую информацию или идентификационные данные, что может привести к краже личности или мошенничеству.

4. Отказ систем: в случае атаки или неисправности, системы безопасности, такие как ABS (антиблокировочная система тормозов), ESP (электронная система стабилизации) или системы предупреждения столкновений, могут выйти из строя, что повышает риск аварии.

Поэтому безопасность информации в транспортных средствах является важным аспектом обеспечения безопасности дорожного движения. Производители и операторы транспортных средств должны принимать меры по защите и обеспечению безопасности систем передачи данных и других электронных систем.

Стандарт ISO/SAE 21434 «Road vehicles – Cybersecurity Engineering» определяет требования и рекомендации по инженерии кибербезопасности для автомобильной промышленности [2]. Он не прямо предусматривает использование конкретных метрик для оценки эффективности мер защиты информации в транспортных средствах. Однако, стандарт содержит рекомендации относительно условий, при которых можно проводить оценку эффективности мер защиты информации.

Раздел 9 стандарта ISO/SAE 21434 называется «Управление рисками» и содержит руководство по управлению рисками в области кибербезопасности транспортных средств. Раздел включает следующие основные аспекты:

1. Определение рисков: данный раздел описывает процесс определения рисков, связанных с кибербезопасностью автомобилей. Включает оценку угроз, определение уязвимостей и оценку последствий инцидентов.

2. Анализ рисков: здесь описываются методы анализа рисков, которые позволяют оценить вероятность возникновения инцидентов и их последствия. Это позволяет выделить самые значимые риски для дальнейшего управления ими.

3. Управление рисками: данный раздел описывает процесс принятия мер по снижению рисков. Включает в себя разработку и внедрение защитных мер, проведение тестирования на безопасность и мониторинг рисков на протяжении всего жизненного цикла автомобиля.

4. Оценка эффективности мер: здесь описываются методы оценки эффективности принятых мер по снижению рисков. Это позволяет определить, насколько успешно были применены защитные меры и внесены изменения для повышения безопасности.

Целью анализа рисков является оценка воздействия сценария угрозы и возможности атаки по каждому пути атаки. Анализ рисков состоит из оценки воздействия и анализа атак [3]. Целью оценки воздействия является оценка величины ущерба, причиненного нарушением свойств кибербезопасности активов. Деятельность по анализу атак в основном включает в себя анализ пути атаки и оценку возможности атаки. Учитывая, что проблемы автомобильной кибербезопасности, снижают безопасность вождения и безопасность конфиденциальных данных пользователей ТС, можно определить [4, 5] связанные с этим воздействия: на безопасность S (safety), на финансы F (finance), на эксплуатацию O (operation) и на конфиденциальность P (privacy). Параметры оценки воздействия можно определить количественно в соответствии с соответствующими отраслевыми стандартами, такими как ISO 26262-3:2018. Параметры оценки воздействия в этой схеме относятся к параметрам метода HEAVENS, как показано в табл. 1 [6].

Таблица 1

Параметры оценки воздействия на автомобильную кибербезопасность

S		F		O		P	
Уровень	Значение	Уровень	Значение	Уровень	Значение	Уровень	Значение
Не воздействует	0	Не воздействует	0	Не воздействует	0	Не воздействует	0
Низкий	10	Низкий	10	Низкий	10	Низкий	10
Средний	100	Средний	100	Средний	100	Средний	100
Высокий	1000	Высокий	1000	Высокий	1000	Высокий	1000

В соответствии с оцененными параметрами воздействия можно вычислить сумму для получения уровня воздействия, как показано в табл. 2. Уравнение выражается следующим образом:

$$I = S+F + O + P, \quad (1)$$

где I – общая величина воздействия.

Таблица 2

Уровень воздействия на автомобильную кибербезопасность

Сумма значений параметров	Уровень	Значение
1-19	Низкий	1
20-99	Средний	2
100-999	Высокий	3
≥1000	Критический	4

Анализ путей атак используется для определения потенциальных путей атак, а затем связывает их со сценариями угроз. Оценка осуществимости атаки применяется для оценки простоты использования каждого пути атаки. В процессе анализа пути атаки и оценки возможности атаки следует учитывать изменения в среде угроз и доступной информации в течение жизненного цикла транспортного средства. Затем применяются три метода оценки возможности атаки: метод, основанный на потенциале атаки, метод, основанный на возможности использования CVSS, и метод, основанный на векторе атаки. Выбор подхода к оценке возможности атаки зависит от этапа жизненного цикла транспортного средства и доступной информации. Когда уровень осуществимости атаки или уровень воздействия равен 0, потенциального риска нет. В данной статье уровень воздействия и уровень осуществимости атаки разделены на 4 уровня (1–4).

Таблица 3

Параметры осуществимости атаки, основанные на потенциале

Параметр				Значение
EX	KN	WI	EQ	
Непрофессионал	Общественный	Критический	Стандартное	0
Опытный	Ограниченный	Высокий	Специализированное	1
Эксперт	Чувствительный	Средний	Выполненное на заказ	2
Несколько экспертов	Критический	Низкий	Широкое разнообразие заказного	3

Таблица 4

Уровень осуществимости атаки на основе потенциала

Сумма значений параметров	Уровень	Значение
7-9	Низкий	1
4-6	Средний	2
2-3	Высокий	3
0-1	Критический	4

Потенциал атаки взят из стандарта ISO/IEC 18045:2008 и был переопределен с учетом характеристик автомобилей. На основании этого параметры оценки (экспертность (EX), знания о TOE (KN), окне возможностей (WI), оборудовании (EQ)) осуществимости атаки определяются согласно методу HEAVENS, как показано в табл. 3. Аналогичным образом, сумма может быть вычислена для получения уровня осуществимости атаки в соответствии с параметрами, показанными в табл. 4. Учитывая, что значение уровня осуществимости атаки равно 0, когда сумма значений параметров превышает 9, в статье это не анализируется. Уравнение записывается следующим образом:

$$AF = EX + KN + WI + EQ\,, \tag{2}$$

где AF – общее значение осуществимости атаки.

На раннем этапе разработки транспортного средства возможность атаки может быть качественно оценена на основе вектора атаки, когда доступной информации недостаточно для определения конкретного пути атаки. Векторы атак можно разделить на 4 категории: сетевые, смежные, локальные и физические, как показано в табл. 5 [7]. Уровень осуществимости атаки возрастает с увеличением удаленности пути атаки. Метод, основанный на возможности использования CVSS, может определяться группой показателей возможности использования в базовых метриках CVSS. Группа показателей возможности использования включает в себя 4 параметра: вектор атаки (V), сложность атаки (C), требуемые привилегии (P) и взаимодействие с пользователем (U), как показано в табл. 6 и 7. В методе, основанном на возможности использования CVSS, уравнение выражается следующим образом [7]:

$$E = 8{,}22 \times V \times C \times P \times U\,, \tag{3}$$

где E – значение возможности эксплуатации уязвимости.

Для определения уровня риска проводится оценка уровня воздействия сценариев ущерба и уровня осуществимости путей атаки. Затем в можно рассчитать значения рисков, сформировав матрицу рисков, которая будет использоваться для оценки рисков. В вышеупомянутых методах оценки

рисков большая часть уровней риска определяется матрицей рисков. Построение матрицы рисков в основном зависит от опыта оценки, без количественного анализа. В данном исследовании алгоритм глобального рейтинга [8] используется для построения матрицы рисков автомобильной кибербезопасности, как показано в табл. 8. Значение риска рассчитывается следующим образом:

$$R = \sqrt{(m(I)2 + n(AF))2}\ , \tag{4}$$

где R – величина риска, m и n – весовые параметры I и AF соответственно.

Предполагается, что факторы воздействия и осуществимости атаки имеют одинаковый вклад в риск. Таким образом, m и n оба установлены равными 0,5.

Уровень риска также следует определять на основе уровня воздействия и уровня возможности атаки. Значения риска, рассчитанные по формуле. (4) может быть использовано для построения матрицы рисков кибербезопасности. Уравнение выражается следующим образом:

$$RL = F(IL, AL)\ , \tag{5}$$

где RL – значение уровня риска; IL – значение уровня воздействия; AL – значение уровня осуществимости атаки; F представляет собой функцию риска IL и AL.

Таблица 5

Уровень осуществимости на основе вектора атаки

Параметр	Уровень	Значение уровня
Физический	Низкий	1
Локальный	Средний	2
Смежный	Высокий	3
Сетевой	Критический	4

Таблица 6

Параметры возможности атаки при использовании CVSS

Параметр	Значение
V	0.2-0.85
C	0.44-0.77
P	0.27-0.85
U	0.62-0.85

Таблица 7

Уровень осуществимости атаки при использовании CVSS

Значение возможности использования	Уровень	Значение уровня
0.12-1.05	Очень низкий	1
1.06-1.99	Низкий	2
2.00-2.95	Средний	3
2.96-3.89	Высокий	4

Таблица 8

Матрица рисков автомобильной кибербезопасности

Уровень риска		Уровень воздействия			
		1	2	3	4
Уровень осуществимости атаки	1	1	2	2	3
	2	2	2	2	3
	3	2	2	3	3
	4	3	3	3	4

Заключение. Оценка рисков кибербезопасности помогает определить состояние безопасности автомобильных систем и выявить требования к системам кибербезопасности автомобилей. В исследовании приведена система оценки рисков автомобильной кибербезопасности, которая включает в себя процесс оценки рисков и методы систематической оценки рисков, применимую ко всем этапам жизненного цикла транспортного средства. Был создан комплексный метод для анализа активов, угроз и путей кибербезопасности автомобилей. В результате была получена матрица рисков, которую можно применять при оценке рисков любых узлов подключенного транспортного средства. Кроме того, предлагаемая структура включает показатели оценки воздействия и осуществимости атак, а также количественную матрицу рисков кибербезопасности, что помогает повысить объективность оценки рисков.

Библиографический список

1. Juan Deng, Lu Yu, Yu Fu, OluwakemiHambolu, Richard R. Brooks, Chapter 6 - Security and Data Privacy of Modern Automobiles // Data Analytics for Intelligent Transportation Systems, Elsevier, 2017, Pages 131–163.

2. ISO 26262-1:2018 Road vehicles – Functionalsafety [Электронный ресурс]. – Режим доступа: https://www.iso.org/standard/68383.html – (дата обращения: 28.10.2023).

3. Wang, Y., Wang, Y., Qin, H. et al. A Systematic Risk Assessment Framework of Automotive Cybersecurity. Automot. Innov. 4, 253–261 (2021). https://doi.org/10.1007/s42154-021-00140-6.

4. Ruddle, A., Ward, D., et al.: Security requirements for automotive on-board networks based on dark-side scenarios. EVITA deliverable D2.3, EVITA project (2009).

5. Islam, M.M., Lautenbach, A., Sandberg, C., et al.: A risk assessment framework for automotive embedded systems. 2nd ACM International Workshop on Cyber-Physical System Security, AMC. (2016).

6. HEAVENS, Security models v2.0. (2016).

7. CVSS: Common vulnerability scoring system version 3.1. specification document, Accessed 27 Oct. 2019, from https://www.first.org/cvss/v3-1/cvss-v31-specification_r1.pdf.

8. Li, J.P., Bao, C.B., Wu, D.S.: How to design rating schemes of risk matrices: a sequential updating approach. Risk Analysis. 38(1), 99–117 (2018).

УДК 004.45 + 004.454/457 + 004.7 + 004.49

РЕАЛИЗАЦИЯ СЕТЕВОГО ДРАЙВЕРА ДЛЯ ОС СЕМЕЙСТВА WINDOWS NT, СКРЫТЫЙ КАНАЛ СВЯЗИ С УДАЛЁННЫМ СЕРВЕРОМ

Д.А. Милицкая
Научный руководитель: ст. преподаватель И.А. Маткин
Челябинский государственный университет, г. Челябинск

На данный момент реализация скрытого канала связи имеет большое значение для обеспечения безопасности, эффективности и расширяемости коммуникаций между различными устройствами и системами, т. к. скрытый канал связи позволяет передавать данные между устройствами, минимизируя риск их перехвата или прослушивания третьими сторонами. Важно заметить, что скрытость повышает устойчивость к блокировке канала передачи данных. Целью данной работы является разработка NDIS драйвера, обеспечивающего передачу сетевых пакетов по скрытому каналу связи и взаимодействие с удалённым сервером. В данной статье описаны наиболее значимые моменты разработки сетевого драйвера-фильтра, реализующего скрытый канал связи с удалённым сервером и возможность внедрения произвольного сетевого трафика. Согласно поставленным задачам в работе рассмотрена архитектура сетевых драйверов в Windows, описаны основные компоненты и функции, а также проделана разработка и отладка драйвера, обеспечивающего передачу сетевых пакетов по скрытому каналу связи. В процессе изучения был сделан вывод о том, что наибольший уровень скрытости позволяет реализовать NDIS драйвер-фильтр. При этом его разработка с целью создания скрытого канала связи является не тривиальной задачей и имеет мало-

известные особенности, отличающие подход к разработке от классического подхода к разработке драйверов-фильтров.

Ключевые слова: канал связи, сетевые драйверы, удалённый сервер.

Сетевые драйвера в Windows имеют достаточно широкое применение и отвечают за управление сетевыми интерфейсами и обработку сетевых пакетов в операционной системе. Существует 2 категории сетевых драйверов в Windows: TDI-драйверы (Transport Driver Interface) и NDIS-драйверы (Network Driver Interface Specification) [1]. В данной работе выбор пал на разработку NDIS драйвера, так как он предоставляет доступ к более низкоуровневому интерфейсу ОС и позволяет

1) анализировать сетевой трафик,

2) блокировать получение и отправку сетевых пакетов с определёнными характеристиками,

3) отправлять произвольные пакеты в сеть,

4) осуществлять скрытую передачу пакетов без открытия новых портов и создания новых сетевых интерфейсов в ОС (виртуальных адаптеров).

Разработанный драйвер реализован в виде NDIS драйвера-фильтра, этот тип драйвера не создаёт собственный интерфейс, он подключается к существующему сетевому интерфейсу как сетевая служба, после чего может осуществлять перехват и модификацию всех пакетов, проходящих через интерфейс. Данная особенность обеспечивает дополнительную скрытость.

Имеется возможность установки разработанного драйвера посредством netcfg с использованием описанных в INF файле настроек, после которой, как видно на снимке экрана (рис. 1), он автоматически подключается к поддерживаемым интерфейсам.

```
PS C:\Users\admin> netcfg /l C:\Users\admin\Desktop\FilterCursARP\Driver\NdisFilter\NdisFilter.inf -v -c s -i ms_ndisfilter
Попытка установки ms_ndisfilter ...

... C:\Users\admin\Desktop\FilterCursARP\Driver\NdisFilter\NdisFilter.inf скопирован в C:\Windows\INF\oem70.inf.

... выполнено.

PS C:\Users\admin> Get-NetAdapterBinding

Name                    DisplayName                                         ComponentID          Enabled
----                    -----------                                         -----------          -------
Ethernet0               Служба доступа к файлам и принтерам сетей Micro...  MS_Server            True
Ethernet0               Протокол мультиплексора сетевого адаптера (Майк...  ms_implat            False
Ethernet0               Драйвер протокола LLDP (Майкрософт)                 ms_lldp              True
Ethernet0               Ответчик обнаружения топологии канального уровня    ms_lltdio            True
Ethernet0               Клиент для сетей Microsoft                          ms_msclient          True
Ethernet0               OUT4.0                                              ms_ndisfilter        True
```

Рис. 1. Установка посредством netcfg

В процессе разработки реализован скрытый канал связи с удалённым сервером (для этого выполнена своя реализация протокола UDP).

С использованием канала реализована передача перехваченного трафика удалённому серверу, отправка пакетов и команд компьютеру для взаимодействия с разработанным драйвером.

Для удалённого общения с драйвером был разработан скрипт с использованием библиотеки языка python под названием Scapy.

На данной схеме (рис. 2) представлена общая схема работы созданного ПО.

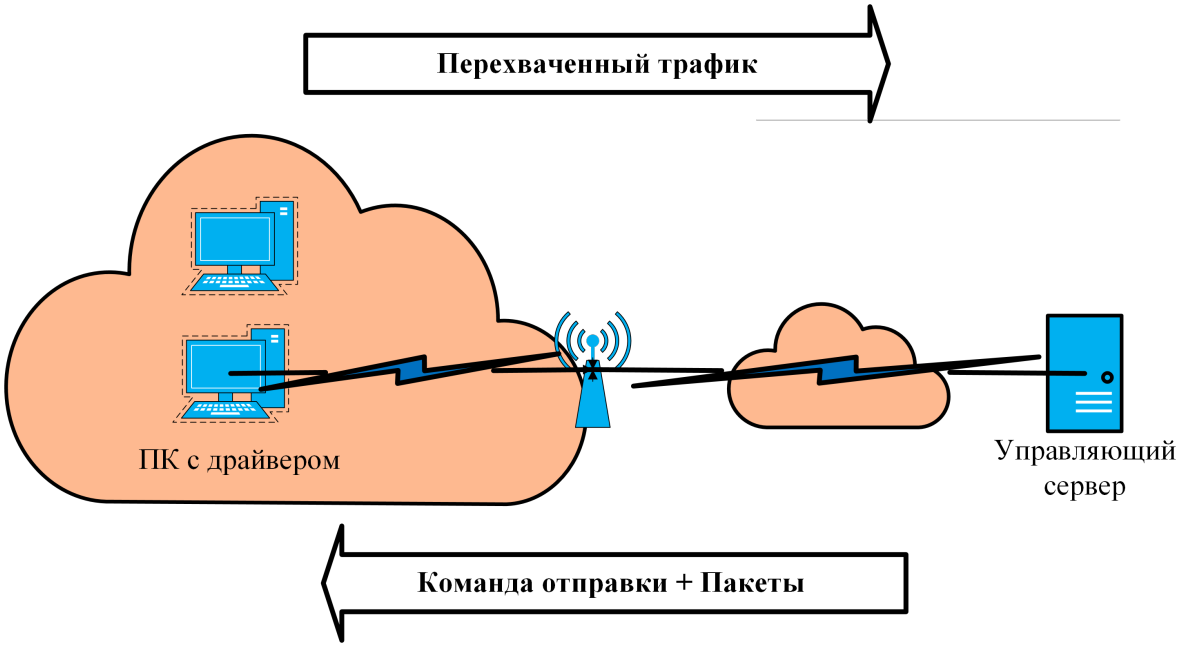


Рис. 2. Общая схема работы созданного ПО

Во-первых, весь трафик, перехваченный драйвером, передаётся на удалённый сервер по созданному нами каналу.

Во-вторых, на удалённом сервере формируются необходимые для решения конкретной задачи пакеты, затем через скрытый канал связи вместе с командой отправки в UDP пакете они передаются нашему драйверу, драйвер разбирает полученные пакеты и передаёт их содержимое в подсеть компьютера, на котором он установлен.

Реализация канала обмена информацией между драйвером и удалённым сервером состоит в следующем: Разработан протокол передачи, инкапсулированный в собственную реализацию UDP. Пакеты имеют структуру, указанную на схеме (рис. 3).

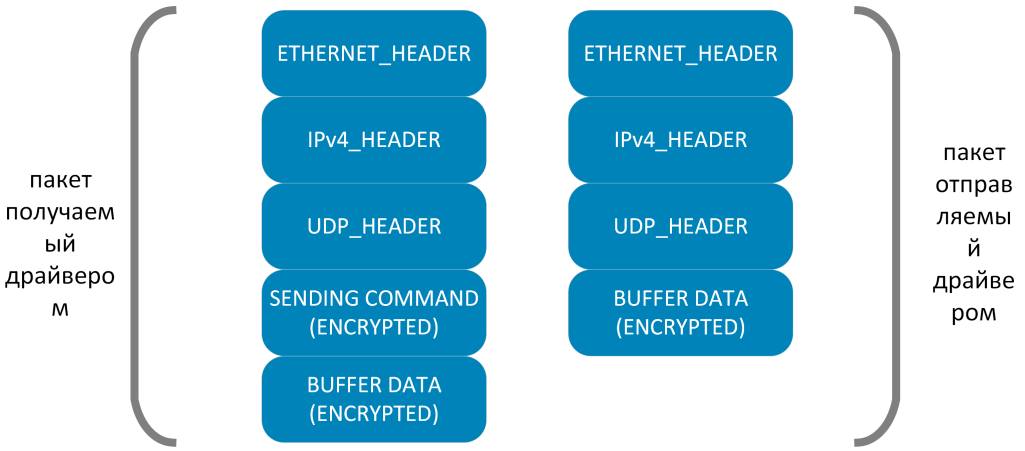


Рис. 3. Пакеты обмена данными между драйвером и удалённым сервером

Заполнение пакетов происходит следующим образом. Для того чтобы сообщить драйверу о необходимости отправки пакета в теле UDP, указывается числовой код отправки пакета (SENDING COMMAND) длиной 1 байт, затем, заполняется буфер с последовательностью байт пакета. Тело пакета шифруется, вычисляются проверочные суммы. Перехваченные драйвером пакеты помещаются в теле UDP. Тело пакета шифруется, вычисляются проверочные суммы.

Классическая схема разработки NDIS драйвера типа фильтр включает в себя следующие этапы:

1. Выбор типа сетевого драйвера (в данной работе было принято решение о разработке NDIS драйвера типа фильтр).
2. Установка необходимой версии WDK, настройка среды разработки и тестирования.
3. Определение настроек установки драйвера, написание INF файла.
4. Разработка функций инициализации и завершения работы драйвера.
5. Разработка функций фильтрации отправляемых и получаемых пакетов.
6. Разработка интерфейса конфигурации драйвера из пользовательского (User Mode) приложения.
7. Сборка и установка драйвера на компьютерную систему для тестирования.

Кроме перечисленного в списке выше, была реализована функция MyPacketSendIoCtl отправки произвольного пакета, которая отличает разработку нашего драйвера от классического подхода к разработке драйверов–фильтров и позволяет генерировать произвольный сетевой трафик. MyPacketSendIoCtl принимает следующие параметры: номер порта, флаги отправки пакета, указатель на буфер с данными, размер буфера. NDIS драйверы имеют функцию для отправки (PNET_BUFFER_LIST) листа пакетов в сеть – NdisFSendNetBufferLists, было принято решение использовать её для отправки произвольных пакетов. В MyPacketSendIoCtl для создания нового MDL с нашими битами используется NdisAllocateMdl, затем указатель на созданную MDL передаётся в функцию NdisAllocateNetBufferAndNetBufferList, которая создает новые NET_BUFFER и NET_BUFFER_LIST для отправляемого пакета. NdisFSendNetBufferLists принимает на вход HANDLE фильтра, указатель на созданный для отправляемого пакета NBL, номер NDIS порта и флаги отправки.

Кроме того, в ходе разработки драйвера выяснилось, что при внедрении трафика в фильтре с помощью NdisFSendNetBufferLists необходимо удалить внедрённый трафик из листа в функции FilterSendNetBufferLists Complete до вызова NdisFSendNetBufferListsComplete, либо освободить лист, состоящий полностью из внедрённого трафика и сделать возврат из функции.

Реализованное в данной работе ПО может выполнять функции VPN канального уровня с некоторыми ограничениями на скрытость. Кроме того, подобный канал может служить инструментом проведения некоторых хорошо известных сетевых атак на компьютеры [2], находящиеся в одной подсети с ПК, на котором установлен драйвер, например возможно осуществление ARP-спуфинг атаки, ARP и TCP–SYN сканирования сети, что было успешно проделано в рамках моей последней курсовой работы при тестировании разработанного драйвера на виртуальной сети.

Библиографический список

1. Сорокина С.И. Программирование драйверов и систем безопасности: учеб. пособие / Светлана Сорокина, Андрей Тихонов, Андрей Щербаков. – СПб.: БХВ-Петербург; М.: Молгачева С. В., 2002. – 241, [1] с.: ил., табл.; 23 см.; ISBN 5-94157-263-8 (Изд-во "БХВ-Петербург").

2. Зейтц Джастин, Арнольд Тим, Black Hat Python: программирование для хакеров и пентестеров. 2-е изд. – СПб.: Питер, 2022. – 256 с.: ил. – (Серия «Библиотека программиста»). ISBN 978-5-4461-3935-4.

УДК 004

ЗАЩИТНЫЕ МЕХАНИЗМЫ В СОВРЕМЕННЫХ ОС СЕМЕЙСТВА WINDOWS NT

П.Ю. Стародубов
Научный руководитель: ст. преподаватель И.А. Маткин
Челябинский государственный университет, г. Челябинск

В работе приведен анализ современных защитных механизмов, внедренных в операционные системы семейства Windows NT. Исследованы низкоуровневые детали их реализации, выявлены некоторые слабые места. Описаны предложенные и реализованные методы обхода данных механизмов.

Ключевые слова: Windows NT, защитные механизмы.

Введение. В операционные системы семейства Windows NT добавляется множество защитных механизмов, призванных уберечь пользователей от действий вредоносного программного обеспечения. Изучение реализации и принципов работы данных механизмов, а также поиск уязвимостей и методов обхода представляют большой интерес.

Цель данной работы: исследовать реализацию защитных механизмов и предложить методы обхода.

Для достижения цели были поставлены следующие задачи:

- изучить современные защитные механизмы,

- исследовать низкоуровневые детали их реализации,
- предложить методы обхода,
- реализовать предложенные методы.

Control Flow Guard. Control Flow Guard (CFG) – механизм безопасности, снижающий риск эксплуатации уязвимостей, связанных с повреждением памяти, путём защиты косвенных вызовов функций. Перед каждым косвенным вызовом добавляются инструкции, которые проверяют, является ли цель допустимой целью вызова. Если цель не является допустимой приложение завершается.

CFG реализован и поддерживается в Windows с версий Windows 8.1 Update 3 и Windows 10. Компиляция программ с поддержкой CFG доступна в Microsoft Visual Studio 2015 [1].

Control Flow Guard обеспечивается работой, проделываемой на всех этапах:

1) при создании кода компилятор с поддержкой CFG перед каждым косвенным вызовом функции генерирует вызов __guard_check_icall, которая проверяет валидность адреса назначения (рис. 1);

2) на этапе загрузки ОС основную работу выполняет функция nt!MiInitializeCfg. Ее основная задача создать разделяемую память, содержащую битовую карту адресов;

3) на этапе загрузки образа функции nt!MiRelocateImage и nt!MiCfgInitializeProcess заполняют битовую карту адресов;

4) на этапе исполнения перед каждым косвенным переходом вызывается функция проверки ntdll!LdrpValidateUserCallTarget, которая валидирует адрес, используя информацию из битовой карты.

Без CFG:

```
mov     ecx, 1
call    [rsp+38h+var_10]
```

CFG x86

```
mov     [ebp+var_4], eax
mov     ecx, [ebp+var_4]
call    ds:___guard_check_icall_fptr
call    [ebp+var_4]
```

CFG x64

```
mov     ecx, 1
mov     rax, [rsp+48h+var_18]
call    cs:__guard_dispatch_icall_fptr
```

Рис. 1. Проверки, добавляемые CFG

Исследование позволило выявить недостатки механизма и реализовать универсальный метод обхода. Метод заключается в том, что удаленный процесс может модифицировать битовую карту адресов [2]. Таким образом, можно пометить все адреса валидными для перехода и эксплуатировать уязвимость в программе, защищенной CFG.

Arbitrary Code Guard. Arbitrary Code Guard (ACG) – функция безопасности, предназначенная для предотвращения запуска произвольного кода в контексте доверенных процессов.

ACG гарантирует соблюдение двух принципов:

1) код неизменяем;

2) данные не могут стать кодом.

На этапе исполнения ядро отслеживает все попытки процесса управления своей виртуальной памятью. Проверки ACG реализованы в функциях, которые вызываются при выделении памяти, смене атрибутов защиты и загрузке в процесс исполняемого модуля [3]. Каждая из функций вызывает nt!MiArbitraryCodeBlocked, которая предотвращает нарушение политик ACG.

Подробное изучение механизма позволило предложить и реализовать два метода обхода. Первый метод заключается в загрузке вредоносных библиотек. ACG способен защитить процесс от загрузки библиотек, содержащих секции с правами RWX, но не может предотвратить загрузку DLL без таких секций, но содержащих вредоносный код. Второй метод – выделение исполняемой памяти из другого процесса. Было выявлено, что разрешение на создание динамического кода проверяется у текущего процесса, а не у процесса, в контексте которого находится данная память. Это означает, что любой процесс с привилегией отладки может выделять исполняемую память в контексте другого процесса, защищённого ACG.

Code Integrity Guard. Code Integrity Guard (CIG) – защитный механизм, внедренный в операционные системы Windows 10 и Windows Server 2016, который гарантирует, что все двоичные файлы, загружаемые в процесс, имеют цифровую подпись Microsoft.

За проверку цифровых подписей в операционной системе Windows отвечает модуль ядра ci.dll. Code Integrity Guard выполняет проверку подписи каждого загружаемого бинарного файла, а поэтому его работа встроена в процесс загрузки библиотек. Функция ci!CiValidateImageHeader вычисляет хэш каждой страницы или всего образа и сравнивает его с расшифрованным значением хэша из исполняемого файла. Если проверка подписи не пройдет, функция вернет код ошибки STATUS_INVALID_IMAGE_HASH.

Исследование механизма позволило выяснить, что подпись исполняемого файла проверяется только на этапе загрузки, а дальнейшее состояние никак не отслеживается. Это позволило реализовать два метода обхода. Первый – внедрение кода из процесса с привилегией отладки. Данный метод аналогичен методу для обхода ACG. Второй метод – эксплуатация уязвимости Time of Check, Time of Use. Проверка подписи и отображение файла в память разделены по времени, что позволяет изменить содержимое библиотеки между этими моментами. Также было выявлено, что выгруженный из рабочего набора образ подгружается обратно без проверки подписи. Таким образом, подмена файла после загрузки и последующее инициирование выгрузки рабочего набора позволяет обойти механизм проверки подписи.

Protected Processes Light. Protected Processes Light (PPL) – механизм, который защищает антивирусные программы и критически важные службы

Windows от несанкционированного доступа даже администраторами. PPL внедрен в Windows 8.1 и Windows Server 2012 R2. PPL является расширением классических Protected Processes, ориентированных на защиту цифрового контента.

В основе PPL находится многоуровневая модель, обеспечиваемая цифровыми подписями исполняемых файлов. Файл может загрузиться как PPL, только если имеет соответствующую цифровую подпись (рис. 2). При этом разные подписывающие стороны имеют различные уровни доверия, некоторые PPL имеют более высокую защиту, чем другие. PPL на одном уровне может открыть для полного доступа любой процесс на том же уровне подписи или ниже. Доступ к более привилегированным процессам возможен только с ограниченными правами.

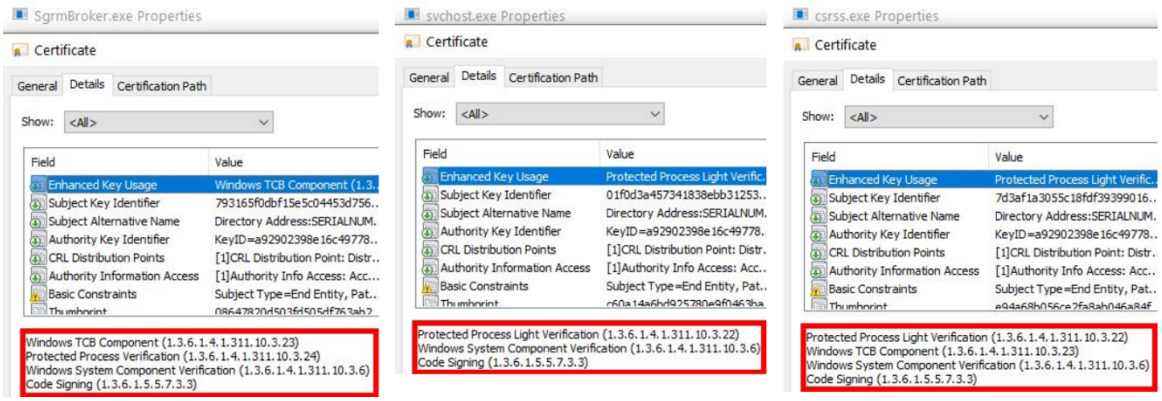


Рис. 2. Цифровые подписи для PPL

Проверки прав при доступе к процессу осуществляет функция ядра nt!RtlTestProtectedAccess. Если значение защиты текущего процесса меньше значения защиты целевого процесса, доступ не разрешается.

Поскольку механизм обхода проверки подписи кода справедлив для всех процессов пользовательского режима, модифицировав исходный сценарий атаки, была получена возможность выполнения произвольного кода в контексте PPL, что позволило получить учетные данные из памяти процесса lsass в режиме PPL [4].

Заключение. В ходе данной работы был проведен анализ современных защитных механизмов, внедрённых в операционные системы семейства Windows NT. Особое внимание было уделено механизмам Control Flow Guard, Arbitrary Code Guard, Code Integrity Guard и Protected Processes Light. Были изучены их низкоуровневые детали реализации, реализованы методы обхода данных механизмов. Также удалось выяснить важную информацию о работе механизма проверки подписи кода, которая может быть полезной для дальнейших исследований механизма проверки подписи драйверов.

Библиографический список

1. Внутреннее устройство Windows. 7-е изд. / М. Руссинович [и др.] – СПб.: Питер, 2018. – 944 с.

2. Bypass Control Flow Guard Comprehensively / Zhang Yunhai. URL: https://www.blackhat.com/docs/us-15/materials/us-15-Zhang-Bypass-Control-Flow-Guard-Comprehensively-wp.pdf (дата обращения 25.10.2023).

3. Disable Dynamic Code Mitigation (ACG). URL: https://blog.sevagas.com /IMG/pdf/code_injection_series_part4.pdf (дата обращения 25.10.2023).

4. PPLdump Is Dead. Long Live PPLdump / Gabriel Landau. URL: https://i.blackhat.com/Asia-23/AS-23-Landau-PPLdump-Is-Dead-Long-Live-PPLdump.pdf (дата обращения 25.10.2023).

УДК 004.056.53

РАЗРАБОТКА МЕТОДИКИ АВТОМАТИЗИРОВАННОГО КРИМИНАЛИСТИЧЕСКОГО АНАЛИЗА ДАННЫХ ОПЕРАТИВНОЙ ПАМЯТИ ДЛЯ ВЫЯВЛЕНИЯ ИНЦИДЕНТОВ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Д.С. Крысин, М.В. Малый, В.В. Гладнев
Научный руководитель: канд. техн. наук Ю.С. Тимошенкова
Уральский федеральный университет
имени первого Президента России Б.Н. Ельцина,
г. Екатеринбург

В работе рассмотрена проблема идентификации и обнаружения и идентификации инцидентов информационной безопасности на персональных компьютерах путем анализа данных дампа памяти оперативного запоминающего устройства данные. Проведен анализ существующих решений, представлены их преимущества и недостатки. На основе полученных результатов анализа авторами разработана методика автоматизированного анализа данных оперативной памяти для выявления и идентификации инцидентов информационной безопасности и несанкционированного доступа к персональным компьютерам.

Ключевые слова: анализ оперативной памяти, идентификация инцидентов, обнаружение несанкционированного доступа.

Введение. Открытые сводные данные, предоставляемые Генеральной прокуратурой, показывают примерное количество регистрируемых преступлений по рассмотренным статьям Уголовного кодекса Российской Федерации (УК РФ). Статистика по преступлениям показана в табл. 1.

Количество таких преступлений возросло. Данное обстоятельство обусловлено, прежде всего, широким внедрением новых средств электронных

платежей, которые еще недостаточно освоены как физическими, так и юридическими лицами, переходом на удаленный режим работы ряда организаций.

Основные причины роста числа преступлений с использованием информационных технологий:

- прибыльность крайне высока, это связанно с большими суммами денег, к которой нарушители получают доступ в результате проведенных операции как с частными лицами, так и с юридическими лицами;
- количество электронных платежей населения с каждым годом увеличивается, это приводит к активному развитию интернет-банкинга, а следовательно, и к росту численности сетевых преступников;
- недостаточная грамотность пользователей и низкая защищенность смартфонов и планшетов, позволяют злоумышленникам с помощью вредоносных программ организовывать похищение денежных средств со счетов пользователей при обработке ими финансовых трансакций.

Таблица 1

Сводная статистика преступлений в сфере компьютерной информации за 2017–2022 годы

Статьи УК РФ	Год					
	2017	2018	2019	2020	2021	2022
ст. 159.3 УК РФ «Мошенничество с использованием электронных средств платежа»	4596	4453	4087	3301	3582	3946
ст. 272 УК РФ «Неправомерный доступ к компьютерной информации»	74	50	85	84	133	179
ст. 273 УК РФ «Создание, использование и распространение вредоносных компьютерных программ»	128	79	76	45	77	46
ст. 274 УК РФ «Нарушение правил эксплуатации средств хранения, обработки или передачи компьютерной информации и информационно-телекоммуникационных сетей»	0	0	4	8	15	55

Пандемия COVID-2019, самоизоляция и перевод сотрудников на удаленный режим работы привели к взрывному росту IT-преступности. Основная их доля имеет отношение с совершением финансовых преступлений, связанных с хищением денег или данных банковских карт.

По оценкам аналитиков Group-IB, во время пандемии прежде всего выросло число финансовых мошенничеств с использованием методов социальной инженерии. Основные сценарии, используемые мошенниками: валютные ограничения, частичная мобилизация, мошенничество под видом государственных органов.

Согласно [2] количество преступлений, в области кибербезопасности в 2022 году достигло 4226, что на 11 % больше, чем в 2021 году (3807). Такие показатели говорят о необходимости разработок для защиты информации на электронных носителях и персональных компьютеров (ПК). Таким образом целью работы является анализ инструментов расследования инцидентов информационной безопасности (ИБ), связанных использованием ПК и разработка методики обнаружения следов инцидентов информационной безопасности в памяти ПК.

Обзор существующих подходов к решению проблемы. Необходимость разработки системы криминалистического анализа для выявления признаков инцидентов ИБ обусловлена повышением эффективности поиска и идентификации данных в оперативной памяти компьютерной системы и формированием предложений по устранению инцидентов. Необходимость разработки также обусловлена тем, что существующие инструменты для анализа изображения операционной являются платными и не всегда соответствуют требованиям. Сравнительная характеристика программного обеспечения (ПО) представлена в табл. 2.

Таблица 2

Сравнительная характеристика ПО для анализа данных ОЗУ

Наименование	Стоимость	ОС	GUI	Страна-производитель
Belkasoft Evidence Center	платно	Windows	+	Россия
Elcomsoft Forensic Disk Decryptor	платно	Windows	+	Россия
Passware Kit Forensic	платно	Windows	+	Россия
Guidance EnCase	платно	Windows	+	США
Volatility Framework	бесплатно	Windows/ Linux	-	Россия
Mandiant's Memoryze	платно	Windows	+	США
Volafox	платно	Linux	+	США
MAC Memory Reader	бесплатно		-	США
Rekall	платно	Windows/ Linux	+	США

Основной трудностью в расследовании инцидентов ИБ является сбор доказательств и создание доказательной базы, а также обеспечение неизменности и целостности доказательств. Это связано с тем, что при расследовании инцидентов ИБ большая часть доказательств – это информация, хранящаяся в накопителях информации или в виде данных, файлов или записей в оперативном запоминающем устройстве (ОЗУ), базе данных (БД) или служебных областях операционной системы ПК.

Предоставленная информация в достаточной степени подвержена преднамеренному или непреднамеренному уничтожению. Такую инфор-

мацию часто легко подделать, поскольку ложная информация, хранящаяся в цифровом виде, не отличается от реальной информации. Кроме того, фальсификация таких доказательств должна определяться либо семантическим содержанием информации, либо следами в другом месте. Не всегда удается обеспечить сохранность следов при хранении. И не только гарантировать, но и доказать эту неизменность.

Наиболее успешными в этом аспекте являются компьютеры, работающие на платформах Sparc, где есть встроенная утилита OpenBoot, создающая копии данных RAM, однако такие утилиты доступны к использованию не всегда.

Как следствие возникает необходимость в создании программного инструмента, который не использует функции ОС при записи дампа оперативной памяти на внешний носитель. В настоящее время таких программных средств и утилит для компьютерных систем, работающих на платформах Intel и AMD, не разработано. Поэтому данная задача является актуальной.

Разработка методики автоматизированного криминалистического анализа данных ОП для выявления инцидентов ИБ. Основными целями реагирования на инциденты ИБ являются минимизация ущерба, скорейшее восстановление исходного состояния ИС и разработка плана по недопущению подобных инцидентов в будущем. Цели достигаются на двух основных этапах: расследование инцидента и восстановление системы.

Для существующих систем принято рассматривать следующие этапы реагирования на инциденты ИБ:

- подготовка;
- идентификация;
- локализация;
- ликвидация;
- восстановление;
- выводы.

Рассматривается этап идентификации, а именно захват, анализ и оценка.

На основании этого был разработан алгоритм процесса идентификации инцидентов ИБ и реагирования на них при возникновении, события требующего реагирования (рис. 1), алгоритм состоит из следующих действий:

1. Снять образ ОЗУ (дамп ОЗУ).
2. Выполнить анализ и оценку полученного образа ОЗУ.
3. Если инцидент ИБ не имеет вредоносного воздействия тревога считается ложной. Иначе выполняется анализ ПК для оценки находится ли под контролем ПК.
4. Если заданное функциональное состояние изменилось, применяются антикризисные действия. Иначе проводится мониторинг инцидента

5. Когда инцидент ИБ находится под контролем (действие нарушителя не изменило исходное состояние ПК и дальнейшие воздействие невозможно), проводится мониторинг инцидента ИБ и идентифицируются данные нарушителя, такие как: IP-адрес, MAC-адрес и т. п.

6. Если невозможно определить данные нарушителя применяются антикризисные меры, а также при необходимости блокируется злоумышленник и его контактные соединения, чтобы снизить риск повреждения ИТ-системы.

В данном случае под антикризисными действиями понимается отключение пораженного компьютера.

7. При получении идентификационных признаков нарушителя (используя специализированное ПО) подключаются организации, уполномоченные на законодательном уровне для дальнейших мероприятий.

В итоге будут получены сведения, которые необходимо учесть для анализа и улучшения системы, а также усовершенствования системы защиты и обработки инцидентов ИБ на ПК. Данная информация, полученная в процессе расследования инцидента ИБ, должна быть отправлена для анализа шаблонов, которые могут значительно способствовать раннему выявлению инцидентов ИБ и предупреждению о последующих инцидентах ИБ, которые могут возникнуть на основе прошлого опыта.

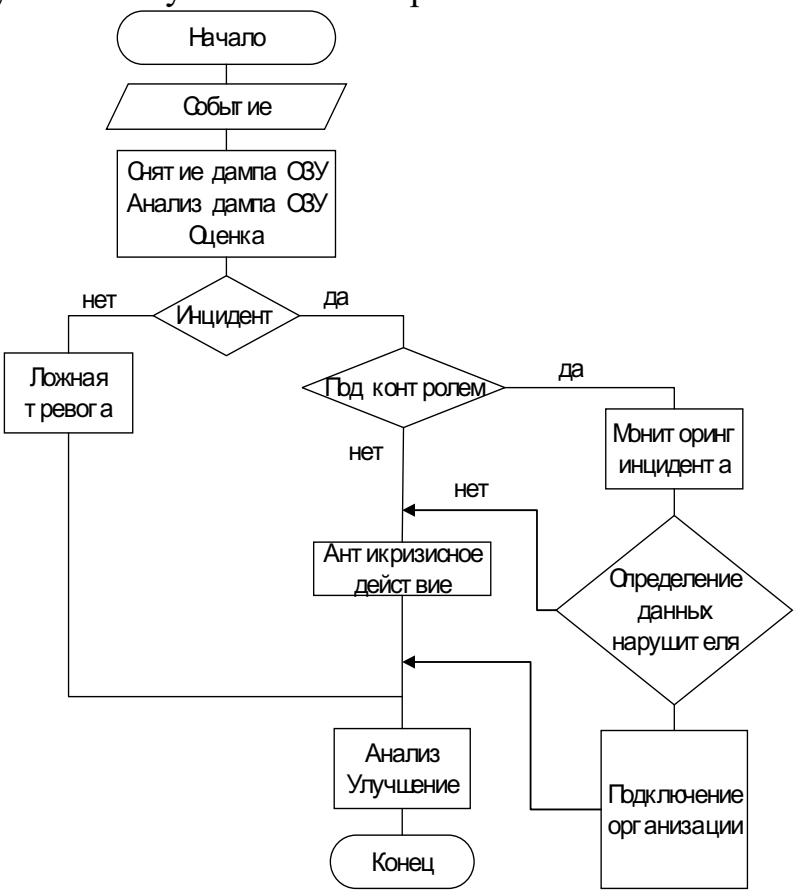


Рис. 1. Схема алгоритма процесса идентификации инцидентов ИБ и реагирования на них

Необходимо проверить не только атакованный компьютер, но и другие компьютеры в организации. Уязвимости, обнаруженные в ходе тестирования, необходимо исправить, чтобы предотвратить дальнейшие атаки. Также необходимо ввести мониторинг и периодическую проверку с помощью системы идентификации символов инцидентов ИС на основе данных из памяти свободного доступа всех персональных компьютеров организации в автоматическом или полуавтоматическом режиме. Все эти изменения также должны предотвратить повторение этих инцидентов или их эффективное расследование.

Заключение. В результате работы были рассмотрены существующие методы и ПО для обнаружения и реагирования на инциденты ИБ. В ходе анализа существующих решений были выделены основные преимущества и недостатки, которые позволили сформулировать методику автоматизированного криминалистического анализа данных оперативной памяти. Разработанная методика представлена в работе. Перспективным направлением исследования является реализация методики с использованием бесплатного отечественного ПО, с удобным графическим пользовательским интерфейсом. Анализ образов оперативной памяти для выявления инцидентов ИБ используя предложенную схему алгоритм и бесплатное программное обеспечение, может значительно повысить надежность расследования инцидентов ИБ.

Библиографический список

1. Доктрина информационной безопасности Российской Федерации: утв. указом Президента РФ от 05.12.2016 № 646; введ. 05.12.2016 № 646. – Москва. – 10 с.

2. Показатели по отдельным категориям дел // Судебная статистика РФ. – Режим доступа: https://stat.апи-пресс.рф/. (дата обращения: 17.10.2023).

3. Российская Федерация. Законы. О безопасности критической информационной инфраструктуры Российской Федерации: федер. закон от 26.07.2017 № 187-ФЗ: [принят Гос. Думой 12 июля 2017 г.: одобр. Советом Федерации 19 июля 2017 г.] // Официальный интернет-портал правовой информации: сайт. – в ред. Федерального закона от 03.08.2018 № 323-ФЗ. – Электрон. дан. – 2005–2020. – Режим доступа: http://publication.pravo.gov.ru/document/0001201808030102. (дата обращения: 17.10.2023).

4. Российская Федерация. Законы. Уголовный кодекс Российской Федерации: [принят Гос. Думой 24 мая 1996 г.: одобр. Советом Федерации 5 июня 1996 г.] // Официальный интернет-портал правовой информации: сайт. – в ред. Федерального закона от 31.07.202 № 260-ФЗ. – Электрон. дан. – 2005–2020. – Режим доступа: http://publication.pravo.gov.ru/document/0001202007310012. (дата обращения: 17.10.2023).

УДК 004.02

МЕТОДЫ ПОИСКА, ИДЕНТИФИКАЦИИ И АНАЛИЗА УЯЗВИМОСТЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ В ОТКРЫТЫХ ИСТОЧНИКАХ ИНФОРМАЦИИ

М.В. Малый, В.В. Гладнев, Д.С. Крысин
Научный руководитель: канд. техн. наук, доц. О.А. Пономарева
Уральский федеральный университет
имени первого Президента России Б.Н. Ельцина,
г. Екатеринбург

В данной работе был проведен обширный анализ методов и средств для поиска информации об уязвимостях в различном программном обеспечении в открытых источниках информации. Были изучены различные подходы к сбору информации, включая автоматизированные средства и методы ручного поиска. Также были проанализированы различные источники информации, которые могут содержать данные об уязвимостях программного обеспечения. Кроме того, были рассмотрены требования к руководящим документам для оценки соответствия требованиям безопасности программного обеспечения и информационных систем. Были изучены стандарты и методические рекомендации, по оценке безопасности программного обеспечения.

Ключевые слова: база данных, безопасность, идентификация, открытые источники, мониторинг, уязвимость.

Введение. В современном мире информационная безопасность является одним из ключевых аспектов защиты как корпоративных, так и частных информационных систем. Одной из наиболее распространенных угроз в этой области являются уязвимости в программном обеспечении. Для обеспечения безопасности информационных систем необходимо постоянно отслеживать изменения в компонентах системы и оперативно принимать меры по их устранению. В данной работе будет рассмотрено актуальное направление в области информационной безопасности – методы идентификации, поиска и анализа уязвимостей программного обеспечения на основе информации из открытых источников. Результаты работы могут быть полезны для IT-специалистов, ответственных за информационную безопасность, а также для всех, кто интересуется защитой своих информационных ресурсов.

Целью данного проекта является идентификация уязвимостей программного обеспечения по сведениям, полученным из открытых источников информации.

Данный проект имеет актуальную практическую ценность в свете увеличения числа кибератак и угроз информационной безопасности. Изучение уязвимостей позволит принять меры к улучшению уровня защиты информационных систем и снизить риски возможных утечек данных, финансовых потерь и репутационных ущербов для организаций и частных лиц.

Анализ методов и средств поиска сведений об уязвимостях программного обеспечения в открытых источниках информации. Для эффективного поиска сведений об уязвимостях программного обеспечения в открытых источниках информации, используются различные методы и средства. Рассмотрим некоторые из них.

1. Сканирование уязвимостей, которое заключается в автоматическом сканировании сети на наличие уязвимостей, используя специальные программы и инструменты.

Сканирование уязвимостей является одним из наиболее популярных методов поиска сведений об уязвимостях программного обеспечения в открытых источниках информации. Этот метод заключается в автоматическом сканировании сети на наличие уязвимостей при помощи специальных программ и инструментов [1].

Программы для сканирования уязвимостей могут использоваться как для сканирования внутренней сети организации, так и для сканирования внешних систем, доступных из Интернета. Они могут осуществлять сканирование портов и сервисов, идентифицировать уязвимости и проверять соответствие настроек безопасности.

2. Анализ кода программного обеспечения на предмет уязвимостей и ошибок в коде, который является важным методом для обеспечения безопасности программного обеспечения. Он заключается в исследовании и анализе кода программного обеспечения с целью выявления возможных уязвимостей и ошибок в коде, которые могут привести к нарушению безопасности системы.

Основным преимуществом анализа кода является возможность выявления уязвимостей, которые могут быть пропущены другими методами поиска, например, сканированием уязвимостей сети. Кроме того, анализ кода позволяет выявить проблемы в дизайне и архитектуре системы, которые могут привести к уязвимостям в будущем.

Однако, проведение анализа кода может быть трудоемким и требует специальных знаний и навыков. Кроме того, такой метод может привести к большому количеству ложных срабатываний, что требует дополнительной работы для их проверки и устранения.

3. Использование баз данных уязвимостей, представляющие собой сборники информации о различных уязвимостях в программном обеспечении, которые были обнаружены и описаны в открытых источниках. Они содержат подробное описание уязвимостей, включая

описание того, как эти уязвимости могут быть использованы для атаки на системы, а также рекомендации по устранению уязвимостей. Примерами подобных баз данных могут быть:

– National Vulnerability Database (NVD) – это база данных, которая содержит информацию о тысячах уязвимостей в различных продуктах, включая операционные системы, приложения и другие программные средства. База данных NVD поддерживается Национальным институтом стандартов и технологий (NIST) США, и она обновляется ежедневно с использованием данных, предоставляемых производителями программного обеспечения и другими источниками;

– Common Vulnerabilities and Exposures (CVE) – это база данных уязвимостей, которая содержит информацию о тысячах уязвимостей в различных продуктах. Каждая уязвимость в базе данных имеет уникальный идентификатор (CVE ID), который используется для ссылки на уязвимость в различных источниках. База данных CVE поддерживается организацией MITRE Corporation и используется многими организациями во всем мире для оценки безопасности и управления рисками.

Использование баз данных уязвимостей может помочь организациям быстро и эффективно выявлять уязвимости в своих системах и принимать меры по устранению этих уязвимостей до того, как они будут использованы злоумышленниками для атаки на информационную систему.

4. Специализированные форумы и ресурсы в интернете, где специалисты по безопасности обмениваются информацией о новых уязвимостях и методах атаки на информационные системы. Эти ресурсы могут быть как открытыми, доступными всем желающим, так и закрытыми, где доступ разрешен только для зарегистрированных пользователей.

Одним из наиболее известных открытых ресурсов является сайт «Exploit Database», на котором публикуются эксплойты и общедоступные уязвимости. Кроме того, существуют форумы, такие как «Full Disclosure», где специалисты обмениваются информацией о новых уязвимостях и обсуждают различные методы атаки на информационные системы.

Однако стоит отметить, что наличие таких ресурсов также может быть использовано злоумышленниками для получения информации о уязвимостях и методах атаки на информационные системы, поэтому необходимо соблюдать осторожность и принимать меры для защиты своей системы.

5. Использование белых и серых методов тестирования на проникновение. Эти методы позволяют проводить тестирование на проникновение в информационную систему с разрешения владельца системы.

Белое тестирование на проникновение (или «белый ящик») подразумевает проведение тестирования при наличии полной информации

о системе, включая ее исходный код, конфигурационные файлы и другую информацию. Тестировщик, проводящий белое тестирование, может использовать различные инструменты и техники для поиска уязвимостей в системе, такие как сканирование портов, анализ исходного кода, тестирование аутентификации и авторизации и т.д. Белое тестирование позволяет выявить наибольшее количество уязвимостей в системе, но требует доступа к полной информации о системе, что может быть недоступно в реальных условиях.

Серое тестирование на проникновение (или «серый ящик») подразумевает проведение тестирования при наличии ограниченной информации о системе. Тестировщик, проводящий серое тестирование, имеет доступ только к открытой информации о системе, такой как информация на сайте компании, информация об аутентификации и т.д. Тестировщик может использовать различные техники для поиска уязвимостей, такие как перебор паролей, сканирование веб-страниц и т.д. Серое тестирование позволяет оценить уязвимости системы в условиях ограниченной информации и является более реалистичным, чем белое тестирование.

6. Использование специализированных служб по поиску уязвимостей. Эти службы могут проводить как автоматическое, так и ручное тестирование на проникновение, использовать инструменты сканирования уязвимостей, анализировать логи, конфигурационные файлы и другую информацию, чтобы выявить потенциальные уязвимости.

Руководящие документы, которые содержат требования к оценке соответствия требованиям безопасности программного обеспечения и информационных систем. Эти документы определяют стандарты и методы оценки безопасности информационных систем, которые могут использоваться для оценки соответствия системы требованиям безопасности.

Некоторые из наиболее распространенных руководящих документов в этой области включают:

1. Стандарты безопасности информации ISO – это серия стандартов, разработанных Международной организацией по стандартизации (ISO), которые содержат требования и рекомендации по управлению безопасностью информации в организациях. Они определяют принятые международные стандарты в области безопасности информации и помогают организациям создавать и поддерживать системы управления безопасностью информации [2].

2. Стандарты IEC (International Electrotechnical Commission) – это серия стандартов, разработанных Международной электротехнической комиссией, которые определяют требования к безопасности систем управления промышленными процессами и контроллерами. Эти стандарты предназначены для обеспечения безопасности и надежности систем автоматического

управления промышленными процессами в различных отраслях, включая энергетику, химическую и нефтегазовую промышленность, производство и транспорт.

3. Стандарты NIST – это наборы документов, разработанных Национальным институтом стандартов и технологий (NIST) США, которые определяют методы и руководства по оценке безопасности информационных систем. Эти стандарты имеют широкое применение во всем мире и используются во многих отраслях, включая правительственные организации, бизнес и научные учреждения.

4. OWASP ASVS (Application Security Verification Standard) представляет собой набор требований к безопасности веб-приложений, созданный Открытым проектом по безопасности веб-приложений (OWASP). Он определяет различные уровни требований к безопасности, которые могут быть применены для оценки соответствия требованиям безопасности веб-приложений.

5. Стандарт PCI DSS (Payment Card Industry Data Security Standard) – это международный стандарт безопасности данных, разработанный в индустрии платежных карт. Он определяет требования к защите данных, связанных с процессом обработки платежей, и регулирует обработку данных, связанных с кредитными картами, дебетовыми картами и другими формами электронной оплаты.

6. CIS Controls (Center for Internet Security Controls) – это стандарт, который определяет набор контрольных мер безопасности для защиты информационных систем от кибератак. Этот стандарт был создан с целью предоставить организациям набор практических и конкретных инструкций по обеспечению безопасности и защите своих информационных систем.

Руководящие документы по оценке соответствия требованиям безопасности программного обеспечения и информационных систем могут помочь организациям создать и улучшить свои системы управления информационной безопасностью, а также обеспечить соответствие законодательным и регуляторным требованиям в области безопасности информации. Эти документы также могут использоваться для оценки безопасности программного обеспечения и информационных систем, чтобы выявить уязвимости и улучшить их защиту.

Кроме того, руководящие документы по оценке соответствия требованиям безопасности информационных систем могут включать различные методы оценки безопасности, которые могут быть использованы для проверки соответствия требованиям безопасности.

Методы тестирования безопасности могут включать тестирование на проникновение, тестирование на соответствие стандартам и регуляторным требованиям, тестирование на действительность и тестирование на переносимость. Эти методы позволяют проверить защищенность системы от уязвимостей и атак со стороны злоумышленников.

Методы аудита могут включать проверку соответствия процессов управления информационной безопасностью требованиям международных стандартов, таких как ISO 27001, а также проверку процедур управления доступом и управления конфигурацией.

Методы проверки уязвимостей могут включать сканирование уязвимостей и пентестинг (проверка на проникновение). Эти методы могут помочь идентифицировать уязвимости и слабые места в системе, которые могут быть использованы злоумышленниками для атак.

В целом, методы оценки безопасности, определенные в руководящих документах, могут помочь организациям выявлять уязвимости и улучшать защиту своих информационных систем и программного обеспечения. Они также помогают обеспечить соответствие законодательным и регуляторным требованиям в области безопасности информации.

В дополнение к методам оценки безопасности, руководящие документы также могут определять процедуры оценки соответствия требованиям безопасности, которые могут помочь организациям выполнить проверку своих систем на соответствие требованиям безопасности и выявить недостатки в их защите.

Процедуры оценки соответствия требованиям безопасности являются важным аспектом обеспечения безопасности информационных систем в организациях. Руководящие документы, такие как стандарты информационной безопасности и регуляторные требования, предлагают методы и процедуры оценки, которые позволяют проверить соответствие информационных систем требованиям безопасности [3, 4].

Данные процедуры оценки соответствия требованиям безопасности программного обеспечения и информационных систем могут быть разделены на пять этапов.

Первый этап – определение требований безопасности, включает в себя анализ возможных угроз и рисков для системы, а также разработку необходимых требований безопасности, чтобы обеспечить защиту от этих угроз.

Второй этап – оценка системы, который включает в себя анализ текущей степени безопасности системы. На этом этапе проводятся различные методы оценки, такие как тестирование, аудит и проверка уязвимостей. Результаты этой оценки помогают выявить недостатки в безопасности системы и определить необходимость разработки плана улучшения.

Третий этап – разработка плана улучшения, включает в себя определение шагов и процедур для улучшения безопасности системы. Это может включать в себя установку новых мер безопасности, обновление программного обеспечения, улучшение процедур управления доступом и другие меры.

Четвертый этап – реализация плана улучшения, включает в себя внедрение новых мер безопасности, обновление программного обеспечения и улучшение процедур управления доступом. На этом этапе также могут проводиться обучающие мероприятия для сотрудников, чтобы обеспечить правильное использование новых мер безопасности.

Пятый этап – повторная оценка, включает в себя повторное тестирование, аудит и проверку уязвимостей после внедрения новых мер безопасности. Это позволяет оценить эффективность новых мер безопасности и выявить возможные недостатки.

В целом, данные процедуры оценки соответствия требованиям безопасности являются необходимыми для обеспечения безопасности информационных систем и программного обеспечения. Они помогают выявить недостатки в безопасности системы и разработать меры по их устранению.

Заключение. По результатам исследований методов поиска, идентификации и анализа уязвимостей программного обеспечения можно сделать следующий вывод: с целью принятия мер к улучшению уровня защиты информационных систем и для успешного выявления и идентификации уязвимостей в программных продуктах, необходимо использовать все указанные в статье методы поиска, а также неукоснительно руководствоваться документами, которые содержат требования к оценке соответствия требованиям безопасности программного обеспечения и информационных систем.

Библиографический список

1. Российская Федерация. Законы. Об информации, информационных технологиях и о защите информации: Федеральный закон № 149-ФЗ от 27 июля 2006 г. (ред. от 30.12.2020) // Собрание законодательства РФ. – 2006. – № 31. – ст. 14. – Доступ из справ. – правовой системы КонсультантПлюс. – Текст: электронный.

2. Российская Федерация. Законы. О техническом регулировании: Федеральный закон № 184-ФЗ от 27.12.2002. (ред. от 28.11.2018) // Собрание законодательства РФ. – 2006. – № 28. – ст. 12. – Доступ из справ. – правовой системы КонсультантПлюс. – Текст: электронный.

3. Инструментация – эволюция анализа: [сайт]. – URL: https://xakep.ru/2013/09/11/61232/ (дата обращения: 10.11.2023) – Текст: электронный.

4. Падарян В.А. Программная среда для динамического анализа бинарного кода. / В.А. Падарян, А.И. Гетьман, М.А. Соловьев – Текст: непосредственный //Труды Института системного программирования. – 2009. – Т. 16. – С. 51–72.

УДК 004.056.53

ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ DNS (DOMAIN NAME SYSTEM) – СЕРВЕРОВ И МЕТОДЫ КОНТЕНТНОЙ ФИЛЬТРАЦИИ ТРАФИКА

В.В. Гладнев, М.В. Малый, Д.С. Крысин
Научный руководитель: канд. техн. наук, доц. О.А. Пономарева
Уральский федеральный университет
имени первого Президента России Б. Н. Ельцина,
г. Екатеринбург

В данной статье исследуется тема информационной безопасности DNS-серверов, причины по которым возможны атаки на них и роль контентной фильтрации в обеспечении безопасности пользователей. Были изучены различные методы, которые используются на сегодняшний день для атаки на DNS-сервера. Рассматриваются уязвимости в популярном DNS-сервере BIND, предостерегая о критических угрозах, которые могут подвергнуть сомнению стабильность и безопасность сети. Также, в данной статье рассматриваются сущность и технологии контентной фильтрации, ее задачи и роль в обеспечении безопасности информационного взаимодействия пользователей и организаций. Анализируются различные методы контентной фильтрации и системы безопасного управления контентом.

Ключевые слова: безопасность, домен, контент, сервер, уязвимость, фильтрация

Введение. DNS является одной из основополагающих систем в современном Интернете. Данная технология обеспечивает преобразование человеко-читаемых доменных имен в IP-адреса, что позволяет браузерам и приложениям находить нужные веб-ресурсы и серверы. Однако, это ключевое звено сети стало мишенью для злоумышленников и объектом исследований в области безопасности информационных систем. Уязвимости в DNS-серверах приобретают особенную важность. Отказ или атака на серверы имен может иметь глобальные последствия, отключив целые сегменты интернет-инфраструктуры и приведя к серьезным нарушениям в работе.

С увеличением объема данных, передаваемых по сетям, сетевая безопасность и контроль за информационными потоками становятся важнее, чем когда-либо. Контентная фильтрация, как технология, базирующаяся на анализе информации, является одним из ключевых инструментов обеспечения информационной безопасности.

В связи с этим обеспечение информационной безопасности становится вопросом первостепенной важности. Данный вопрос является максимально актуальным, как для частного сектора, так и для государственного.

Данный проект имеет актуальную практическую ценность в свете увеличения числа кибератак и угроз информационной безопасности. Изучение уязвимостей позволит принять меры к улучшению уровня защиты информационных систем и снизить риски возможных утечек данных, финансовых потерь и репутационных ущербов для организаций и частных лиц.

Анализ уязвимостей DNS-сервера, их использование.

Одной из критической уязвимостью считается спуфинг-атака. Спуфинг (от английского слова spoofing) – это кибер-атака, в рамках которой мошенник выдает себя за какой-либо надежный источник, чтобы получить доступ к важным данным или информации. Данная атака опасна тем, что её целью является не сам сервер, а клиенты. Уязвимости такого рода стали доступны, потому что DNS сервер использует предсказуемый номер порта для отправки DNS запросов. Одним из первых, об этой проблеме сообщил Дэн Камински в 2008 году в своем блоге и опубликовал более подробное описание уязвимости которой был присвоен максимальный уровень опасности.

В тестах, которые проводил Дэн Камински, ему удалось отравить кеш сервера имен приблизительно за 5–10 секунд. Эта уязвимость позволяет атакующему перезаписать данные, которые уже находятся в кеше сервера. Сервера имен, которые являются только авторитетными, не подвержены этой уязвимости. Установка высокого значения TTL для хостов на авторитетном сервере не помешает злоумышленнику отравить кеш уязвимых резолверов, так как атака обходит защиту TTL.

Уязвимость затрагивает также и клиентские библиотеки (рабочие станции и сервера, которые обращаются к вышестоящим серверам имен) и может быть проведена против одиночного хоста [1]. Также, некоторые межсетевые экраны с функционалом трансляции адресов, рассчитанные на домашний сектор, используют предсказуемые номера для порта источника запросов, что позволяет злоумышленнику удачно произвести атаку, даже если было установлено исправление на сервер имен или клиент.

Эта атака представляет серьезную опасность, так как в сочетании с другими уязвимостями в программном обеспечении или социальной инженерией она может стать мощным инструментом в руках злоумышленника.

Злоумышленник может:

- произвести фишинг атаку и получить доступ к важным данным;
- произвести атаку типа «человек посередине» и получить доступ к потенциально важным данным (паролям, номерам кредитных карт и другим данным, которые передаются);

• используя уязвимость в ПО, получить доступ к важным данным и даже скомпрометировать целевую систему (например, из-за недостаточной проверки подлинности сервера при установке обновлений приложения, при перенаправлении пользователя на специально сформированный сайт и т.д.).

Даже один из самых популярных DNS-серверов BIND имеет ряд критических уязвимостей. Три основные уязвимости заслуживают более детального рассмотрения:

• **CVE-2016-1285**: Эта уязвимость связана с обработкой входных данных управляющего канала и позволяет злоумышленникам отключать сервер BIND, отправляя специально сформированные пакеты на адрес, указанный в секции «controls» файла named.conf, или при наличии доступа к машине с работающим сервером.

• **CVE-2016-1286**: Данная уязвимость связана с некорректной обработкой записей DNAME, что может вызвать сбой модулей resolver.c или db.c и, как следствие, отказ в обслуживании пользователей сервера.

• **CVE-2016-2088**: Уязвимость связана с реализацией поддержки DNS cookies и может позволить злоумышленникам вызвать отказ в обслуживании с помощью специально сформированных пакетов.

Для всех перечисленных выше ошибок уже выпущены патчи. В бюллетенях безопасности компании ISC содержатся рекомендация для пользователей уязвимых версий BIND как можно скорее обновить версию сервера на ту, где уязвимости устранены, и которая ближе всего по номеру к используемой в текущий момент. Однако, некоторые серверы продолжают использовать старые версии BIND, по большей степени из-за отсутствия у их администраторов привычки смотреть актуальные уязвимости на специализированных сайтах.

Анализ существующих способов (систем) контентной фильтрации трафика в компьютерных сетях.

Суть контентной фильтрации заключается в анализе потоков информации, передающихся в компьютерных сетях. Она выполняет важную задачу в обеспечении безопасности информационного взаимодействия пользователей, ресурсов и информационных сервисов. Эффективная контентная фильтрация позволяет предотвратить передачу информации, запрещенной законодательством, политикой безопасности организации, а также нарушающей нормы этики и морали.

Существует несколько методов контентной фильтрации, которые можно разделить на четыре основные группы:

• Категоризация – классификация информации по определенным категориям для более точного контроля.

• Списки ресурсов – создание списков разрешенных и запрещенных ресурсов в Интернете.

• Контроль доступа – установление правил доступа к информации.

• Фильтрация данных – анализ содержания данных на предмет соответствия политике безопасности.

Целью контентной фильтрации является не только предотвращение передачи запрещенной информации, но и обеспечение общего уровня информационной безопасности. Это особенно важно для организаций, которые хотят защитить свои ресурсы и данные. Фильтрация информационных потоков может осуществляться на разных уровнях модели OSI [2].

Системы контентной фильтрации известны как системы безопасного управления контентом (Secure Content Management, SCM). Они включают управление веб-контентом, контроль обмена сообщениями, защиту от вирусов и нежелательных приложений, а также другие аспекты безопасности.

Обычно выделяются следующие SCM-подсистемы:

• Employee Internet Management (EIM) – контроль доступа пользователей в сеть Интернет;

• Internet Application Security (IAS) – контроль проникновения нелегального контента в сеть организации;

• E-mail scan (ES) – контроль утечки конфиденциальной информации из сети организации и фильтрация спама;

• Virus scan (VS) – контроль проникновения вирусов.

В России к SCM системам относятся межсетевые экраны. Классификация межсетевых экранов (МЭ) по уровню защищенности от несанкционированного доступа (НСД) к информации на базе перечня показателей защищенности и совокупности описывающих их требований устанавливается руководящими документами ФСТЭК [3].

Любой информационный поток декомпозируется на составные элементы до необходимой уровня вложенности. Исходя из этого, различаются способы фильтрации трафика:

• По способу фильтрации (рис. 1);

• По уровню ЭМВОС (рис. 2);

• По используемому инструментарию (рис. 3);

• По используемому подходу (рис. 4).

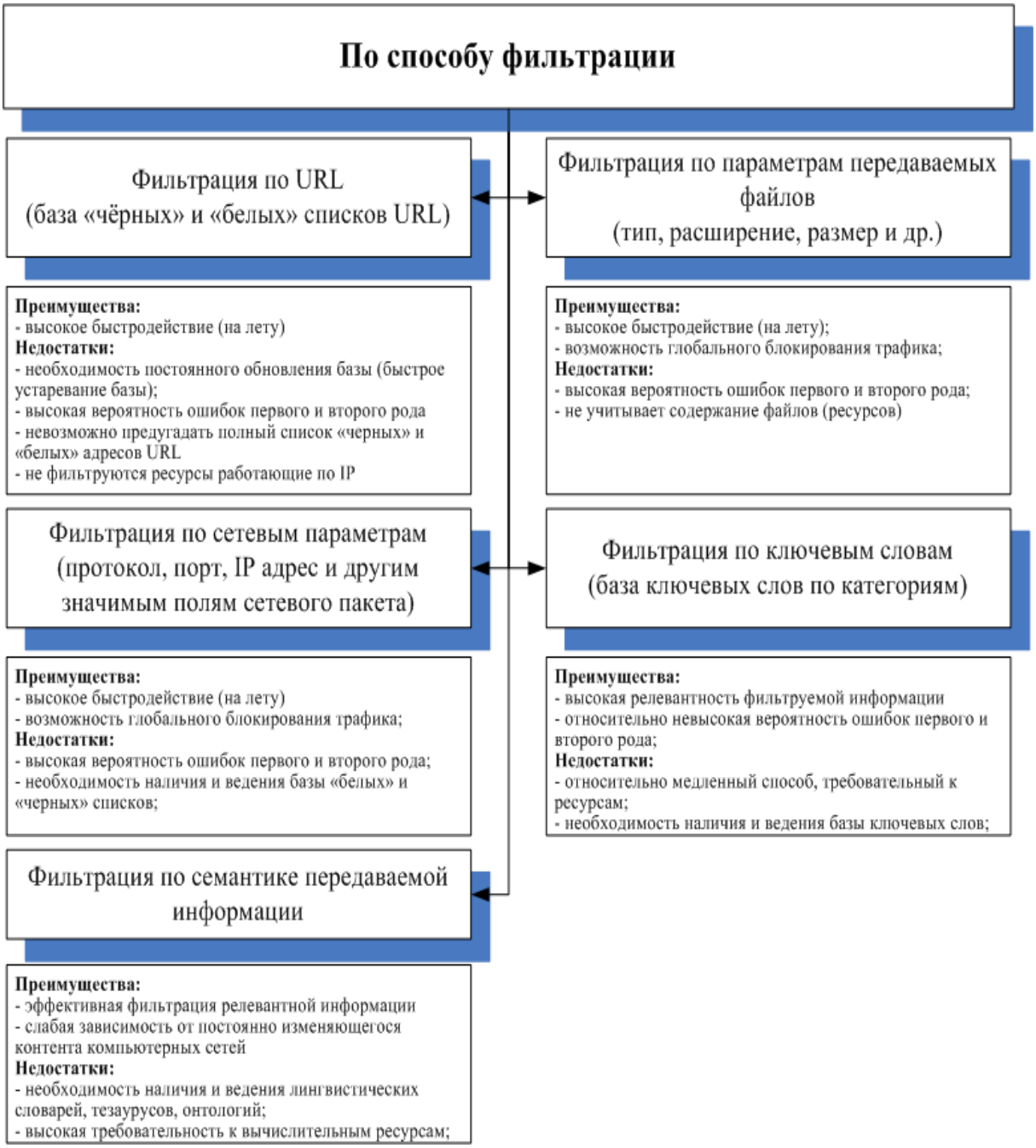


Рис. 1. Классификация по способу фильтрации

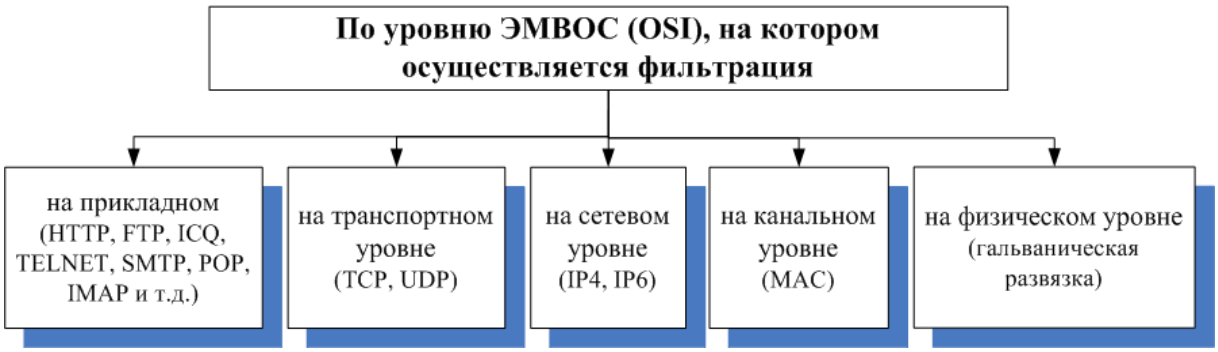


Рис. 2. Классификация методов фильтрации по уровню ЭМВОС

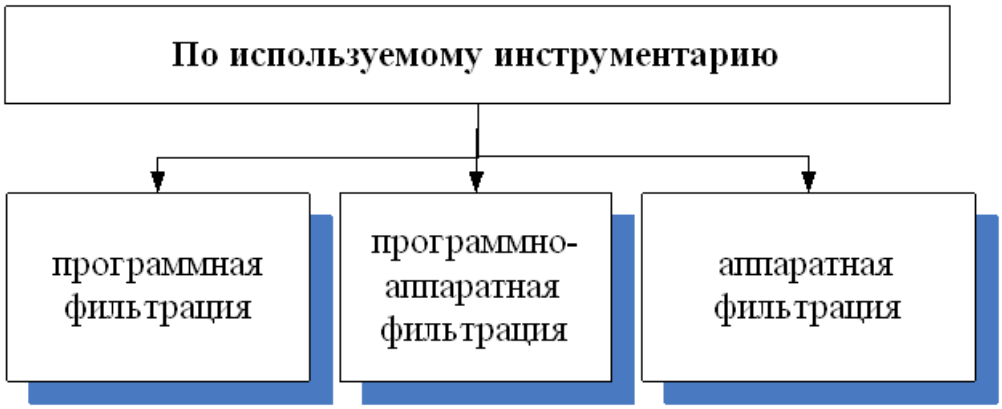


Рис. 3. Классификация методов фильтрации по используемому инструментарию

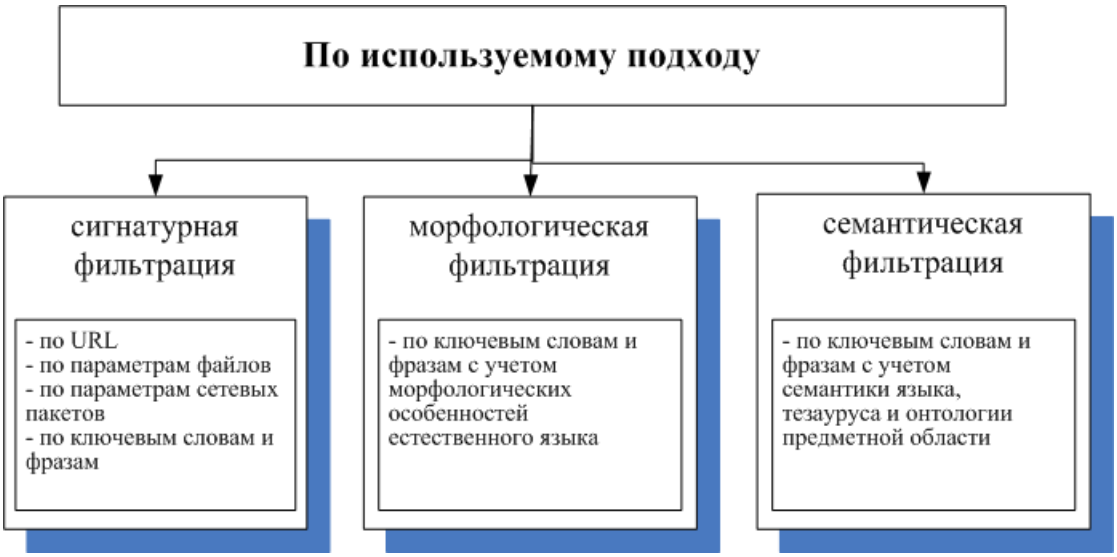


Рис. 4. Классификация методов фильтрации по используемому подходу

Вопросы контентной фильтрации в компьютерных сетях являются актуальными для современных исследований и должны решать важные задачи фильтрации информационных потоков и ресурсов с целью обеспечения информационной безопасности. В современной нормативно-правовой базе уже сейчас существуют нормативные акты, обосновывающие и регламентирующие применения средств фильтрации в государственных и образовательных учреждениях, общественных местах, а также в личном пользовании.

На рынке существует множество программ с функцией родительского контроля, поддерживающих русский язык. Рассмотрим некоторые из них:

- NetPolice – это мощный фильтр, который может блокировать даже доступ на сайты с самым нежелательным контентом, включая ресурсы со взрослым и сексуальным материалом [4].
- K9 Web Protection – в данной программе одним из ключевых элементов является обновление списков сайтов в режиме реального

времени. Как только пользователь пытается зайти на сайт с нежелательным контентом, он добавляется в список блокируемых [5].

• KidGid – программа способна определять наличие ненормативной лексики на веб-страницах, что полезно для родителей, беспокоящихся о языке, используемом на сайтах [6].

• ContentKeeper – данная программа предоставляет тотальный контроль, блокируя доступ к развлекательным ресурсам и предоставляя подробные отчеты о действиях пользователей [7].

Большинство программ контентной фильтрации коммерческие, однако существуют и бесплатные альтернативы. Например, «Интернет Цензор» – это бесплатный интернет-фильтр для детей, разработанный для предотвращения доступа к сайтам, противоречащим законодательству Российской Федерации и ресурсам с деструктивным контентом для лиц моложе 18 лет.

Однако стоит отметить, что контентные фильтры могут использоваться и для цензуры или контроля над действиями других людей. В многих офисах и рабочих средах, они используются, чтобы ограничить доступ сотрудников к определенным ресурсам с целью повышения производительности и предотвращения вирусных атак. Важно помнить, что незаконно следить за действиями других людей в интернете, читать личные сообщения или переписку. Эта информация является конфиденциальной, и ее следует уважать.

Однако, необходимо понимать, что ни одна система не является полностью безопасной, и регулярное обновление и контроль серверов DNS является важным аспектом поддержания безопасности. Также необходимо обучать пользователей о методах защиты и осведомлять их о возможных угрозах с целью предупредить потенциальные атаки.

В целом, понимание уязвимостей DNS-серверов и использование эффективных методов контентной фильтрации являются ключевыми факторами обеспечения безопасности сетевой инфраструктуры и защиты данных. Регулярное обновление и современные технологии помогают минимизировать риски и эффективно бороться с угрозами.

Библиографический список

1. Уязвимость в DNS. Джин на свободе [Электронный ресурс]: сайт / SecurityLab.ru by Positive Technologies – 2008. – Режим доступа: http://www.securitylab.ru/analytics/356362.php. (дата обращения: 16.10.2023).

2. ГОСТ Р ИСО/МЭК 7498-1-99 Информационная технология. Взаимосвязь открытых систем. Базовая эталонная модель. Часть 1. Базовая модель. – Москва. – 62 с. – (Государственный стандарт Российской Федерации).

3. Руководящий документ " Средства вычислительной техники. Межсетевые экраны. Защита от несанкционированного доступа к информации. Показатели

защищенности от несанкционированного доступа к информации" (Гостехкомиссия России, 1997 год).

4. Продукты NetPolice [Электронный ресурс]: сайт / NetPolice.RU – 2008–2016. – Режим доступа: http://www.netpolice.ru/collection/all-items. (дата обращения: 12.10.2023).

5. K9 Web Protection [Электронный ресурс]: сайт / K9 Web Protection – 2010. – Режим доступа: http://www1.k9webprotection.com/. (дата обращения: 20.10.2023).

6. KidGid [Электронный ресурс]: сайт / Интернет-контроль – сайт для умных родителей – 2014. – Режим доступа: http://www.internet-kontrol.ru/polezniy-soft/kidgid.html. (дата обращения: 21.10.2023).

7. Web Filter Pro [Электронный ресурс]: сайт / KONTENTKEEPER Proven Internet Security – 2016. – Режим доступа: https://www.contentkeeper.com/web-filtering. (дата обращения: 22.10.2023).

УДК 004.056

ИССЛЕДОВАНИЕ ВСТРОЕННЫХ СРЕДСТВ ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ В АСУ ТП

В.А. Вишневский
Научный руководитель: канд. техн. наук, доц. А.С. Коллеров
Уральский Федеральный университет
имени первого Президента России Б. Н. Ельцина,
г. Екатеринбург

Проведено исследование встроенных средств обеспечения информационной безопасности в АСУ ТП. Проанализированы методы и механизмы защиты информации, встроенные в современные АСУ ТП. Проведен детальный анализ встроенных средств защиты BIOS, ОС семейства Linux, прикладного ПО SCADA и активного сетевого оборудования.

Ключевые слова: АСУ ТП, встроенные средства защиты информации, информационная безопасность, BIOS, Linux, SCADA.

Автоматизированные системы управления технологическими процессами (АСУ ТП) играют ключевую роль в обеспечении безопасности и эффективности функционирования сложных производственных объектов. Однако, с развитием технологий и увеличением количества информационных потоков, возрастает и риск нарушения информационной безопасности (ИБ) АСУ ТП, что может привести к серьезным последствиям, таким как нарушение работы оборудования, утечка конфиденциальной информации, финансовые потери и даже угроза жизни и здоровью людей.

Целью данной научной статьи является исследование встроенных средств обеспечения ИБ в АСУ ТП. В ходе работы будут проанализированы методы и механизмы защиты информации, встроенные в современные АСУ ТП.

Научная новизна работы заключается в комплексном анализе встроенных средств обеспечения ИБ и определении их влияния на общую защищенность АСУ ТП от внешних и внутренних угроз.

К встроенным средствам обеспечения ИБ относятся механизмы безопасности:

1) Basic Input/Output System (BIOS)/ Unified Extensible Firmware Interface (UEFI);

2) операционных систем (ОС);

3) прикладного программного обеспечения (ПО);

4) активного сетевого оборудования (АСО).

BIOS – набор микропрограмм, позволяющих выполнить начальную проверку работоспособности и настройку комплектующих средств вычислительной техники (СВТ), загрузку ОС и конфигурацию периферийных устройств. BIOS относится к системному ПО и хранится в микросхеме EEPROM на материнской плате СВТ. Дальнейшим развитием BIOS является UEFI – интерфейс между ОС и аппаратной частью СВТ, предназначенный для управления низкоуровневыми функциями комплектующих СВТ. UEFI заменяет прежний интерфейс BIOS и добавляет новые возможности [1].

Механизмы защиты BIOS:

1) доступ к настройкам BIOS по паролю или запрос пароля при включении СВТ;

2) возможность отключения портов периферийных устройств (например, USB, COM, LPT);

3) возможность отключения загрузки со съемных носителей информации (например, переносного жесткого диска, компакт-диска, USB-флешнакопителя);

4) возможность загрузки ОС с определенного жесткого диска.

Механизмы защиты UEFI (в дополнение к механизмам защиты BIOS):

1) режим загрузки «UEFI»;

2) функция Secure Boot Option («Secure boot»), предотвращающая запуск неавторизованных ОС и ПО во время загрузки СВТ и тем самым обеспечивающая защиту от вирусов и других вредоносных программ на этапе загрузки ОС.

На всех СВТ в BIOS/UEFI осуществляется отключение неиспользуемых портов, интерфейсов и отключения загрузки со съемных носителей информации (например, переносного жесткого диска, компакт-диска, USB-флешнакопителя). Включается возможность загрузки ОС только с системного диска.

При поддержке интерфейса UEFI включается функция Secure Boot Option («Secure boot») и осуществляется переразметка жесткого диска с использованием GPT-разделов.

Доступ к параметрам BIOS/UEFI ограничивается путем установки пароля на изменение настроек (пользователь «Administrator»). При наличии технической возможности ограничивается паролем доступ на просмотр настроек BIOS/UEFI (пользователь «User»).

При необходимости дополнительно устанавливается запрос пароля при включении СВТ.

Идентификация и проверка подлинности субъектов доступа при входе в систему ОС семейства Alt Linux осуществляется с помощью набора разделяемых библиотек в составе подключаемых модулей аутентификации Pluggable Authentication Modules (PAM) [2]. Среди требований, предъявляемых к паролям учетных записей пользователей, выделяют следующие: пароль должен содержать не менее 12 символов, среди которых должны быть буквы и цифры.

Функциональный комплекс управления доступом реализован посредством применения механизмов идентификации, аутентификации пользователей при попытке доступа к ОС и различных механизмов разграничения доступа к ресурсам системы:

1) традиционный механизм разграничения доступа в Unix-системах;

2) механизм ролевого доступа (Role-based Access Control).

Традиционный механизм разграничения доступа заключается в следующем: пользователь на уровне ОС имеет уникальный идентификатор пользователя (User Identifier, UID) и принадлежит к одной или нескольким группам, которые обладают уникальным групповым идентификатором (Group Identifier, GID). Принадлежность к каждой из групп наделяет пользователя определенными полномочиями. Полномочия представляют собой набор действий, которые пользователь может выполнять над определенными объектами. Объекты управления на уровне ОС могут быть сгруппированы. Каждая группа в данном случае так же, как и объекты управления, имеет уникальное имя. При входе в ОС пользователь проходит процесс аутентификации. Далее в зависимости от принадлежности к группам пользователь наделяется полномочиями. В зависимости от полномочий пользователь управляет объектами, группами объектов и задачами, которые могут быть выполнены над одним объектом или группой объектов.

Традиционный механизм Unix-систем позволяет задавать права доступа к объекту в зависимости от его отношения к следующим классам:

1) владелец объекта;

2) член группы владельца объекта;

3) остальные пользователи (не входящие впервые 2 группы).

Ролевая модель доступа позволяет пользователю выборочно выполнять задачи суперпользователя. При использовании ролевой модели права до-

ступа на объекты системы группируются с учетом специфики их применения, образуя роли. Смысл ролей в том, что любому пользователю может быть назначена роль. Право выполнения определенных функций привилегии не назначается пользователю непосредственно, а приобретается им только через роль, и управление индивидуальными правами пользователя сводится к назначению ему ролей.

Функциональный комплекс регистрации и учета реализован с помощью следующих сервисов и служб:

1) встроенная служба отправки сообщений Syslog;

2) встроенная служба аудита уровня ядра ОС (audit).

Встроенная служба отправки сообщений Syslog реализует следующий функционал:

1) прием сообщений по протоколу Syslog от локальных процессов;

2) прием сообщений Syslog по сети;

3) отправка полученных сообщений по сети на Syslog-сервер.

Средствами Syslog может производиться регистрация следующих типов событий:

1) успешная авторизация;

2) неуспешные попытки авторизации;

3) сообщения об ошибках, уведомления, отладочная информация для уровней пользователя, системных служб и ядра ОС.

Встроенная служба аудита уровня ядра ОС позволяет регистрировать следующие типы событий:

1) вход/выход из системы;

2) чтение/запись файлов;

3) чтение/изменение атрибутов файлов;

4) создание/удаление файлов;

5) сетевая активность служб и процессов;

6) взаимодействие между процессами;

7) запуск/остановку служб и процессов;

8) административные действия;

9) администрирование учетных записей;

10) работа встроенной криптографической подсистемы ОС.

Встроенная служба аудита ОС регистрирует события в файлах бинарного формата. Для анализа получаемых файлов используется встроенное средство praudit. Для отправки событий на встроенный Syslog-сервер ОС в текстовом формате используется дополнительно подключаемый встроенный модуль audit_syslog.so.

Контроль целостности программных компонент подсистемы защиты информации и компонентов системного и прикладного ПО, а также контроль изменений конфигураций СВТ реализуется с помощью встроенного модуля Alterator-osec.

Резервное копирование в ОС Alt Linux реализуется следующими методами:

1) при помощи команды «ufsdump» можно выполнить полное или инкрементное резервное копирование. Эта процедура создает резервную копию зоны /export/my-zone в /backup/my-zone.ufsdump, где my-zone – имя зоны в системе;

2) создание снимка UFS при помощи fssnap. В этой процедуре используется команда «fssnap», которая создает временный образ файловой системы, для которой будут создаваться резервные копии. Этот метод может использоваться только для последовательного резервного копирования файлов зоны и только на работающих зонах. Однако также рекомендуется на время создания снимка приостановить или зафиксировать в контрольной точке работу активных приложений, которые обновляют файлы.

Резервное копирование при помощи команд «find» и «cpio» выполняется в режиме суперпользователя или роли главного администратора (Primary Administrator).

Системы SCADA (Supervisory Control And Data Acquisition), как прикладное ПО, играют ключевую роль в АСУ ТП. Они выполняют следующие функции:

1) сбор данных: SCADA-системы собирают данные с конечных устройств и датчиков, определяющих такие параметры системы, как температура, давление, уровень, расход и т.д. Эти данные передаются на сервер или автоматизированное рабочее место для дальнейшего анализа;

2) визуализация: SCADA-системы отображают данные в интуитивно понятном виде, чтобы операторы могли контролировать процесс и принимать соответствующие решения. Визуализация включает в себя отображение графиков, диаграмм, таблиц и других видов информации;

3) управление: SCADA-система позволяет операторам управлять технологическим процессом с помощью различных устройств, таких как клапаны, насосы, двигатели и т. д. Команды на выполнение операций могут быть переданы через SCADA-систему;

4) архивирование данных: Большинство SCADA-систем позволяют сохранять данные в архив для последующего анализа и составления отчетности. Это может помочь выявить проблемы в процессе производства и принять меры по их устранению;

5) безопасность: SCADA-система обеспечивает безопасность процесса, так как она позволяет управлять доступом к АСУ ТП, предотвращая несанкционированный доступ и изменения в настройках.

К SCADA-системам могут предъявляться следующие требования:

1) в системе SCADA, на базе которой строится/построена АСУ ТП, должны быть задействованы ее штатные средства идентификации/аутентификации субъектов доступа, которые реализуются посредством многоступенчатого контроля и управления доступом пользователей

к системе SCADA, компонентам (приложениям) системы SCADA, технологическим объектам;

2) проверка прав доступа пользователя должна осуществляться при каждом обращении к системе SCADA, компонентам (приложениям) системы SCADA, технологическим объектам;

3) в системе SCADA должны быть обеспечены регистрация и учет таких событий, как вход/выход пользователя в систему/из системы, попытки доступа пользователя к компонентам (приложениям) системы SCADA, технологическим объектам, попытки изменения прав доступа к компонентам (приложениям) системы SCADA, технологическим объектам.

Для выполнения заданных требований осуществляется настройка следующих функций прикладного ПО:

1) парольная защита информации;

2) разграничение доступа пользователей;

3) регистрация и учет действий пользователей и администраторов;

4) настройка механизмов контроля целостности.

Требования к паролю должны удовлетворять действующей парольной политике АСУ ТП.

Для защищенного сетевого взаимодействия в АСУ ТП осуществляется настройка следующих функций АСО (в частности коммутаторов Eltex серии MES):

1) управление разграничением доступа сетевых администраторов к конфигурированию АСО (заведение учетных записей);

2) управление правами доступа к диагностическим и конфигурационным портам АСО (назначение минимально необходимых привилегий учетным записям администраторов);

3) управление портами и протоколами удаленного доступа;

4) разделение сетей информационных систем на сегменты;

5) логирование событий;

6) резервное копирование и восстановление конфигурации;

7) администрирование АСО.

Для разграничения доступа к конфигурированию АСО создаются учетные записи администратора и пользователя. Пароли учетных записей должны соответствовать требованиям локальных нормативных актов и регламентов ИБ Эксплуатирующей организации. Имя учетной записи и пароль вводятся при входе в систему во время проведения сеансов администрирования АСО.

Управление правами доступа к диагностическим и конфигурационным портам и функциям АСО осуществляется присвоением уровня привилегий учетным записям: от 1 до 15. Уровень привилегий 1 разрешает только доступ к устройству, настройка запрещена. Уровень привилегий 15 разрешает доступ и настройку устройства.

Удаленный доступ к АСО должен осуществляться с использованием безопасных протоколов SSH и HTTPS (для доступа к веб-интерфейсу). Протоколы Telnet, HTTP (для доступа к веб-интерфейсу) и неиспользуемые порты должны быть отключены.

Разделение сетей информационных систем на сегменты производится посредствам конфигурирования VLAN на АСО.

Функция логирования событий производится автоматически. Доступен вывод введенных команд администратором.

Резервное копирование и восстановление конфигурации осуществляется встроенными средствами АСО. Резервное копирование конфигурации АСО необходимо производить только при внесении каких-либо изменений в настройки: до и после внесения изменений, убедившись в корректной работе АСО. Резервная копия перемещается на сервер резервного копирования с использованием безопасного протокола передачи данных Secure Copy Protocol.

Администрирование АСО доступно как локально, так и удаленно, с использованием безопасных протоколов удаленного доступа, например SSH.

Заключение. В данной научной статье проанализированы встроенные средства обеспечения ИБ АСУ ТП общесистемного, прикладного ПО и АСО.

Исследование показало, что СВТ и АСО АСУ ТП имеют встроенные механизмы защиты, такие как парольная защита, отключение портов, отключение загрузки со съемных носителей информации, идентификация и аутентификация пользователей, разграничение доступа, регистрация и учет, контроль целостности и резервное копирование. Однако, для обеспечения максимальной безопасности, необходимо комплексное использование всех доступных средств защиты, а также регулярное обновление ПО и обучение персонала.

Обеспечение ИБ АСУ ТП требует особого внимания, так как сбои в работе таких систем могут привести к серьезным последствиям для производства и окружающей среды. Поэтому необходимо продолжать исследования в этой области, разрабатывать новые методы и технологии защиты информации, а также проводить обучение специалистов ИБ.

Библиографический список

1. Коломойцев В.С. Задачи и средства обеспечения безопасности информационных систем в условиях цифровой экономики / Технико-технологические проблемы сервиса. – 2017. – №. 4 (42). – С. 50–55.

2. Окороков В.А. Защищенные операционные системы //Вестник УрФО. Безопасность в информационной сфере. – 2015. – №. 1 (15). – С. 33–37.

3. Баринов А.Е., Скурлаев С.В., Соколов А.Н. Методика оценки рисков, вызванных уязвимостями в программном обеспечении автоматизированных систем

управления технологическими процессами //Вестник УрФО. Безопасность в информационной сфере. – 2017. – №. 3 (25). – С. 34–42.

4. Баранкова И.И. и др. Подход к проектированию сети предприятия в защищенном исполнении //Вестник УрФО. Безопасность в информационной сфере. – 2018. – №. 1 (27). – С. 24–28.

УДК 004.056.2

ОБЕСПЕЧЕНИЕ ЦЕЛОСТНОСТИ И ДОСТУПНОСТИ ИНФОРМАЦИИ В МОДЕЛИ ДЕЦЕНТРАЛИЗОВАННОЙ СИСТЕМЫ ХРАНЕНИЯ ДАННЫХ

А.А. Повышев, А.Н. Соколов
Научный руководитель: канд. техн. наук, доц. А.Н. Соколов
Южно-Уральский государственный университет,
г. Челябинск

Представлена модель системы хранения данных, функционирующей на основе алгоритма децентрализованного управления. Нейтрализация угрозы безопасности информации осуществляется за счёт территориального распределения и многократного дублирования данных, что предотвращает потерю и уничтожение информации вследствие кибератак, природных и техногенных угроз. Алгоритм управления децентрализованной системой хранения основан на автоматизированном формировании специальных мастерхостов, организующих реестр сетевых связей между другими хостами. Такие мастерхосты являются опорными точками сети хранения данных, они рассчитывают рейтинг доступности хостов на основе данных о сетевой латентности, продолжительности работы хоста в сети и его производительности. Применение стратегий территориального распределения и избыточного кодирования позволяет использовать предложенную модель для обеспечения целостности и доступности хранимых данных в сравнении с традиционной централизованной моделью хранения.

Ключевые слова: децентрализация, доступность, избыточное кодирование, резервирование информации, система хранения данных.

Альтернативой централизованным системам хранения данных стали децентрализованные системы на основе технологий шифрования и блокчейн: Storj [1], Filecoin [2] и SIA [3]. Принцип их работы основан на использование дискового пространства пользователей, за что пользователи получают вознаграждение в цифровых финансовых активах [4]. Эти системы имеют фиксированную юрисдикцию и центральные узлы управления, поэтому они

не являются в полной мере децентрализованными. За счёт территориального распределения и отсутствия конкретной юрисдикции использование децентрализованной системы хранения позволяет устранить угрозы безопасности информации, имеющие природное, техногенное, антропогенное и стратегическое происхождение [5]. Классический подход к нейтрализации таких угроз предусматривает использование комплекса инженерно-технических и программно-аппаратных мер по защите информации: резервирование центров обработки данных, источников электропитания, размещение их в сейсмически безопасных районах, организация складского резерва оборудования, наличие стабильных цепочек поставок запасных частей и т.д. Такие меры требуют существенных капиталовложений и трудозатрат, использование же децентрализованной системы хранения данных позволяет отказаться от большинства этих мер, сэкономив бюджет и снизив трудозатраты. Реализовать такое хранение позволяет новый алгоритм управления децентрализованной системой хранения данных, основанный на самоорганизации недоверенных хостов путём выстраивания иерархической структуры с учётом их динамического рейтинга. Такая самоорганизованная сеть предусматривает возможность создания прямых сетевых связей между хостами с требуемым уровнем сервиса. Прямые связи, как правило, предусматривают построение одноранговой сети, примером является классическая офисная сеть. Адреса накапливаются в специальных таблицах, и хранятся на всех устройствах при первом обращении к ним, так строится ARP-таблица в сети Ethernet [6].

В децентрализованной системе хранения данных такой подход приведёт к быстрому росту таблицы IP-адресов, ведь количество пользователей Интернет составляет более 5 миллиардов [7]. Объём таблицы может соответствовать количеству пользователей сети Интернет, что равно 20 ГБ. Управление такой таблицей требует существенных вычислительных ресурсов, что исключает функционирование системы на персональном компьютере среднестатистического пользователя. Для исключения роста таблицы IP адресов предлагается использовать оптимизацию прямых связей и динамический расчёт рейтинга, позволяющий определить высокорейтинговые хосты в качестве хранителей этих связей, являющихся опорными точками сети или мастерхостами.

Расчёт рейтинга хоста (R) осуществляется в два этапа на основании трёх технических показателей конкретного хоста:

1. Сетевая латентность (L, миллисекунды).

2. Продолжительность нахождения в сети (далее – аптайм хоста) (U, дни).

3. Производительность центрального процессорного устройства (далее – ЦПУ) (P, миллисекунды), определённая на основе продолжительности расчёта эталонного задания.

Рейтинг хоста на первом этапе осуществляется на основании показателей L и U:

$$R_1 = \frac{U}{L \times 0{,}1}.$$

При достаточном рейтинге $R_1 > 2$, алгоритм переходит на второй этап, используя показатели L, U и P, т.е. учитывается производительность ЦПУ:

$$R_2 = \frac{U}{L \times 0{,}1} + \frac{3000}{P - L}.$$

Расчёт рейтинга осуществляется ежечасно в отношении всех хостов, о которых знает существующий хост. Как видим из формулы, наиболее значимым для определения рейтинга является показатель U – аптайм хоста. Для рядового хоста объём базы известных хостов ограничен 1024.

Хосты рассчитывают рейтинг друг друга, в результате хостам с наиболее высоким рейтингом направляется предложение стать мастерхостом первого уровня. Логика работы рядового хоста основывается на стремлении стать мастерхостом, поэтому такие предложения должны приниматься во внимание любым рядовым хостом. Если количество предложений составляет более половины (>1024) от максимального количества хранимых IP-адресов хостов (2048), хост становится мастерхостом первого уровня. Один хост может знать не более чем о 32 мастерхостах первого уровня. Мастерхост первого уровня обладает следующим функционалом: аккумулирует IP-адреса, хранящиеся на хостах, предоставляет доступ к своей базе данных хостов, а также поддерживает базу данных хостов в актуальном состоянии, рассчитывая рейтинг каждого хоста. Максимальный размер таблицы мастерхоста первого уровня не может превышать 8192 IP адресов. Такая таблица сортируется за 7 970мс. на лабораторной машине, что приемлемо с учётом выбора хостами наиболее производительных мастерхостов.

Мастерхост второго уровня забирает IP-адреса, хранящиеся на мастерхосте первого уровня, и формирует из них группы по 2-м первым октетам IP адреса. После этого мастерхост второго уровня размещает сформированные группы в мастерхостах первого уровня, создавая таким образом ориентированность части мастерхостов на конкретную группу IP-адресов. Далее мастерхост второго уровня рассылает известным мастерхостам первого уровня информацию об их специализации на конкретном диапазоне IP-адресов. Принципиальная схема созданной таким образом сети представлена на рис. 1.

Рис. 1. Принципиальная схема организации управления децентрализованной системы хранения данных

Для оценки нагрузки сети определим общий объём служебного трафика V для одного хоста. Пусть расчёт рейтинга (R) осуществляется циклически в отношении каждого хоста, IP-адрес которого храниться в базе данных один раз в час. Определим объём переданных данных для расчёта L. Расчёт L осуществляется отправкой 32 байт случайных данных, в ответ приходит 32 байта данных с задержкой L. Для проверки производительности ЦПУ зададим ему задачу - циклический расчёт хэш-суммы от случайной последовательности размером 64 байта. Цикл состоит из 10 000 итераций. Отправим задание удалённому хосту (64 байта)). От удалённого хоста получаем строку, соответствующую эталонной. Это позволяет хосту h рассчитать рейтинг хоста h_1, затратив объём трафика с учётом накладных расходов в размере 50%: $V(h_1) = \frac{\mathrm{V(L)+V(P)}}{100} \times 50 + \mathrm{V(L)} + \mathrm{V(P)} = \frac{32+32+64+64}{100} \times 50 + 32 + 32 + 64 + 64 = 288.$

Следовательно, часовая загрузка хоста с заполненной базой данных составляет:

$$V(h_{2048}) = \sum_{k=1}^{2048} \frac{\mathrm{V(L)} + \mathrm{V(P)}}{100} \times 50 + \mathrm{V(L)} + \mathrm{V(P)} = 589\,824 = 576\text{КБ}$$

Приведённый в примере объём трафика очень низкий. Это позволяет пренебречь им в расчёте совокупной производительности сети хранения данных. Для обеспечения надёжного хранения информация многократно дублируется с требуемым качеством сервиса и избыточное кодирование. В связи с низким доверием пользователя к хосту в отношении продолжительности хранения данных или нахождения его онлайн дублирование данных совместно с использованием функционала избыточного кодирования, что повышает вероятность доступности файла. В качестве алгоритма кодирования используется кодирование Хэмминга [8].

Классические системы хранения данных имеют несомненные преимущества в производительности перед децентрализованными системами, од-

нако при реализации рисков информационной безопасности, имеющих природное, техногенное или стратегическое происхождение доступность данных может быть существенно снижена [5]. Военные действия или разрушения, вызванные землетрясениями, могут повлиять не только на доступность информации, но и её целостность. Для оценки предложенного подхода предлагаем сравнить две принципиальные схемы построения системы распределённого хранения данных: централизованную схему, предусматривающую центр управления и 100 распределённых копий данных, а также предложенную децентрализованную схему хранения. В качестве критериев сравнения возьмём объём затраченного дискового пространства и вероятность доступности информации.

Пусть каждый день из сети хранения навсегда отключается каждый десятый хост и появляется новый хост. В течение суток недостающая часть данных восстанавливается на новом хосте. Таким образом, каждый хост может работать с вероятностью $P_{work} = 0.9$ и не работать с вероятностью $P_{dwork} = 0.1$. Для собственника информации приемлемая вероятность доступности данных P в информационной системе составляет 0,99671, что соответствует стандарту надёжности дата-центров TIA-942 [9] по классу «Tier I» и составляет около 29 часов простоя в год.

В первой схеме (рисунок 2), информация, объёмом x хранится на n хостах, $n = 100$, при этом каждый хост хранит 100% информации. Отсюда количество затраченного дискового пространства для хранения одной единицы данных V составляет $V = x * n$. Так как для получения данных необходим любой активный хост вероятность доступности данных $P(1) = 1$.

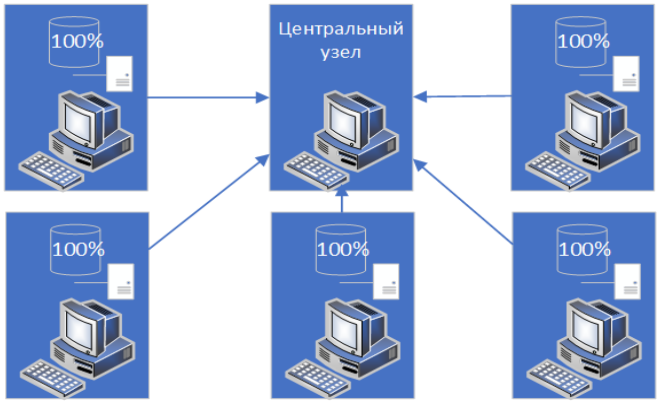


Рис. 2. Схема хранения с центральным узлом

Вместе с тем центральный узел может отключаться от сети в среднем два раза в год на один день. При отключении центрального узла данные становятся недоступными. С учётом этого факта вероятность доступности данных составляет: $P(1) = \frac{363}{365} = 0{,}99452$.

Расчёт показывает, что приемлемая вероятность доступности данных не достигнута. Для достижения требуемого показателя необходимо повысить уровень надёжность центра управления или дублировать его.

Во второй схеме (рис. 3) информация хранится независимо от центральных узлов при этом реже дублирует друг друга. Хранение осуществляется на n хостах (n = 100), при этом каждый хост хранит только 20% от информации, объёмом x.

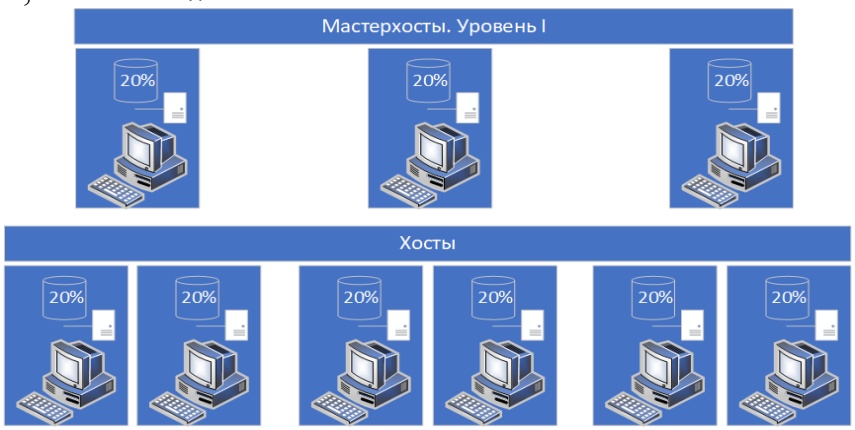


Рис. 3. Схема хранения без центрального узла

Следовательно, при отключении 20 хостов одновременно, хранящих одну и ту же часть информации доступность данных будет нарушена.

Рассчитаем вероятность доступности $P(2)$ через расчёт вероятность отключение любых 20 хостов одновременно в течение суток $P_{20,100}$. Для этого будем использовать предельную теорему Пуассона [10]:

$$P_{m,n} \approx \frac{\lambda^m e^{-\lambda}}{m!} = P_m(\lambda)\ , \lambda = np,$$

где n – общее количество хостов,
p – вероятность отключения одного хоста, в течение суток,
m – количество отключенных хостов.

$$\lambda = np = 100 \times 0{,}1 = 10,$$

$$P_{20,100} \approx \frac{10^{20}2{,}7182818284^{-10}}{20!} = 0{,}00186608$$

Теперь оценим вероятность одновременного отключения именно тех 20 хостов, на которых храниться одинаковая, и не повторяющаяся на других хостах информация. Так как информация разделена на 5 частей, количество вариантов одновременного отключения 20 произвольных хостов N_{20} составляет $N_{20} = 20^5 = 3\ 200\ 000$, отсюда вероятность отключения 20 хостов с одинаковой частью хранимых данных P_{20} составляет:

$$P_{20} = \frac{1}{3\ 200\ 000} = 0{,}0000003125$$

Умножая вероятность рассмотренных событий $P_{20,100}$ и P_{20} получим искомую вероятность доступности данных $P(2)$:

$$P(2) \approx 1 - (P_{20,100} \times P_{20}) \approx \ 0{,}99999999941685,$$

$$0{,}99999999941685 \ > 0{,}99452 > 0.99671,$$

$$P(2) \ > P(1) > P.$$

Таким образом, расчётная вероятность доступности информации в децентрализованной системе хранения $P(2)$ выше вероятности доступности в централизованной системе хранения и выше вероятности доступности требуемой собственником хранимой информации. Кроме того, благодаря экономии на дублировании данных информационная система дополнительно получает 60% дискового пространства.

Применение предложенного алгоритма управления с применением стратегий территориального распределения, избыточного кодирования даёт возможность организовать самоуправляемую информационную систему с большим количеством хостов за счёт распределённого хранения информации об их адресах и выделения наиболее стабильных и производительных хостов в качестве основных хранителей сетевой информации. Благодаря территориальному распределению и отсутствию центральных точек управления построенная на основе этого алгоритма децентрализованная сеть хранения данных может быть использована для нейтрализации угрозы безопасности информации, имеющих природное, техногенное, антропогенное и стратегическое происхождение.

Библиографический список

1. S. Wilkinson, T. Boshevski, J.h Brandoff, J. Prestwich, G. Hall, P. Gerbes, P. Hutchins, C. Pollard. Storj. A Peer-to-Peer Cloud Storage Network // Storj. Official Website – (https://www.storj.io/storjv2.pdf).

2. B. Fisch, J. Bonneau, N. Greco, and J. Benet. Scaling Proof-of-Replication for Filecoin Mining // Semantic Scholar – (https://www.semanticscholar.org/paper/Scaling-Proof-of-Replication-for-Filecoin-Mining-Fisch-Bonneau/e39ee204c52d98b8f06fc873f9c6a472a65a5145).

3. D. Vorick., L. Champine. Sia: Simple Decentralized Storage // Nebulous, Inc. Protocol Labs – (https://sia.tech/sia.pdf).

4. О цифровых финансовых активах, цифровой валюте и о внесении изменений в отдельные законодательные акты Российской Федерации [Текст]: Федер. закон от 31 июля 2020 г. № 259-ФЗ – (http://www.kremlin.ru/acts/bank/45766)

5. Повышев А.А., Соколов А.Н., Мищенко Е.Ю. Универсальная классификация угроз безопасности информации и её применение для разработки модели угроз и оценки рисков //Вестник УрФО. Безопасность в информационной сфере. – 2023. – Т. 3. – №. 49. – С. 68–80.

6. "ISO/IEC/IEEE International Standard for Ethernet," in *ISO/IEC/IEEE 8802-3:2014(E)*, vol., no., P. 1-3754, 1 April 2014 – (https://ieeexplore.ieee.org/abstract/document/6781545).

7. Digital 2023: Global Overview Report // Kepios – (https://datareportal.com/reports/digital-2023-global-overview-report).

8. Золотарёв В.В., Овечкин Г.В. Повышение надежности передачи и хранения данных с использованием многопороговых методов декодирования помехоустойчивых кодов //Цифровая обработка сигналов. – 2012. – №. 1. – С. 16–21.

9. Jew A. Data Center Telecommunications Cabling and TIA Standards //Data Center Handbook: Plan and Operations of a Smart Data Center. 2021. С. 193–210.

10. Гмурман В.Е. Руководство к решению задач по теории вероятностей и математической статистике // Высшая школа. 2004. С. 37–39.

УДК 004

АНАЛИЗ УГРОЗ КОНФИДЕНЦИАЛЬНОСТИ ИНФОРМАЦИИ: УТЕЧКА ДАННЫХ ЧЕРЕЗ HTML5

Д.Е. Власова
Научный руководитель: канд. техн. наук, доц. А.С. Коллеров
Уральский федеральный университет
имени первого Президента России Б.Н. Ельцина,
г. Екатеринбург

Статья посвящена анализу потенциальных угроз безопасности данных, связанных с использованием технологии, работающих на платформе HTML5. Специалисты по информационной безопасности сталкиваются с вызовами, связанными с возрастанием числа утечек конфиденциальной информации. Исследование сосредотачивается на возможной утечке данных через Guacamole. Проведен анализ возможностей приложения для удаленного доступа, работающего по 80 порту. В ходе исследования выявлены ограничения в возможности категоризации веб-сервера Guacamole, что приводит к увеличению риска утечки информации. Приведены результаты экспериментального исследования. Определены меры митигации рисков, связанных с утечкой информации через браузер, включающие законодательные, организационные и технические меры.

Ключевые слова: guacamole, html5, удаленный доступ, утечка.

В настоящее время одной из главных задач внутри любой компании все чаще становится обеспечение безопасности конфиденциальных данных. Проводятся регулярные работы по усовершенствованию мер безопасности, а также работы по повышению квалификации сотрудников. Это позволяет специалистам по информационно безопасности поддерживать высокий уровень эффективности превентивных мер защиты внутри организации.

На фоне повышения ценности конфиденциальной информации в период пандемии и кибератак прослеживается снижение уровня защищенности цифровых активов и рост умышленных утечек внутреннего характера. Так, доля нарушений внутреннего характера к середине 2022 года выросла до 67% [1].

Одним из наиболее распространённых каналов передачи информации, являются интернет-ресурсы. В качестве защиты от утечек информации через браузер могут использоваться DLP-системы, позволяющие осуществлять мониторинг POST и GET запросов, а также же системы web-контроля, ограничивающие доступ к определенным сайтам или целым категориям сайтов [2].

Описанные меры способны покрыть наиболее популярные категории сайтов, такие как: облачные хранилища, социальные сети, мессенджеры, почтовые сервисы и т.д., однако, помимо вышеперечисленного, браузер открывает безграничный доступ и к другим, некатегоризированным ресурсам.

В данной статье рассматривается возможность утечки данных с использованием Apache Guacamole. Guacamole – это шлюз удаленного рабочего стола, не требующий клиентское программное обеспечение. Программное обеспечение выполняет роль прокси между RDP/VLC и HTML5[3].

Приложение поддерживает стандартные протоколы удалённого администрирования, в том числе VNC, RDP, SSH и telnet.

Тестирования возможности утечки предполагается с использованием виртуального стенда, состоящего из Guacamole Server, клиента Windows, к которому осуществляется подключение по RDP и клиента Windows, находящегося в корпоративной сети (рис. 1).

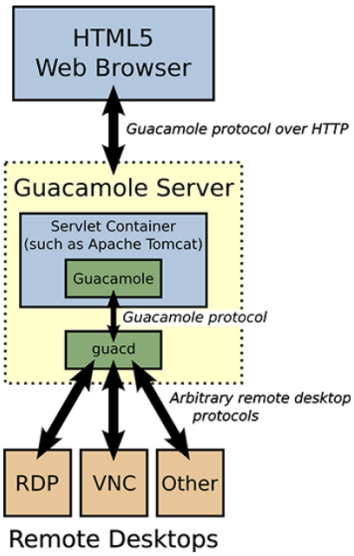


Рис. 1. Архитектура Apache Guacamole

Настройка сервера Guacamole предполагает возможность загрузки и выгрузки файлов с локального хоста на удаленный и наоборот. Для этого в

конфигурационном файле /etc/guacamole/user-mapping.xml прописываются параметры:

<param name="enable-drive">true</param>

<param name="drive-path">Путь до папки</param>

Указанные параметры создадут на сервере Guacamole расшареную папку, которая будет служить посредником между локальным и удаленным хостами (рис. 2).

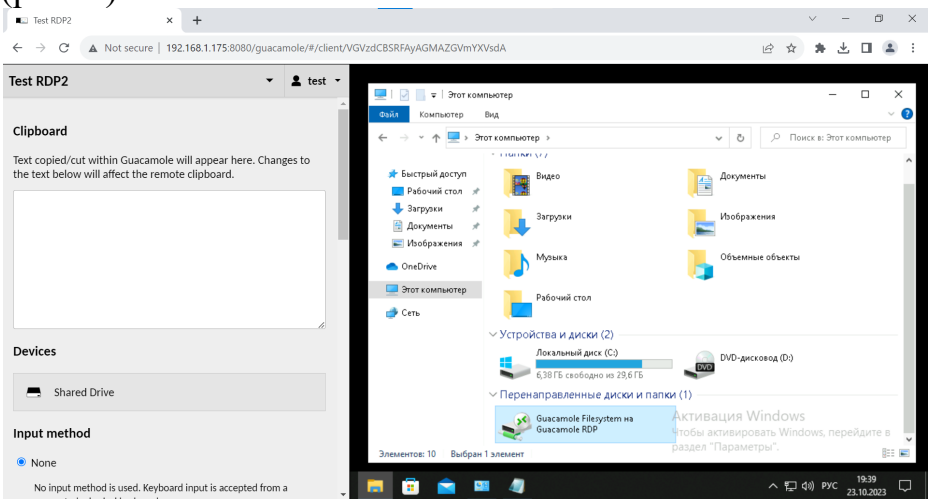


Рис. 2. Интерфейс для обмена файлами

Выполним загрузку на сервер файла vde.docx, содержащего конфиденциальную информацию. С помощью анализатора трафика изучим заголовки запроса (рис. 3).

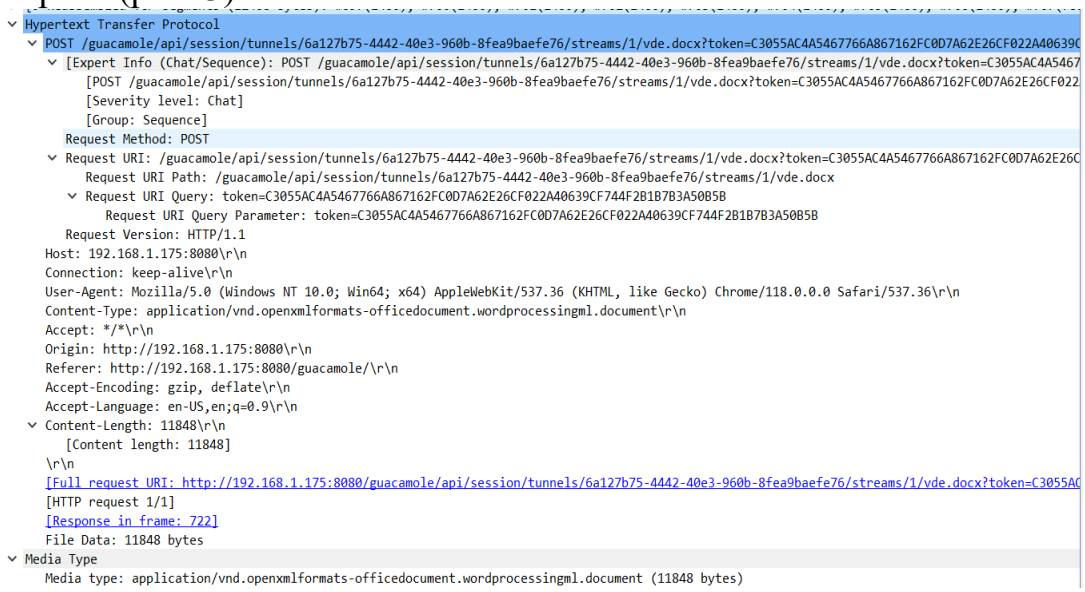


Рис. 3. Заголовки POST-запроса

Заголовок Content-Type: application/vnd.openxmlformats-officedocument.wordprocessingml.document указывает на тип передаваемых данных, в данном случае – документ в формате Office Open XML (docx), что свидетельствует о наличии файла. Content-Length: 11848 указывает на длину (раз-

мер) передаваемых данных. File Data: 11848 bytes явно указывает на наличие данных размером в 11848 байт, что подтверждает наличие файла.

О том, что запрос успешно получен и обработан сервером, свидетельствует ответ сервера с кодом 204 (рис. 4).

Рис. 4. HTTP ответ

С целью обнаружения и предупреждения утечек данных на компьютере, с которого осуществлялась загрузка документа, используются следующие технические средства: агент DLP, web-контроль, NGFW. Ввиду того, что системам безопасность не удалось категорировать веб-сервер Guacamole загрузку данных выявить не удалось.

Отслеживание подобного рода каналов утечки информации возможно с использованием комплексного подхода, включающего:

1. Подробное ведение логов веб-сервера: в конфигурационных файлах веб-сервера необходимо активировать параметр отслеживания HTTP-запросов. Настройка параметра описывается в документации используемого веб-сервера.

2. Использование инструментов для анализа логов в реальном времени.

3. Создание фильтров, которые будут искать в логах HTTP-запросы, содержащие расширения файлов, связанных с документами. Например, для logstash:

```
filter {
  if [message] =~ /\.docx/ {
  }
}
```

4. Настройка правил безопасности. При обнаружении событий, свидетельствующих об отправке файлов проинформировать ответственную группу.

5. Разработка и соблюдение строгих политик безопасности, включая ограничение передачи файлов через веб-интерфейсы, может помочь в предотвращении утечек данных.

6. Регулярное обучение сотрудников информационной безопасности и возможным последствиям, связанным с нарушением внутренних политик компании.

7. В случае, если специфика работы позволяет - ограничение списка доступных интернет – ресурсов.

В ходе исследования была рассмотрена проблема потенциальной утечки конфиденциальных данных через Apache Guacamole. Эта система, предназначенная для обеспечения удаленного доступа, может стать вектором угрозы безопасности данных, особенно при недостаточной настройке и мониторинге.

Проанализированы методы предотвращения утечек данных, включая использование DLP-систем, web-контроля и других технических средств. Однако, выявлены ограничения в возможности категоризации веб-сервера Guacamole, что создает потенциальные точки уязвимости.

Подготовлен список рекомендаций, включающих не только технические, но и законодательные и организационные меры.

Библиографический список

1. Отчёт об исследовании утечек информации ограниченного доступа в I половине 2022 года / Экспертно-аналитический центр InfoWatch. – Москва, 2022. – 33 с.

2. Web Control: удобная и надежная защита от нежелательных сайтов // АО «Лаборатория Касперского». – URL: https://www.kaspersky.ru/blog/web-control-convenient-reliable-protection-from-unwanted-sites/791/ (дата обращения: 22.10.2023).

3. Apache Guacamole™ [Электронный ресурс]. – URL: https://guacamole.apache.org/ (дата обращения: 22.10.2023).

УДК 004.056

АНАЛИЗ СРЕДСТВА ЗАЩИТЫ ИНФОРМАЦИИ SECRET NET STUDIO

А.А. Голынский
Научный руководитель: канд. техн. наук, доц. Т.Ю. Зырянова
Уральский государственный университет путей сообщения, г. Екатеринбург

В статье рассмотрено средство защиты информации Secret Net Studio для операционных систем Windows в сравнении с подобными актуальными СЗИ. Приведен обзор модулей, обеспечивающих защиту системы. Сделан вывод об актуальности и потребности данного средства в организациях.

Ключевые слова: информационная безопасность, информация, средство защиты информации.

Сети и информационные технологии стали неотъемлемой частью современных предприятий. В современном мире, где данные играют решающую роль, обеспечение безопасности информации является важнейшей задачей для каждой организации. В этом контексте возникает необходимость в средствах, специально разработанных для обеспечения высокого уровня конфиденциальности и безопасности, и именно поэтому средства защиты информации (СЗИ), такие как Secret Net Studio, становятся жизненно важными для предприятий. В данной статье будет рассмотрено СЗИ Secret Net Studio, его функционал и произведено сравнение с подобными сертифицированными СЗИ.

С учетом постоянных угроз, связанных с кибератаками, утечками конфиденциальной информации и другими видами атак, предприятия должны стремиться к обеспечению максимальной безопасности своих данных и коммуникаций. Это особенно важно для компаний, работающих в секторах, где конфиденциальность информации имеет ключевое значение, таких как финансовые учреждения, медицинские организации, госучреждения.

Средство защиты информации Secret Net Studio представляет собой интегрированный комплекс, используемый для обеспечения защиты данных и обеспечения безопасности коммуникаций [1, 2].

Основными функциональными возможностями Secret Net Studio являются [3]:

– Шифрование информации: Secret Net Studio предоставляет надежные механизмы шифрования данных, обеспечивая защиту информации от несанкционированного доступа.

– Идентификация и проверка подлинности: система Secret Net Studio позволяет надежно идентифицировать пользователей и устройства, обеспечивая аутентификацию для доступа к ресурсам.

– Управление правами доступа: с помощью Secret Net Studio администраторы имеют возможность настраивать и контролировать доступ к ресурсам и данным, включая установление разрешений и проведение аудита действий пользователей.

– Мониторинг и анализ безопасности: Secret Net Studio предоставляет инструменты для непрерывного мониторинга событий, выявления инцидентов и проведения анализа уровня безопасности.

Система защиты данных Secret Net Studio обеспечивает всеобъемлющую безопасность благодаря интеграции следующих ключевых модулей: [4]

– Модуль дискреционного управления доступом: этот модуль обеспечивает гранулированный контроль над доступом пользователей к локаль-

ным дискам, каталогам и файлам. Он также надежно управляет правами доступа.

– Модуль полномочного управления доступом: обеспечивает разграничение доступа к информации разного уровня конфиденциальности для пользователей.

– Модуль затирания данных: Secret Net Studio обеспечивает гарантированное уничтожение информации, исключая возможность восстановления или повторного использования данных. Это критически важно для сохранения конфиденциальности.

– Модуль управления устройствами: этот механизм позволяет контролировать подключение различных устройств, таких как USB-носители, съемные диски и жесткие диски, обеспечивая высокий уровень безопасности. Кроме того, этот модуль оповещает о любых изменениях в аппаратной конфигурации и может при необходимости блокировать их.

– Изолированная программная среда: этот модуль позволяет определить, какое программное обеспечение имеет разрешение для использования на компьютере. Он следит за файлами, используемыми для запуска программ, библиотеками и сценариями.

– Контроль печати: гарантирует индивидуализацию доступа к принтерам для пользователей, допускает вывод документов определенного уровня секретности и предоставляет возможность проставления специальных маркировок во время печати.

– Защита дисков и шифрование: этот компонент обеспечивает надежную защиту информации от несанкционированного доступа, а также дарует возможность шифровать содержимое файловой системы для дополнительной безопасности.

– Межсетевой экран: гарантирует внимательный контроль за сетевым трафиком на различных уровнях: сетевом, транспортном и прикладном.

– Авторизация сетевых соединений: этот умный механизм обогащает сетевые пакеты специальными служебными данными, обеспечивая тем самым их аутентичность и целостность, а также защищая от потенциальных атак.

– Обнаружение вторжений: гарантирует активное выявление и последующее блокирование как внешних, так и внутренних попыток нарушения безопасности, направленных против защищаемого компьютера.

– Антивирус: позволяет проводить эвристический анализ данных и автоматизированную проверку на наличие вредоносных программ, зарегистрированных в базе. Кроме того, оно способно выполнять сканирование жестких дисков, сетевых папок и внешних носителей информации.

Центр управления Secret Net Studio представлен на рис. 1.

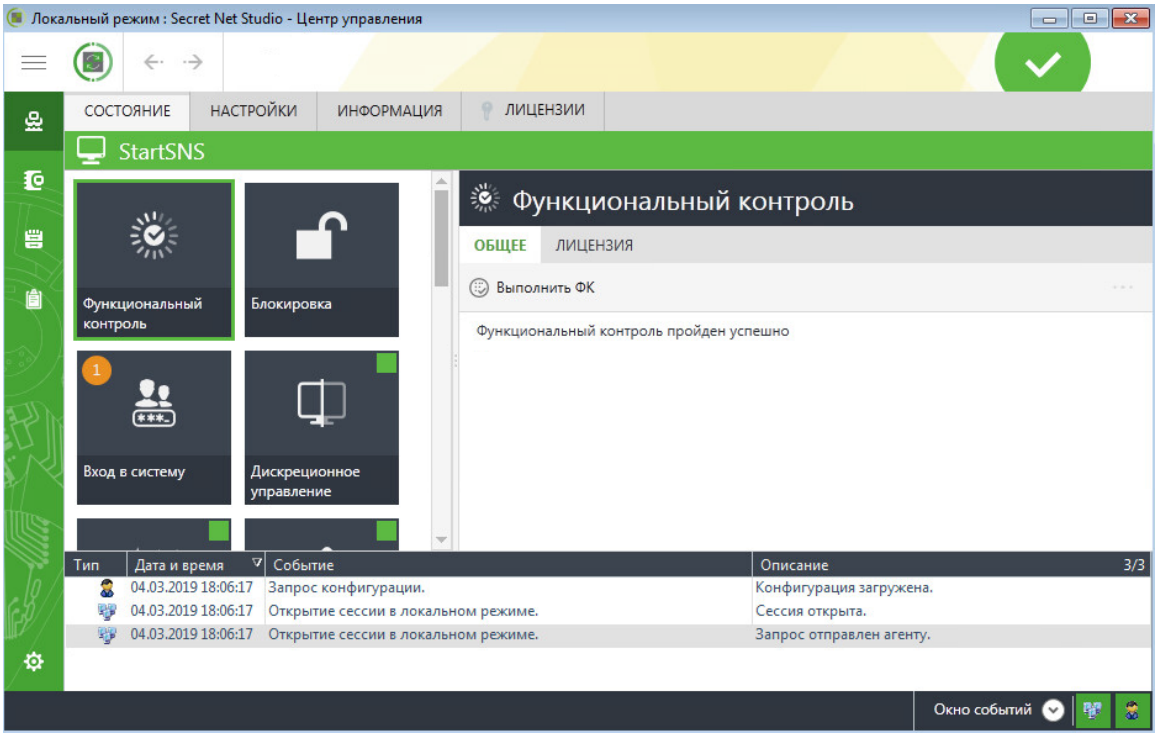


Рис. 1. Центр управления Secret Net Studio

Рассмотрим разницу между СЗИ Secret Net Studio и схожими СЗИ, которые могут быть использованы и развернуты на предприятии. Для сравнения будут рассмотрены СЗИ Dallas Lock от ООО «Конфидент» и Страж NT от ООО «Рубинтех».

Как и СЗИ Secret Net, Dallas Lock и Страж NT предназначены для исключения несанкционированного доступа к компьютерам сотрудников и серверам предприятия.

Таблица 1

Таблица сравнения средств защиты информации

Параметр	Secret Net Studio	Dallas Lock	Страж NT
Сервер безопасности	+	+	–
Клиентская часть	Полный комплекс локальной защиты с консолью	Полный комплекс локальной защиты с консолью	Полный комплекс локальной защиты с консолью
Консоль администратора	Есть в серверной и клиентской части	Есть в серверной и клиентской части	Есть в клиентской части
Сбор данных в локальное хранилище при отсутствии связи с сервером	+	+	–

Окончание табл. 1

Параметр	Secret Net Studio	Dallas Lock	Страж NT
Поддерживаемые ОС	Windows Server 2008 R2 Windows Server 2012/2012R2 Windows Server 2016 Windows Vista Windows 7, 8, 8.1, 10	Windows Server 2008 R2 Windows Server 2012/2012R2 Windows Server 2016 Windows Vista Windows 7, 8, 8.1, 10	Windows Server 2008 R2 Windows Server 2012/2012R2 Windows 7, 8, 8.1, 10
Основные функции СЗИ	+	+	+
Проверка обновления и наличие утилиты	–	+	–
Средство антивирусной защиты	+	–	–
Интеграция компьютеров под управлением ОС Linux	+	+	–
Средство обнаружения вторжений	+	+	–
Возможность разделения сетей на доверенные и недоверенные	–	+	–
Ограничение пользователей, допущенных к печати	+	Dallas Lock 8.0	+
Теневая копия напечатанных документов	+	+	–
Назначение категорий конфиденциальности на устройства	+	Dallas Lock 8.0	+
Контроль запуска задач	+	–	+

Dallas Lock и Страж NT выполняют схожие задачи, выполняя такие функции как: идентификация и аутентификация пользователя, контроль

подключаемых устройств и аппаратной конфигурации компьютеров, а также контроль целостности операционной системы. Для детального анализа различий между этими средствами защиты информации составим таблицу сравнения [5].

В табл. 1 приведены немногие параметры сравнения, но тем не менее видно, что основным конкурентом Secret Net Studio является Dallas Lock. Secret Net Studio и Dallas Lock имеют множество совпадающих функций, но есть и разница. Например, в вопросе мандатного разграничения доступа Secret Net Studio имеет множество возможностей контроля, которые у Dallas Lock учтены только в новых версиях. У Secret Net Studio имеется антивирус, в отличии от конкурентов. Dallas Lock с другой стороны имеет большее количество модулей. В остальном функционал Secret Net Studio и Dallas Lock схож и имеет мало отличий. Оба средства имеют полный контроль подключения внешних устройств, полный или почти контроль печати и дискреционное управление доступом, средство обнаружение вторжений и т.д.

В сравнении с Secret Net Studio, Страж NT проигрывает по многим пунктам. Страж NT не имеет сервера безопасности, при отключении и подключении устройств к автоматизированному рабочему месту, нет возможности блокировки, нет теневого копирования информации с внешних носителей и напечатанной информации, отсутствует возможность контроля доступа к сетевым интерфейсам. Страж NT имеет централизованное управление, как и Secret Net Studio, однако его функционал заметно меньше, а также у Страж NT отсутствует возможность установить дополнительные модули защиты.

СЗИ Secret Net Studio подтверждает, что его комплексные модули обеспечивают высокий уровень безопасности, и значительное удобство использования на предприятии. Secret Net Studio обеспечивает всестороннюю защиту конфиденциальных данных и информации от внешних и внутренних угроз. Его функциональные возможности модулей делают Secret Net Studio полезным компонентом для любого современного предприятия. Совокупность этих функций обеспечивает безопасность и надежность при работе с конфиденциальными данными, устраняя вредоносные программы и предотвращая возможные угрозы.

Библиографический список

1. Secret Net Studio. [Электронный ресурс] URL: https://www.securitycode.ru/products/secret-net-studio/ (дата обращения: 21.10.2023).

2. Secret Net Studio: что это за программа. [Электронный ресурс] URL: https://komrunet.ru/blog/detail/secret-net-studio-chto-eto-za-programma/ (дата обращения: 21.10.2023).

3. Средство защиты информации Secret Net Studio. [Электронный ресурс] URL: https://www.securitycode.ru/upload/iblock/2ce/Руководство%20администратора.%20Принципы%20построения.pdf (дата обращения: 21.10.2023).

4. Обзор Secret Net Studio 8.1. Часть 1 – защитные механизмы. [Электронный ресурс] URL: https://www.anti-malware.ru/reviews/Secret_Net_Studio_part1 (дата обращения: 20.10.2023).

5. Сравнение сертифицированных средств защиты информации от несанкционированного доступа для серверов и рабочих станций [Электронный ресурс] URL: https://www.anti-malware.ru/compare/information-protection-unauthorized-access-fstek-certified (дата обращения: 24.10.2023).

УДК 004.056

АУДИТ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ ЛОКАЛЬНОЙ СЕТИ, ПОСТРОЕННОЙ НА ОСНОВЕ ОПЕРАЦИОННЫХ СИСТЕМ НА БАЗЕ ЯДРА LINUX

Т.Ю. Ряпасов, Т.Ю. Зырянова
Научный руководитель: канд. техн. наук, доц. Т. Ю. Зырянова
Уральский государственный университет путей сообщения,
г. Екатеринбург

В статье приведены результаты аудита информационной безопасности локальной сети, построенной на основе операционных систем на базе ядра Linux, в связи с тем, что они являются распространенными в качестве систем для импортозамещения.

Ключевые слова: аудит, информационная безопасность, операционные системы; Linux.

В современном обществе всё больше растёт популярность операционных систем на базе ядра Linux, поскольку они являются надежными, безопасными и применяемыми в системах с различными задачами [1]. Для проведения аудита были выбраны операционные системы Centos 9 и Astra Linux 1.6, так как они являются распространенными в качестве систем для импортозамещения.

Цель работы провести аудит операционных систем Centos 9 и Astra Linux 1.6.

Для достижения поставленной цели необходимо выполнить следующие задачи:

1. Проанализировать предоставленную локальную сеть;
2. Подготовить инструментарий для проведения аудита информационной безопасности;
3. Провести аудит и проанализировать результаты, дав рекомендации по их устранению.

Локальная сеть состояла из двух серверов баз данных, связанных между собой через маршрутизатор. На первом сервере была установлена ОС Astra Linux 1.6, на втором сервере CentOS 9.

Методом проведения аудита являлось сканирование при помощи системы контроля защищенности и соответствия стандартам RedCheck версии 2.6.9.6679 [2].

Для более комплексного обзора безопасности данных операционных систем было выбрано три метода проверок [3]:

1. Аудит уязвимостей.
2. Аудит в режиме «Пентест».
3. Аудит конфигураций.

Аудит уязвимостей – аудит с точки зрения пользователя с привилегиями (root), то есть в режиме «белого ящика».

Таблица 1

Результаты аудита уязвимостей

Операционная система	Уровни уязвимостей			С чем связаны	Как устранить
	высокий	средний	низкий		
Astra Linux 1.6	2	6	0	Отсутствие точечной настройки операционной системы, в частности уязвимости ядра Linux и ненадежными алгоритмами шифрования	Установкой специализированных патчей системы и обновлением от разработчиков ОС
CentOS 9	1	33	2	Уязвимости, связаны с libtiff, runc, lua, python3-louis. С их помощью злоумышленник может выполнять вредоносный код на устройстве и получать полный доступ к системе	

В результате проверки Astra Linux 1.6 было выявлено восемь уязвимостей, среди которых две высокого уровня, шесть среднего уровня и ноль уязвимостей низкого уровня [4].

Уязвимости связаны с отсутствием точечной настройки операционной системы, в частности уязвимости ядра Linux и ненадежными алгоритмами шифрования.

Проведя аудит уязвимостей CentOS 9, было выявлено тридцать шесть уязвимостей, среди которых одна высокого уровня, тридцать три среднего уровня и две низкого уровня.

Большинство уязвимостей – это уязвимости, связанные с libtiff, runc, lua, python3-louis. С их помощью злоумышленник может выполнять вредоносный код на устройстве и получать полный доступ к системе.

Уязвимости систем, выявленные аудитом уязвимостей, устраняются установкой специализированных патчей системы и обновлением от разработчиков ОС.

Аудит в режиме Пентест – аудит с точки зрения злоумышленника. Проводится в режиме «Черный ящик», когда у проверяющего нет никаких привилегий в системе.

Таблица 2

Результаты аудита в режиме Пентест

Операционная система	Уровни уязвимостей			С чем связаны	Как устранить
	высокий	средний	низкий		
Astra Linux 1.6	4	9	2	Отсутствие точечной настройки операционной системы, в частности OpenSSH	Произвести настройку сервиса OpenSSH
CentOS 9	1	33	2	OpenSSH	

В результате проверки Astra Linux 1.6 было выявлено пятнадцать уязвимостей, среди которых четыре высокого уровня, девять среднего уровня и две уязвимости низкого уровня.

Уязвимости связаны с отсутствием точечной настройки операционной системы, в частности OpenSSH.

Проведя аудит уязвимостей CentOS 9, было выявлено пять уязвимостей, среди которых две высокого уровня, две среднего уровня и одна низкого уровня.

Большинство уязвимостей – это уязвимости, связанные с OpenSSH.

По итогам аудита в режиме пентест было выявлено, что девяносто процентов уязвимостей связаны с удаленным подключением злоумышленника к устройствам и получением прав суперпользователя для выполнения вредоносного кода.

Третьим этапом и основным в данной работы было проведения аудита конфигураций.

Аудит конфигураций – это аудит, который позволяет обнаружить несоответствие системы эталонным требованиям ФСТЭК России, общим настройкам безопасности АЛТЭКС-СОФТ, ГОСТ Р 57580-2017, общим настройкам безопасности Debian (Astra Linux 1.6) или RedHat (CentOS 9).

Проведение аудита конфигураций Astra Linux 1.6 выявило шестьдесят несоответствий эталонным требованиям, в их число вошли:

1. Несоответствия по регистрации событий системы и контроль действий системы.

2. Запрет множественной аутентификации.

3. Реализация и контроль информационного взаимодействия между сегментами контуров безопасности.

4. Контроль формирования данных регистрации о событиях защиты информации объектов информатизации.

Одни из действий, предназначенных для устранения несоответствий являются: приведение sources. list к эталонному образцу, точечная настройка конфигов в /etc, ограничение прав на использование crontab, включение hardened, отключение параметра ядра для отправки перенаправлений ICMP для всех интерфейсов, включение аудита процессов, которые запускаются раньше, чем служба аудита.

Проведение аудита конфигураций CentOS 9 выявило двадцать шесть несоответствий эталонным требованиям, в их число вошли:

1. Настройка механизмов защиты ядра Linux.

2. Уменьшение периметра атаки ядра Linux.

3. Настройка средств защиты пользовательского пространства со стороны ядра Linux.

Действиями для устранения несоответствий являются: настройка паролей в файле /etc/shadow, установка для параметра PermitRootLogin значения no в файле /etc/ssh/sshd_config (запрет входа root по SSH), настройкой параметров sysctl, обеспечение ограничения доступа к команде su путём добавления в файл /etc/pam.d/su следующей строки: auth required pam_wheel.so use_uid, ограничение списка пользователей имеющих доступ к Sudo, установкой корректные права доступа к файлам настройки пользователей chmod 644 /etc/passwd chmod 644 /etc/group chmod go-rwx /etc/shadow.

В ходе работы был проведен аудит операционных систем Centos 9 и Astra Linux 1.6. Проведенные аудиты позволили выявить уровни уязвимостей данных систем, с чем они связаны, а также даны рекомендации по их устранению. В результате анализа было выявлено, что каждой операционной системе необходимо установка собственных параметров, с целью обеспечения информационной безопасности предприятия.

Библиографический список

1. Метод обеспечения и проведения внутреннего аудита информационной безопасности организаций на основе риск-ориентированного подхода // dissercat URL: https://www.dissercat.com/content/metod-obespecheniya-i-provedeniya-vnutrennego-audita-informatsionnoi-bezopasnosti-organizats (дата обращения: 24.10.2023).

2. Что необходимо знать для работы с RedCheck? // RedCheck URL: https://docs.redcheck.ru/articles/#!redcheck/what-need-know (дата обращения: 24.10.2023).

3. Инструкция администратора // RedCheck URL: https://docs.redcheck.ru /articles/#!redcheck-269/annotation (дата обращения: 24.10.2023).
4. National Vulnerbility Database // National Vulnerbility Database URL: https://nvd.nist.gov/ (дата обращения: 24.10.2023).

УДК 004.056.5

ОБЕСПЕЧЕНИЕ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ И ЗАЩИТЫ ОТ УГРОЗ ВЕБ-ПРИЛОЖЕНИЙ И ИХ СЕГМЕНТОВ НА АРХИТЕКТУРНОМ УРОВНЕ

А.С. Цуканов, Д.Е. Лащук
Научный руководитель: канд. техн. наук, доц. А.Н. Соколов
Южно-Уральский государственный университет,
г. Челябинск

Отмечена проблематика безопасности веб-приложений и их компонентов, в частности сегментов на архитектурном уровне, на предмет реализации возникающих уязвимостей и их инициации злоумышленниками. Процесс массового внедрения цифровых систем и ИТ-решений постоянно возрастает в государственном, а также корпоративном сегментах. Следовательно, крайне важно уделять особое внимание вопросу архитектуры ИБ программного обеспечения от действий несанкционированных пользователей. В статье рассмотрены подходы и методы проверок программных продуктов, обеспечение информационной безопасности при выстраивании концепции веб-приложения на архитектурном уровне и его сегментов от реализации угроз злоумышленников.

Ключевые слова: информационная безопасность, веб-приложения, уязвимости программного обеспечения, архитектура ИБ, безопасный жизненный цикл разработки.

Введение. Одним из ключевых вопросов в области информационной безопасности является защита веб-приложений. Поскольку веб-платформы используются практически повсеместно, иными словами, в каждой отрасли нашей жизни, тенденция угроз и различных векторов атаки не спадает, а вовсе, увеличивается постоянно. Уровень возможных атак возрастает, а значит, что в первую очередь ставится под угрозу доступность, целостность и конфиденциальность данных и систем.

Важность проблематики защиты от угроз веб-приложений и их сегментов на архитектурном уровне обуславливается ее актуальностью, поскольку выпуск программного обеспечения, которое пренебрегает или не в полной мере соблюдает принципы безопасного построения ИБ процессов в

архитектуре веб-продукта, подвержено различным сценариям нападения злоумышленников, начиная от SQL-инъекций, заканчивая DDoS-атаками.

Немаловажно сказать, что в представленной работе рассмотрены следующие аспекты:

– базовая модель архитектуры веб-приложений без применения средств защиты информации;

– улучшенная модель архитектуры веб-приложений с использованием средств защиты информации;

– контролирующие шаги по обеспечению информационной безопасности веб-продуктов.

Значимость темы статьи определяет ключевые требования к обеспечению информационной безопасности веб-приложений, затрагивая один из основных уровней – архитектурный, а также вовлечение в изучение дополнительных методов и способов защиты программных продуктов.

Базовая модель архитектуры веб-приложений без применения средств защиты информации.

Важность и понимание процесса исследования находит свое отображение на основополагающем этапе, иными словами, архитектурном уровне веб-приложения, который имеет трехуровневое представление:

– сегмент веб-браузера;

– уровень технологий (языки программирования, библиотеки, API и т.д.);

– база данных [1].

Таким образом, пользователь, находясь на первом уровне выполняет запрос в веб-браузере к соответствующему приложению, далее, уровень технологий обрабатывает запрос и отдает первичный ответ пользователю, а также обращается к базе данных для получения полного результата. Итак, базовая модель архитектуры веб-приложения представлена на рис. 1 [1, 2].

Необходимо отметить, что модель веб-приложения и его сегментов не используют СЗИ.

На наш взгляд, программные продукты такого формата наделены уникальными свойствами, из-за которых большинство коммерческих предприятий, а также государственных организаций выбирают именно их:

– обновления без побочного влияния на приложение;

– постоянная доступность для пользователей [1, 3].

Стоит выделить, что для злоумышленников, как и для обычных пользователей веб-приложение все также постоянно доступно из сети-интернет. Именно данный фактор несет за собой ключевую роль безопасности, поскольку на прямую зависит от того, как выстроена сегментарная архитектура ИБ для программного продукта.

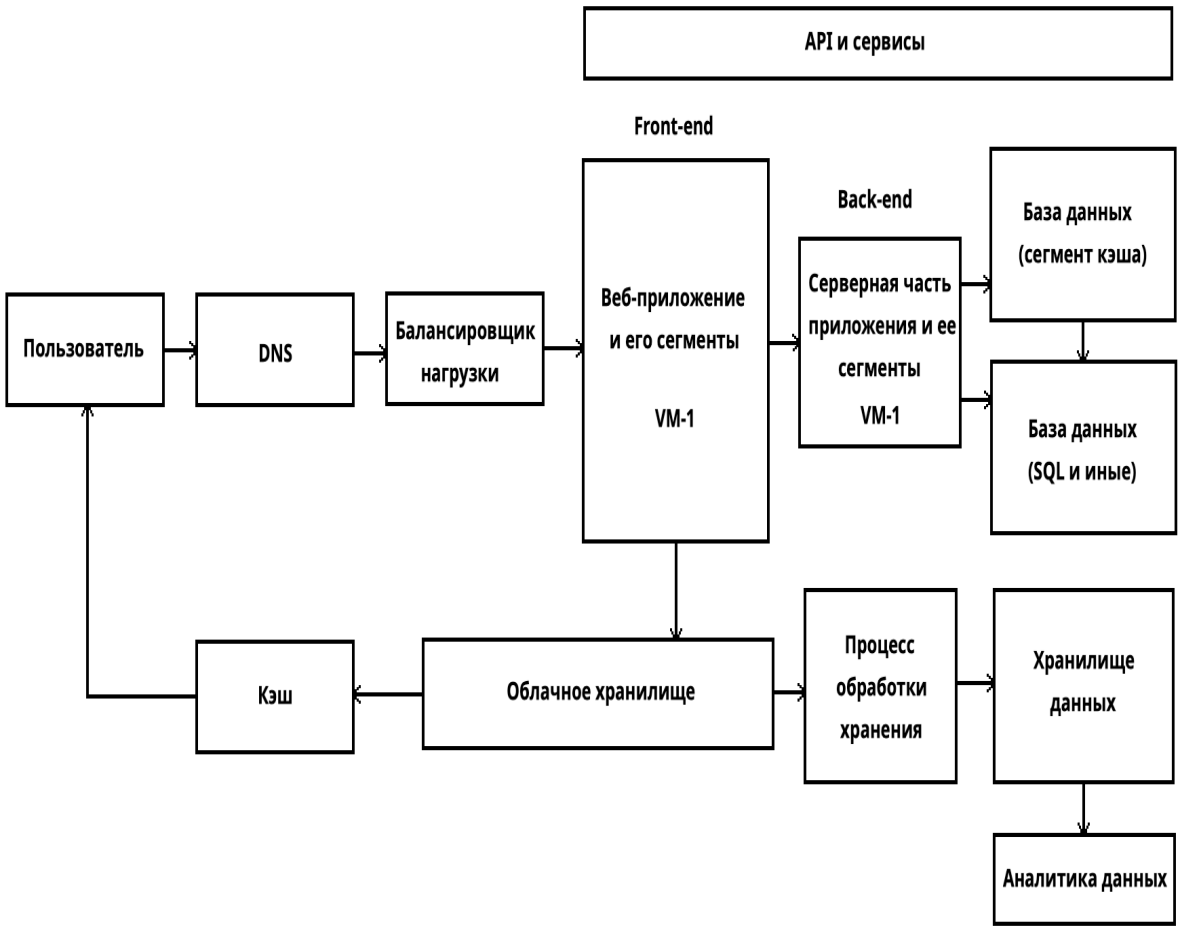


Рис. 1. Базовая модель архитектуры веб-приложений без применения СЗИ

Важно подчеркнуть, что зачастую, при реализации программного продукта используется принцип SSDLC (Secure Software Development Life Cycle), иными словами, безопасный жизненный цикл разработки, где существуют этапы проработки требований, архитектуры, тестирования и т.п. В случае, если ПО исполняется без данных итераций, то на выходе могут возникать проблемы с точки зрения информационной безопасности для веб-приложения.

Улучшенная модель архитектуры веб-приложений с использованием средств защиты информации.

В состав задач, обеспечивших проведение данной исследовательской работы, входил вопрос о представлении улучшенной модели архитектуры веб-приложения, с точки зрения использования средств защиты информации (рис. 2).

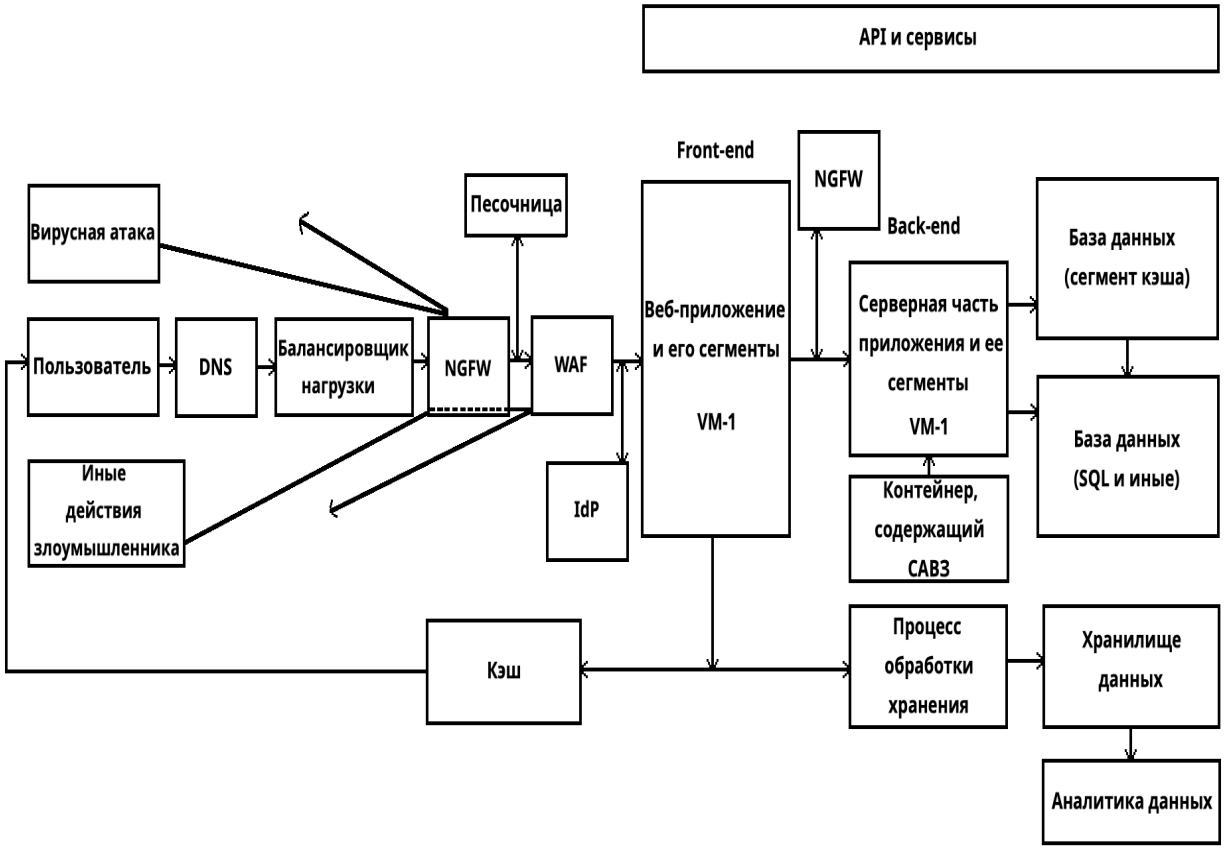


Рис. 2. Улучшенная модель архитектуры веб-приложения с использованием средств защиты информации

Стоит отметить, что были введены несколько возможных сценариев, с точки зрения нарушителя, которые отражают процесс улучшения:

– NGFW (Next-generation firewall), иными словами, межсетевой экран для глубокой фильтрации трафика, который в нашем случае является первым звеном с точки зрения защиты. При вирусной атаке, с большей вероятностью данное СЗИ будет отражать неправомерное действие, где проводится проверка сценария по сигнатурам NGFW [4];

– в случае иных несанкционированных действий злоумышленника, атака может проходить через сигнатуры NGFW, но в защиту вступает следующее средство безопасности – WAF (Web Application Firewall), который является межсетевым экраном для веб-приложений, выявляющий различного рода атаки, в том числе уязвимость нулевого дня. Таким образом, WAF будет являться дополнительным отражающем СЗИ от атак [5];

– важно подчеркнуть, что после NGFW была введена песочница, которая позволяет провести анализ возможных входящих файлов;

– применение IdP (Identity provider), иными словами, провайдер аутентификации, позволяет проводить проверку внешних пользователей и систем, что усиливает меры безопасности [6];

– дополнительными предложенными мерами защиты информации являются: внедрение второго NGFW на архитектурной связи веб-части и

серверного сегмента, позволяющего вновь пройти по сетевой модели OSI и другим функциям безопасности, а также контейнера, который содержит в себе подготовленное САВЗ. По нашему мнению, данные меры обеспечения защиты, позволят создать дополнительные слои проверки запросов на серверную часть, поскольку она напрямую связана с базой данных приложения.

Необходимо сказать, что в улучшенной модели архитектуры веб-приложения отсутствует облачное хранилище данных, поскольку оно может вызывать дополнительную область возможных атак. Таким образом, мы добились хорошо изолированных и защищенных компонентов программного обеспечения.

В связи с вышеупомянутым, требуется внести небольшое уточнение, а именно, настраиваемая сетевая часть (протоколы, порты, ip-адреса и т.п.) не была отражена на данных моделях, поскольку были затронуты общие концепции построения архитектуры ИБ, но для обеспечения безопасности программного продукта, это является крайне важным аспектом, на который всегда важно обращать внимание.

Контролирующие шаги по обеспечению информационной безопасности веб-продуктов.

К числу крайней задачи исследовательской работы относится рассмотрение и предложение контролирующих шагов, в рамках обеспечения ИБ для веб-приложений. Следует отметить, что их спектр довольно велик и в различных сценариях может быть иным.

Итак, можно выделить некоторый перечень контролирующих предложений:

– проверка программных продуктов сканерами на наличие уязвимостей (SAST, DAST, IAST);

– проведение тестирования на проникновение (penetration test);

– применение САВЗ для серверного сегмента;

– использование средств мониторинга для систем;

– проведение оценки безопасности архитектуры программного продукта и его сегментов;

– соблюдение цикла безопасной разработки (SSDLC).

Таким образом, важно сказать, что обеспечение защиты информации веб-приложений напрямую зависит от того, как устроена архитектура с точки зрения ИБ. Крайне необходимо, чтобы методики и улучшения в данной части построения программного продукта постоянно развивались, а, следовательно, количество угроз безопасности информации снижалось, что вызывает интерес к дальнейшим исследованиям по данному вопросу.

Библиографический список

1. Кинтонова А.Ж., Баенова Г.М., Урынбасарова А.Ж. Вопросы безопасности веб приложений // Colloquium-journal. 2020. №13 (65). Режим доступа: https://cyberleninka.ru/article/n/voprosy-bezopasnosti-veb-prilozheniy (дата обращения: 27.11.2023).

2. Abdulhannan Shaikh. Web Application Architecture: The Latest Trends and Best Practices for 2023. Режим доступа: https://www.peerbits.com/blog/web-application-architecture.html (дата обращения: 27.11.2023).

3. Михаил Черкашин. Архитектура современных веб-сервисов и способы их защиты. Режим доступа: https://www.anti-malware.ru/analytics/ Technology_Analysis/Architecture-of-modern-web-services (дата обращения: 27.11.2023).

4. Дмитрий Ким, Виктор Рыжков. NTA, IDS, UTM, NGFW – в чем разница? Режим доступа: https://www.securitylab.ru/analytics/517592.php (дата обращения: 27.11.2023).

5. Web Application Firewall. Режим доступа: https://habr.com/ru/companies /otus/articles/733142/ (дата обращения: 27.11.2023).

6. Блог компании NIX: Какой Identity-провайдер выбрать для реализации технологии Single Sign On. Режим доступа: https://habr.com/ru/companies /nix/articles/595997/ (дата обращения: 27.11.2023).

СЕКЦИЯ «МАТЕМАТИЧЕСКИЕ МЕТОДЫ И АНАЛИЗ ДАННЫХ В ОБЕСПЕЧЕНИИ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ»

УДК 004.056

АЛГОРИТМ РАСШИРЕНИЯ ТАБЛИЦЫ ЗАШИФРОВАНИЯ

А.В. Стрекалов, С.С. Титов
Научный руководитель: д-р физ.-мат. наук, проф. С.С. Титов
Уральский государственный университет путей и сообщения, г. Екатеринбург

В статье рассматривается задача расширения таблицы зашифрования совершенного шифра путём добавления новой шифрвеличины. При этом ставится условие сохранения свойства совершенности шифра. Для класса аффинных шифров, основанных на линейных функциях над конечными полями, предложен алгоритм погружения в проективную плоскость с бесконечно удалённой прямой. Этот алгоритм позволяет построить соответствующие инъекции зашифрования с неравновероятными ключами, делающие полученный шифр совершенным.

Ключевые слова: алгоритм расширения таблицы зашифрования, вероятности ключей, ключи, неравновероятные ключи, пассивный злоумышленник, который прикрывает активного, проективная плоскость, совершенный шифр, условные вероятности.

В качестве введения, перед тем как поставить чёткую цель, предложена к рассмотрению задача:

Дан шифр с **X={1,2,3,4,5}, Y=**{1,2,3,4,5}**, K={1,2,3,4,5,6,7,8}** с равновероятными ключами и таблицей зашифрования, где X – алфавит открытых текстов, Y – алфавит зашифрованных текстов, K – множество ключей (табл. 1).

Таблица 1

Таблица зашифрования

k	**1**	**2**	**3**	**4**	**5**	$\mathbf{P_k}$
1	1	2	3	4	5	0,125
2	2	3	4	5	1	0,125
3	3	4	5	1	2	0,125
4	3	5	1	2	4	0,125
5	4	1	5	2	3	0,125
6	4	5	2	1	3	0,125
7	5	1	2	3	4	0,125
8	5	4	1	3	2	0,125

Если в шифр тексте встретилось шифр-обозначение 2, то какова вероятность, что была зашифрована шифр-величина **4**?

Решение:

P{x=**4** | y=2} = 0,125 + 0,125 = 0,25

Итак, в решении этой задачи получился несовершенный шифр [1], так как при шифр-величине **3**, шифр-обозначение 4 встречается только один раз, и поэтому она имеет вероятность 0,125. Тогда как при шифр-величине **4**, шифр-обозначение 2 встречается два раза и полученная вероятность, как показано в примере выше, равна 0,25.

Для того, чтобы получше разобраться в совершенных шифрах [2], был рассмотрен геометрический подход к совершенным и почти совершенным шифрам [3]. И для дальнейшей работы очень понадобится проговорить, что такое совершенные шифры. Шифр называется совершенным, если апостериорные вероятности символов открытого текста совпадают с их априорными вероятностями.

Таблица линейных функций y = ax + b над полем GF(q) является шифр-таблицей совершенного шифра.

Таблица линейных функций y=*ax+b* над полем X={0,1,2,3}=GF(4), где числа 0,1,2,3 кодируют соответственно элементы 0,1,x,x+1 в фактор-кольце GF(2)[x]/f(x) по модулю неприводимого многочлена $f(x)=x^2+x+1$**:**

Занумеруем все ключи k=(a, b).

Зададим им равные вероятности $P_k = 1/12$ (табл. 2).

Таблица 2

Таблица линейных функций

k	a	b	0	1	2	3	P_k
1	1	0	**0**	**1**	**2**	**3**	1/12
2	1	1	**1**	**0**	**3**	**2**	1/12
3	1	2	**2**	**3**	**0**	**1**	1/12
4	1	3	**3**	**2**	**1**	**0**	1/12
5	2	0	**0**	**2**	**3**	**1**	1/12
6	2	1	**1**	**3**	**2**	**0**	1/12
7	2	2	**2**	**0**	**1**	**3**	1/12
8	2	3	**3**	**1**	**0**	**2**	1/12
9	3	0	**0**	**3**	**1**	**2**	1/12
10	3	1	**1**	**2**	**0**	**3**	1/12
11	3	2	**2**	**1**	**3**	**0**	1/12
12	3	3	**3**	**0**	**2**	**1**	1/12

P{x=0 | y=0} = 1/12 + 1/12 + 1/12 = 3/12

P{x=0 | y=1}= 1/12 + 1/12 + 1/12 = 3/12 и т.д.

Таким образом, Таблица линейных функций y=*ax+b* над полем GF(4) является шифр-таблицей совершенного шифра. На аффинной

плоскости 4 на 4 получаем 12 прямых, задающих инъекции зашифрования.

Цель работы: С сохранением свойства совершенности погрузить шифр-таблицу в проективную плоскость с добавлением новой шифр-величины.

«Бесконечность» заменяется числом для удобства – табл. 3 числом 4.

Таблица линейных функций y=*ax+b* над полем GF(4).

Таблица 3

Таблица линейных функций, то, что было

k	a	b	0	1	2	3	4	P_k
1	1	0	0	1	2	3	4	1/12
2	1	1	1	0	3	2	4	1/12
3	1	2	2	3	0	1	4	1/12
4	1	3	3	2	1	0	4	1/12
5	2	0	0	2	3	1	4	1/12
6	2	1	1	3	2	0	4	1/12
7	2	2	2	0	1	3	4	1/12
8	2	3	3	1	0	2	4	1/12
9	3	0	0	3	1	2	4	1/12
10	3	1	1	2	0	3	4	1/12
11	3	2	2	1	3	0	4	1/12
12	3	3	3	0	2	1	4	1/12

Первой задачей для достижения цели – модифицировать эту таблицу так, чтобы шифр стал совершенным. Этот шифр должен быть совершенным при любом распределении вероятностей {r0, r1, r2} на множестве X. И вероятности будут $r_0 \geq 0$, $r_1 \geq 0$, $r_2 \geq 0$, $r_0+r_1+r_2=1$. При этом ключи получатся неравновероятными (табл. 4).

Таблица 4

Таблица линейных функций, то, что стало

k	a	b	0	1	2	3	4	P_k
1	1	0	0	1	2	3	4	4/20
2	1	1	1	4	3	2	0	1/20
3	1	2	4	3	0	1	2	1/20
4	1	3	3	2	4	0	1	1/20
5	2	0	4	2	3	1	0	1/20
6	2	1	1	3	4	0	2	1/20
7	2	2	2	0	1	4	3	2/20
8	2	3	3	4	0	2	1	1/20
9	3	0	4	3	1	2	0	2/20
10	3	1	1	2	0	4	3	2/20
11	3	2	2	4	3	0	1	2/20
12	3	3	3	0	4	1	2	2/20

СЧИТАЕМ:

P {x=0 | y=1} = 1/20+1/20+2/20=4/20

P {x=0 | y=2} = 2/20+2/20=4/20 и т.д.

Да! Задача успешно выполнена, а на входе её выполнения был сформирован алгоритм выполнения таких задач для достижения цели, с любым полем.

Сформирован алгоритм, с помощью которого решается задача с одной оговоркой: Алгоритм не будет работать для GF(3), так как q должно быть больше 3. Иначе говоря, оно слишком маленькое. То есть, для GF(4) уже будет работать (Ну и предлагаю считать его универсальным).

Универсальный алгоритм для расширения:

1. Индивидуализация какой-либо строчки (присутствует вариативность). Для примера индивидуализировали первую строчку, это означает, что при x=0 y=0 встретится только один раз, x=1 y=1, x=2 y=2, x=3 y=3, x=4 y=4, x=5 y=5, x=6 y=6, x=7 y=7, x=8 y=8 тоже только один раз. Иначе говоря, псевдослучайно выбирается строчка I (назову её индивидуальной), берётся столбик x и делается так, чтобы y больше нигде не встретился, кроме этой строчки – заменить y в остальных строчках в столбике x на q, и так проделать со всеми столбиками. В результате останется некоторое ключевое пространство (я назвал его главным). Оставшееся ключевое пространство будет равно количеству столбиков минус два.

2. Теперь нужно произвести замены в главном ключевом пространстве. Выбирается любая строчка вне главного ключевого пространства и не индивидуальная. Предлагаю назвать эту строчку главной.

По этой главной строчке, в главном ключевом пространстве производятся замены: начиная от x=0, заканчивая x=q-1, так как при x=q уже нечего будет заменять, там будет все перестановлено. Очень важен один нюанс: в один момент, при перестановках, при каком-то x, y=q. Такой столбик мы пропускаем и переходим к следующему.

3. Как рассчитывать вероятность, которую будем ставить в индивидуальную строчку? Всё очень просто! Используем формулу:

$$I = \frac{W}{L}\ , \tag{1}$$

где L – количество ключей k.

Рассчитали и сразу же поставили вероятность.

4. Переходим к главной строчке. В ней вероятности нужно расставить, используя формулу:

$$M = \frac{I}{W - 1}\ . \tag{2}$$

Начинаем расстановку вероятностей в других строчках следующим образом: начинаем с главной строчки. Здесь работает правило: для каждого x

нужно найти все y и выставить у этих строк вероятность, равную главной строчке.

По итогу, у нас остаются строчки без вероятностей. Это ключевое пространство названо “пространством двоечников” по одной причине: в этом ключевом пространстве в каждом x по два y. Вероятность каждой строчки двоечника рассчитывается по формуле:

$$N = \frac{M}{2} . \quad (3)$$

5. Все строчки приводятся к общему знаменателю.

При этом используются формулы:

$$M_n = 2 * N , \quad (4)$$

$$I_n = \left(\frac{I}{N}\right) * N . \quad (5)$$

6. Теперь сумма вероятностей, исключающая вероятность индивидуальной строчки равна 1. Зато, учитывая индивидуальную строчку, получаем сумму вероятностей больше 1.

Чтобы это исправить, нужно:

1. Сосчитать, сколько будет та самая 1 + вероятность индивидуальной строчки. Делается это по формуле:

$$S = I_n + 1 . \quad (6)$$

2. Теперь считаем новые вероятности:

$$I_{new} = \frac{I_n}{S} . \quad (7)$$

Для остальных так же:

$$M_{new} = \frac{M_n}{S} , \quad (8)$$

$$N_{new} = \frac{N}{S} . \quad (9)$$

Используя формулы (1)–(9) получается пример работы алгоритма:

Таблица линейных функций y=*ax*+*b* над полем GF(5) (табл. 5–6):

Таблица 5

Таблица линейных функций, то, что было

k	a	b	0	1	2	3	4	5	P_k
1	1	0	0	1	2	3	4	5	1/20
2	1	1	1	2	3	4	0	5	1/20
3	1	2	2	3	4	0	1	5	1/20
4	1	3	3	4	0	1	2	5	1/20
5	1	4	4	0	1	2	3	5	1/20
6	2	0	0	2	4	1	3	5	1/20
7	2	1	1	3	0	2	4	5	1/20
8	2	2	2	4	1	3	0	5	1/20
9	2	3	3	0	2	4	1	5	1/20
10	2	4	4	1	3	0	2	5	1/20
11	3	0	0	3	1	4	2	5	1/20
12	3	1	1	4	2	0	3	5	1/20
13	3	2	2	0	3	1	4	5	1/20
14	3	3	3	1	4	2	0	5	1/20
15	3	4	4	2	0	3	1	5	1/20
16	4	0	0	4	3	2	1	5	1/20
17	4	1	1	0	4	3	2	5	1/20
18	4	2	2	1	0	4	3	5	1/20
19	4	3	3	2	1	0	4	5	1/20
20	4	4	4	3	2	1	0	5	1/20

Таблица 6

Таблица линейных функций, то, что стало

k	a	b	0	1	2	3	4	5	P_k
1	1	0	0	1	2	3	4	5	6/36
2	1	1	5	2	3	4	0	1	1/36
3	1	2	2	5	4	0	1	3	1/36
4	1	3	3	4	5	1	2	0	1/36
5	1	4	4	0	1	5	3	2	1/36
6	2	0	5	2	4	1	3	0	1/36
7	2	1	1	3	0	2	5	4	2/36
8	2	2	2	4	1	5	0	3	1/36
9	2	3	3	0	5	4	1	2	1/36
10	2	4	4	5	3	0	2	1	1/36
11	3	0	5	3	1	4	2	0	2/36
12	3	1	1	4	5	0	3	2	2/36
13	3	2	2	0	3	1	5	4	2/36
14	3	3	3	5	4	2	0	1	2/36
15	3	4	4	2	0	5	1	3	2/36

Окончание табл. 6

k	a	b	0	1	2	3	4	5	P_k
16	4	0	5	4	3	2	1	0	2/36
17	4	1	1	0	4	5	2	3	2/36
18	4	2	2	5	0	4	3	1	2/36
19	4	3	3	2	1	0	5	4	2/36
20	4	4	4	3	5	1	0	2	2/36

СЧИТАЕМ:

P{x=0 | y=1} = 2/36+2/36+2/36 = 6/36

P{x=0 | y=2} = 1/36+1/36+2/36+2/36 = 6/36 и т.д.

Проверка:

6/36+(8*1)/36+(11*2)/36 = 1

Вывод по примеру имеем следующий: цель поставлена и успешно выполнена, созданный алгоритм проверен и работает.

Дальнейшее направление для исследований: сделан алгоритм для y=ax+b, а интересен вопрос, а как обстоят дела с другими аффинными шифрами, как их расширить, чтобы они остались совершенными, чувствуется, что это можно не всегда, и это трудная задача, а если расширить невозможно, то актуальна задача построить почти совершенный шифр, то есть, нахождение такого распределения вероятностей, при которой отклонение условного распределения вероятностей от нормального минимально.

Работа может быть полезна студентам при изучении криптографии, аффинных шифров, конечных геометрий.

Применение: совершенные шифры нужны, чтобы передавать короткие и важные сообщения. В алгоритме ключи неравновероятны – почему это важно и как использовать: передаваемый сигнал по стоимости разный. Логично использовать дорогие сигналы реже. Алгоритм может быть применён для сигналов с разной стоимостью. Более дорогие используем реже, они забивают канал связи. Почти совершенные шифры были придуманы относительно недавно, они были предложены от безысходности, когда условные вероятности почти равны и при решении заменялся совершенный шифр на почти совершенный, а в полученном алгоритме для ax+b, где a, b – части ключа, ничего заменять не надо. Раньше было так – если вкладывать в аффинную плоскость проективную, ломается совершенность и вводили почти совершенность, а теперь, с полученным алгоритмом, не ломается. Итог – таблица есть, вероятность есть, работает. На входе алгоритма таблица сложения умножается и складывается в поле, на выходе получается таблица зашифрования с вероятностями ключей. Расширение шифра можно использовать, если в аппарате добавилась новая кнопка и нужно, чтобы эта кнопка не ухудшала характеристики. Ведь если шифр совершенный, то активный злоумышленник, которого прикрывает пассивный, не смогут понять, какие сигналы подаёт аппарат.

В заключение, были получены весьма интересные и полезные результаты. Был произведён глубокий анализ и учтено множество факторов. Поставленная цель: с сохранением совершенства погрузить шифр-таблицу в проективную плоскость с добавлением новой шифр-величины, была решена. Также алгоритм сработает и для более масштабных таблиц, а самое главное – не забывать, что таблица составляется по двум другим – таблице сложения и умножения, которые свои для каждого поля.

Библиографический список

1. Зубов А.Ю. Криптографические методы защиты информации. Совершенные шифры. М.: Гелиос АРВ, 2005.

2. Зубов А.Ю. О понятии ε – совершенных шифров // Прикладная дискретная математика. 2016. № 3(33).

3. Зубов А.Ю. Почти совершенные шифры и коды аутентификации // Прикладная дискретная математика. 2011. № 4(14). С. 28–33.

4. Геут К.Л., Титов С.С. О рекуррентных соотношениях в информационной безопасности. Вестник УрФО. Безопасность в информационной сфере. 2017. №1(23). С. 24–27.

УДК 004.510.6

О ЗАДАЧЕ NSUCRYPTO 2022 SUPER DEPENDENT S-BOX

И.Н. Боровков, К.Л. Геут
Научный руководитель: ст. преподаватель К.Л. Геут
Уральский государственный университет путей сообщения,
г. Екатеринбург

В статье приведено решение задачи олимпиады *NSUcrypto* второго раунда 2022 года, дана характеристика S-блока и суперзависимого *S*-блока, найдено количество перестановок для $n = 2$ и $n = 3$.

Ключевые слова*: S*-блок, *NSUcrypto* 2022, булевы функции.

S-блок (*S-box*) – это компонент, используемый в блочных шифрах для замены одних битов на другие во входных данных блока [1]. *S*-блок может быть независимым и суперзависимым (*super dependent)*, в зависимости от того, как один *S*-блок связан с другим *S*-блок в шифре.

Суперзависимый *S*-блок зависит от других компонентов шифра или других *S*-блоков. Их использование может усилить стойкость шифра к различным атакам, таким как дифференциальный или линейный криптоанализ.

В задаче требуется найти суперзависимый *S*-блок. Для этого используются перестановки, существенно зависящие от каждой из своих переменных. Необходимо оценить количество таких перестановок.

Определение.

Векторная булева функция $F(x) = (f_1(x), f_2(x), \ldots, f_n(x)))$, *где* $x \in \mathbb{F}_2^n$, является перестановкой на $\mathbb{F}_2^n$, если она является взаимно однозначным отображением на множество $\mathbb{F}_2^n$.

Координатная функция f(x) (т. е. булева функция от $\mathbb{F}_2^n$ до F_2) существенно зависит от переменной x_j, если существуют значения $b_1, b_2, \ldots, b_{j-1}, b_{j+1}, \ldots, b_n \in \mathbb{F}_2$ такие, что $f_k(b_1, b_2, \ldots, b_{j-1}, 0, b_{j+1}, \ldots, b_n) \neq f_k(b_1, b_2, \ldots, b_{j-1}, 1, b_{j+1}, \ldots, b_n)$. Другими словами, существенная зависимость от переменной x_j функции f означает наличие x_j в алгебраической нормальной форме f (уникальное представление функции в основе бинарных операций AND, *X*OR и констант 0 и 1).

Для выполнения свойств *S*-блока, описанных в условии задачи, должно выполняться два условия:

1. Все n переменных должны входить в полином Жегалкина каждой из n координатных функций;

2. Получившаяся булева функция $F(x) = (f_1(x), f_2(x), \ldots, f_n(x))$ должна быть биективна [2].

Рассмотрим задачу для $n = 2, 3$.

Из условия биективности следует, что нужно рассматривать только уравновешенные функции, чтобы в результате появились все элементы множества $\mathbb{F}_2^n$ [3, С. 13].

Уравновешенной функцией называется такая булева функция, которая на всей области определения функции принимает значение 0 ровно столько же раз, как и значение 1.

Для $n = 2$ имеем 5 вариантов с операциями *AND*, *XOR* с двумя переменными x_1, x_2:

$k_1 = x_1x_2 = (0, 0, 0, 1)$;

$k_2 = x_1 \oplus x_2 = (0, 1, 1, 0)$;

$k_3 = x_1x_2 \oplus x_1 = (0, 0, 1, 0)$;

$k_4 = x_1x_2 \oplus x_2 = (0, 1, 0, 0)$;

$k_5 = x_1x \oplus x_1 \oplus x_2 = (0, 1, 1, 1)$.

Из функций набора x_1, x_2 только функция k_2 является уравновешенной.

Рассмотрим комбинации данных функций с $\oplus 1$:

$m_1 = x_1x_2 \oplus 1 = (1,1,1,0) = \overline{k_1}$

$m_2 = x_1 \oplus x_2 \oplus 1 = (1,0,0,1) = \overline{k_2}$

$m_3 = x_1x_2 \oplus x_1 \oplus 1 = (1,1,0,1) = \overline{k_3}$

$m_4 = x_1x_2 \oplus x_2 \oplus 1 = (1,0,1,1) = \overline{k_4}$

$m_5 = x_1x_2 \oplus x_1 \oplus x_2 \oplus 1 = (1,0,0,0) = \overline{k_5}$

Рассмотрим комбинации функции k_2 с инвертированием элементов:

$p_1 = \overline{x_1} \oplus x_2 = (1,0,0,1) = m_2$
$p_2 = x_1 \oplus \overline{x_2} = (1,0,0,1) = m_2$
$p_3 = \overline{x_1} \oplus \bar{x} = (0,1,1,0) = k_2$

Для $n = 2$ только одна функция (и ее инверсия) удовлетворяют условиям задачи, следовательно, искомых перестановок не существует.

Рассмотрим существование перестановок для $n = 3$.

Для появления всех наборов в итоговой перестановке условие уравновешенности функций должно сохраняться. Проверим полиномы, составленные из элементов x_1, x_2, x_3, x_1x_2, x_1x_3, x_2x_3, на уравновешенность.

Путем перебора 256 комбинаций из 3 элементов x_1, x_2, x_3 со знаками (*XOR*) и (*AND*) найдено 29 уравновешенных полиномов. Добавление (*XOR*) $x_1x_2x_3$ нарушает уравновешенность функций, поэтому функции с данным элементом исключены из расчета.

Далее произведем проверку комбинации двух функций с исключением повторяющихся троек. Проверка сочетаний трех функций на уникальность (биективность) дала 512 троек функций, удовлетворяющих условию. Прибавление $\oplus 1$ не влияет на уравновешенность функции и не влияет на баланс наборов элементов в перестановке.

После проверки всех комбинаций (без учета перестановок) получаем 512 функций из трех переменных.

Перестановки функций увеличивают количество в 3! раз.

При инвертировании функций в каждой тройке количество перестановок увеличивается в 8 раз.

Для функций двух переменных ($n = 2$) не существует перестановок суперзависимого *S*-блока с заданными условиями.

Для функций трех переменных ($n = 3$) количество перестановок суперзависимого *S*-блока с заданными условиями 512*8*3! = 24 576.

В октябре 2023 года было опубликовано официальное решение данной задачи, которое совпало с найденным значением [4].

Применение.

Блочные шифры широко используются в криптографии [6]. Суперзависимые *S*-блоки имеют более высокую нелинейность по сравнению со стандартными S-блоками. Нелинейность – важное свойство криптографии, поскольку злоумышленникам становится сложнее находить линейные приближения или алгебраические уравнения, которые могут взломать шифрование. Повышенная нелинейность в суперзависимых *S*-блоках обеспечивает более сильную устойчивость к дифференциальному и линейному криптоанализу, тем самым обеспечивает надежную защиту конфиденциальных данных [5].

Суперзависимые *S*-блоки вносят более высокий уровень сложности, гарантируя, что каждый выходной бит зависит от нескольких входных битов. Это затрудняет злоумышленникам выявление закономерностей или связей между открытым и зашифрованным текстами.

Библиографический список

1. Как создаются S-блоки // Хабр URL: https://habr.com/ru/articles/533732/ (дата обращения 18.10.2023).

2. Введение в теорию множеств // Хабр URL: https://habr.com/ru/articles/457312/ (дата обращения 15.10.2023).

3. Агафонова И.В. Криптографические свойства нелинейных булевых функций. 2007. – С. 13.

4. Idrisova V.A., Tokareva N.N., Gorodilova A.A., Beterov I.I., Bonich T.A., Ishchukova E.A., Kolomeec N.A., Kutsenko A.V., Malygina E.S., Pankratova I.A., Pudovkina M.A., Udovenko A.N. Problem «Super dependent S-box» // Mathematical problems and solutions of the Ninth International Olympiad in Cryptography NSUCRYPTO. 2023. – С. 18–19.

5. Молдовян А.А. Криптография Скоростные шифры / А. А. Молдовян, Н.А. Молдовян, Н.Д. Гуц, Б.В. Изотов, – Санкт-Петербург: БХВ-Петербург, 2002. – С. 231–235.

6. Бондакова О.С., Зязин В.П. О первичной оценке вероятности возникновения коллизии для уменьшенного варианта хэш-функции «СТРИБОГ» // Вестник УрФО Безопасность в информационной сфере. – 2018. – №4(30). – С. 27–30.

УДК 004.056

ЗАДАЧА «ГИПОТЕЗА», ПРОБЛЕМА NSUCRYPTO – 2015

М.А. Середа, С.С. Титов
Научный руководитель: д-р физ.-мат. наук, проф. С.С. Титов
Уральский государственный университет путей сообщения, г. Екатеринбург

Статья посвящена анализу и построению решения задачи второго тура олимпиады NSUCRYPTO.

Ключевые слова: NSUCRYPTO, криптография, булевы функции, битовый ряд, программирование.

Участвуя в олимпиаде под названием NSUCRYPTO, мы узнали о разделе с задачами «Unsolved» – задачи, которые не смогли решить во время проведения данной олимпиады, и по сей день решения у этих задач отсутствует. Мы решили попытать свои силы в решении одной из них. Задача № 5 второго раунда 2015 года «Hypothesis». Для дальнейшей работы нужно сформулировать цель, а также план работы.

Цель: решение проблемы № 5 второго раунда NSUCRYPTO 2015 года.

План работы: проанализировать условие задачи, понять от чего отталкиваться, проработать примеры, на их основе построить свои примеры, удовлетворяющие условию, построить свою гипотезу, доказать приведенную гипотезу, что приведет к решению проблемы.

Данная задача посвящена проблеме поиска булевой функции, с помощью которой возможно построение двоичной последовательности с минимальным периодом равным 2^n. Что в свою очередь является проблемой генерации псевдослучайных последовательностей для создания ключей шифра гаммирования. Шифр гаммирования – это метод симметричного шифрования, заключающийся в «наложении» последовательности, состоящей из случайных чисел, на открытый текст. Создание и использование данных последовательностей позволяет обеспечить эффективную защиту данных от криптоанализа.

В задаче требуется доказать, что существуют функции для любого n, такие что удовлетворяют двум поставленным условиям [1]:

1. ДНФ функции не должна включать повторы переменных, то есть ДНФ вида x1x2 V x3 подходит под наше условие, а x1 V x2 V x2x3 не удовлетворяет нашему условию.

2. Существует битовый ряд U, который составляется по определенному правилу (рис. 1).

$$u_{t+n} = u_t \oplus g(u_{t+1}, u_{t+2}, \ldots, u_{t+n-1})$$

Рис. 1. Правило битового ряда

Где n является количеством переменных в функции g() и определяет длину стартового значения, в свою очередь стартовые значения – это первые n битов, выбранные случайным образом из поля F_2. Функция g() – это искомая функция с минимальным периодом равным 2^n, существование которой требуется доказать.

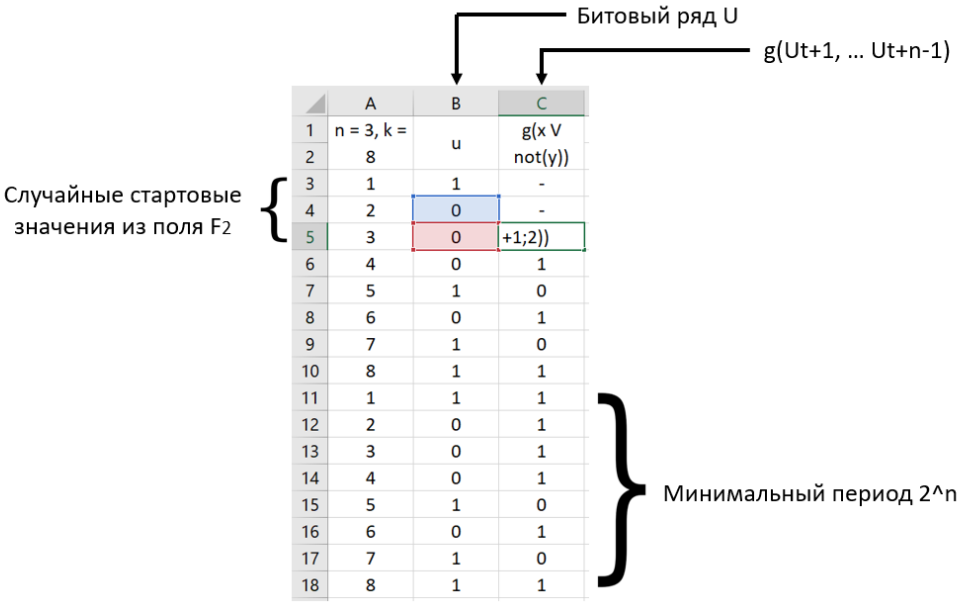
К данной задаче представлены примеры функций, которые удовлетворяют условиям для малых n (рис. 2).

n	the examples of $g(x_1, \ldots, x_{n-1})$
2	1
3	$x_1 \vee \overline{x}_2$
4	$x_1 \vee \overline{x}_2\overline{x}_3, \quad x_1 \vee x_2 \vee \overline{x}_3$
5	$x_2 \vee \overline{x}_1\overline{x}_3\overline{x}_4, \quad x_1 \vee x_2x_3 \vee \overline{x}_4$

Рис. 2. Примеры функций

Требуется проверка данных функций, которая будет заключаться в построении вектора функции g() и на его основе будем записывать битовый ряд U. Если построенный нами ряд будет иметь минимальный период 2^n при любых стартовых значениях, то функции удовлетворяют условию задачи.

Рассмотрим подробнее принцип построения для n = 3 (рис. 3).



	A	B	C
1	n = 3, k =	u	g(x V not(y))
2	8		
3	1	1	-
4	2	0	-
5	3	0	+1;2))
6	4	0	1
7	5	1	0
8	6	0	1
9	7	1	0
10	8	1	1
11	1	1	1
12	2	0	1
13	3	0	1
14	4	0	1
15	5	1	0
16	6	0	1
17	7	1	0
18	8	1	1

Рис. 3. Проверка для n = 3

Проверим примеры приведенные в условии задачи. Нужно построить таблицы истинности и битовый ряд u по вышеописанному правилу. После выполнения вышеописанного алгоритма мы можем наглядно убедиться, что для приведенных функций выполняется условие битового ряда u (в каждой таблице мы видим повторение ряда начинающиеся с позиции 2^{n+1}).

Нужна закономерность, которая позволит строить функции g() для любого n. Для этого требуется анализ уже имеющихся функций. Построим таблицы истинности для каждой [2].

Можно заметить, что все функции, которые удовлетворяют правилу, не сохраняют «0» на нулевом наборе и сохраняют «1» на единичном. Давайте использовать эту закономерность. Для того чтобы записать функцию, удовлетворяющую правилу, нужно составить вектор значений функции F так чтобы функция не сохраняла «0» и сохраняла «1». Для любого n функция F имеет $2^{(n-1)}$ значений следовательно вектор функции выглядит F(1*…*1), где «*» это логический «0» или «1», количество этих * равно $2^{(n-1)-2}$. Теперь мы готовы составить общую формулу нашей функции для любого n:

$$x_1 \, V \, x_2 \, \ldots V \, x_{n-2} \, V \, not(x_{n-1})$$

Данная функция удовлетворяет двум нашим основным правилам: ДНФ данной функции не включает в себя повторы переменных и данный вид функций не сохраняет «0» и сохраняет «1».

Давайте проверим правило, которое мы сейчас вывели. Возьмем n = 8, значит наша функция по правилу выглядит следующим образом:

$$x_1 \, V \, x_2 \, V \, x_3 \, V \, x_4 \, V \, x_5 \, V \, x_6 \, V \, not(x_7)$$

ДНФ данной функции не включает повторов переменных, давайте проверим ее на сохранение «1» и не сохранение «0», построив мини таблицу истинности.

Для проверки данного вида функций сделаем выборку n [5, 20], используя алгоритм проверки, построим битовый ряд и найдем минимальный период. С помощью программной среды Python напишем программу для автоматизации построения битового ряда [3]. Программа состоит из трех основных частей (рис. 4–6).

```
# функция g() x1 V x2 V... xn-2 V not(xn-1)
def f(n1, n2):
    s = ''
    for x in range(n1, n2+1):
        s += str(a[x])
    if s[-1] == '0':
        return 1
    elif s.count('1') == 1 and s[-1] != '1' or s.count('1') > 1:
        return 1
    else:
        return 0
```

Рис. 4. Блок, задающий вектор функции

```
a = ['_']
n = int(input('n = '))
for g in range(1, n+1):
    un = int(input())
    a.append(un)
print(a)
```

Рис. 5. Блок, задающий стартовые значения

```
for i in range(n+1, 2**(n+1)+1):
    un = (a[i-n] + f(i-n+1, i-1))%2
    a.append(un)
print(a)
```

Рис. 6. Блок построения битового ряда

После построения рядов вручную проверяем период. Проверив периоды функций, понимаем, что для n [5, 20] построенные функции не удовлетворяют условию задачи, а значит первая гипотеза неверна. Нужно искать новые закономерности и усовершенствовать программу для построения, чтобы оптимизировать временные затраты на проверку функций.

Упростим ввод функции, сделаем перебор всех начальных значений для битового ряда, а также заставим программу анализировать построенные ею ряды (рис. 7, 8).

```
import random
from itertools import product

def rd(d, fc):
    if fc[0] == 'n':
        c = not(int(d[int(fc[2])-1]))
        p = 3
    else:
        c = int(d[int(fc[1])-1])
        p = 2

    for y in range (p, len(fc)):
        if fc[y] == '+':
            o = '+'
        if fc[y] == '*':
            o = '*'
        if fc[y] == 'x':
            if fc[y-1] != 'n':
                m = int(d[int(fc[y+1])-1])
                if o == '+':
                    c = c or m
                else:
                    c = c and m
            else:
                m = not(int(d[int(fc[y+1])-1]))
                if o == '+':
                    c = c or m
                else:
                    c = c and m
    return c
```

Рис. 7. Блок считывания и построения

```
fc = input('Функция: ')
n = int(input('n = '))
res=[]
ls = []

for i in range(2**n):
    s=""
    for j in range(n):
        s=str(i%2)+s
        i=i//2
    res.append(s)

for v in range(0, 2**n):
    a = res[v]
    b = a



    for i in range (n, (2**n)*2):
        d = ''
        d += a[i-n+1:i]
        rs = (int(a[i-n]) + int(rd(d, fc)))%2
        a += str(rs)



    for i in range(2, 2**n+1):
        r = 1
        g = a[0:i]
        for j in range(0,len(a), i):
            h = a[i+j: 2*i+j]
            if g != h and len(g) == len(h):
                r = 0
                break
            else:
                g = h
        if r == 1 and not(i in ls) and 2**n % i == 0:
            ls.append(i)
            break

ls.sort()
print("Минимальный период =", ls[0])
```

Рис. 8. Блок ввода и анализа

Теперь программа получилась удобной в использовании, так как считывает консольный ввод функции в привычной для всех записи, а самое

главное выводить минимальный период из всех возможных комбинаций стартовых значений.

Можно использовать примеры для создания новых функций, как мы убедились ранее, все примеры удовлетворяют условию задачи. Попробуем скомбинировать данные функции и посмотреть получиться ли составить подходящую для задачи (табл. 1).

Таблица 1

Комбинирование примеров

Функция	Период 2^n
Удвоение	
X1 V nX2 V X3 V nX4	–
X1 V nX2nX3 V X4 V nX5nX6	–
X1 V X2 V nX3 V X4 V X5 V nX6	–
X2 V nX1nX3nX4 V X6 V nX5nX7nX8	–
X1 V X2X3 V nX4 V X5 V X6X7 V nX8	–
Комбинации	
X1 V nX2 V X3 V nX4nX5	–
X1 V nX2 V X3 V X4 V nX5	–
X1 V nX2 V X4 V nX3nX5nX6	–
X1 V nX2 V X3 V X4X5 V nX6	–
X1 V nX2nX3 V X5 V nX4nX6nX7	–
X1 V nX2nX3 V X4 V X5X6 V nX7	–
X1 V nX2 V nX3 V X5 V nX4nX6nX7	–
X1 V nX2 V nX3 V X4 V X5X6 V nX7	–

Комбинация примеров тоже не дала положительных результатов. Среди составленных не нашлось функции, которая сохраняет период 2^n.

Тогда нужно составить для каждого указанного в примере n методом перебора функции, которые удовлетворяют и не удовлетворяют условию.

Составленные ранее функции помогут нам проанализировать векторы функций, подходящий и не подходящий по условию. Булевым вектором называется упорядоченный набор (α_1, α_2, α_3, … α_n), где α_i принимают значения 0 или 1.

Составим сводную таблицу векторов для каждого приведенного n (табл. 2).

Основываясь на данной таблице, можно предположить, что у подходящих функций определенное соотношение нулей и единиц, а также их позиции влияют на результат. Требуется понять закономерности позиций от увеличения n.

Таблица 2

Векторы функций

Удовлетворяют	Не удовлетворяют
n = 3	
1101	0111
1011	1110
n = 4	
11110111	11011111
11010101	10110011
11110001	11001101
n = 5	
1000111100001111	1111011111111111
n = 6	
1000000000000000111111111111111	1111111101111111111111111111111

Отрицательный результат тоже результат. Была проделана большая работа по автоматизации проверки условия и автоматизации построения битовых рядов. Было определено, что методика состыковки подходящих функций не дает результата и двигаться в данном направлении бессмысленно. Обнаружен один из критериев для построения правильных функций, но как показала практика требуется больше, чем один критерий. Перспективным направлением является анализ векторов уже проверенных функций для нахождения закономерностей и составления идеального шаблона для записи функций любого n.

Библиографический список

1. Problem 5. "Hypothesis" // NSUCRYPTO URL: https://nsucrypto.nsu.ru/archive/2015/round/2/task/5/#data (дата обращения: 10.09.2023).

2. Определение булевой функции. Основные сведения // IFMO URL: https://neerc.ifmo.ru/wiki/index.php?title=Определение_булевой_функции (дата обращения: 10.09.2023).

3. Самоучитель Python // PythonWorld URL: https://pythonworld.ru/samouchitel-python (дата обращения: 10.09.2023).

УДК 004.032.26

ОБНАРУЖЕНИЕ АНОМАЛИЙ, ВЫЗВАННЫХ КИБЕРАТАКАМИ, В НАБЛЮДАЕМЫХ ПРОЦЕССАХ АСУ ТП С ИСПОЛЬЗОВАНИЕМ САМООРГАНИЗУЮЩЕЙСЯ КАРТЫ КОХОНЕНА

А.Д. Плетенкова
Научный руководитель*: *канд. техн. наук, доц. А.Н. Соколов
Южно-Уральский государственный университет,
г. Челябинск

В данной статье исследуется задача обнаружения аномалий, вызванных кибератаками, в процессах автоматизированной системы управления технологическими процессами. Предлагается использовать самоорганизующуюся карту Кохонена для обнаружения аномалий, основываясь на построении качественной модели и вычислении порогового значения. Исследование проведено на модели с синтетически сгенерированным набором данных. Полученные результаты показывают эффективность предложенного подхода в обнаружении аномалий, что делает его перспективным для применения в системах безопасности промышленных объектов.

Ключевые слова: обнаружение аномалий, пороговое значение, самоорганизующаяся карта Кохонена, DBSCAN, АСУ ТП.

Обнаружение аномалий, вызванных кибератаками, в автоматизированных системах управления технологическими процессами (АСУ ТП) является актуальной задачей в области информационной безопасности. Кибератаки могут привести к сбоям в работе системы, утечке конфиденциальных данных и серьезным материальным потерям. В связи с этим, разработка эффективных методов обнаружения и классификации аномалий становится крайне важной, как и вопрос построения систем обнаружения вторжений, использующих данные методы. Существует множество исследований по обнаружению и классификации вторжений с использованием методов вычислительного интеллекта, а именно искусственных нейронных сетей.

Одним из таких подходов, применяемых для обнаружения аномалий, является использование нейронной сети под названием самоорганизующаяся карта Кохонена (SOM), которая способна проецировать многомерные данные на двухмерное пространство. Она создает топологическую карту, которая сохраняет отношения между объектами в исходном пространстве.

Алгоритм самоорганизующейся карты был разработан более двух десятилетий назад, но его успех в различных областях науки на протяжении многих лет превосходит многие другие нейронные алгоритмы на сегодняшний день.

В работах [1, 2] предлагается использовать самоорганизующиеся карты для обнаружения аномалий в характеристиках системных вызовов внутри компьютерной сети. Самоорганизующиеся карты также используются в работах [3, 4] для обработки и кластеризации данных о сетевом трафике. Описанные методы работы с самоорганизующимися нейронными сетями показывают значительное повышение точности обнаружения аномалий при анализе компьютерных сетей передачи данных, а также помогают при составлении топологии компьютерной сети и поиске ошибок в существующих сетях [5].

Возможности кластеризации SOM позволяют использовать его в качестве эффективного детектора аномалий, который можно использовать в системах реального времени, в зависимости от решаемой проблемы. Детекторы аномалий, независимо от модели, действуют по принципу сравнения данных эталона с текущей обстановкой и сигнализации, при выходе за указанный порог. Любая модель должна иметь эталон [6].

Одним из способов использования самоорганизующейся карты Кохонена в качестве порогового детектора является обучение карты на данных и использование ее для классификации новых данных. Предполагается, что данные представляют собой информацию о нормальной работе АСУ ТП, а новые данные могут быть потенциально аномальными, вызванными кибератакой или другими внешними воздействиями.

Рассмотрим модель, написанную на библиотеке MiniSom на языке программирования Python. В виде графика рассеяния (рис. 1) отображается сгенерированный набор данных на двумерной плоскости.

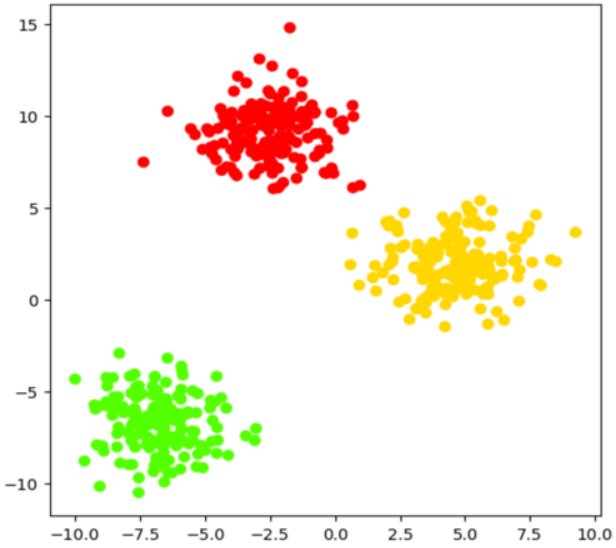


Рис. 1. График рассеяния набора данных

Синтетический набор данных был сгенерирован с использованием функции make_blobs() из модуля sklearn.datasets. Набор данных состоял из 500 случайных точек данных, или в данном случае, 500 случайных выборок, каждая с двумя признаками (n_features=2) и условно разбитая на три

кластера (centers=3). Каждая точка данных представляет собой одну выборку (x[:, 0] и x[:, 1] - координаты этой точки), а цвет точек соответствует определенному классу точек данных (y). Для обучения самоорганизующейся карты Кохонена набор данных был разделен на обучающуюся и тестовую выборку. Тестовая выборка составила 20% данных от всего набора.

Далее создается модель SOM для обучения с указанными гиперпараметрами, такими как размер карты, количество входных признаков, параметр sigma=1,5, который определяет радиус вокруг нейрона-победителя, в пределах которого будет происходить обновление весовых коэффициентов всех нейронов, значение learning_rate=0,5 для скорости обучения и количество итераций или эпох обучения. При обучении самоорганизующихся карт Кохонена каждому нейрону на карте присваивается вектор весов, который выступает как центр кластера. Обучение происходит путем присваивания ближайшему нейрону вектора данных и обновления его весов, чтобы он стал еще ближе к входным данным. Для каждого входного вектора данных выполняется следующее:

– Вычисляется Евклидово расстояние между входным вектором и весовыми векторами каждого нейрона на карте SOM.

– Находится нейрон с наименьшим расстоянием (называемым нейрон-победитель) и нейроны вокруг него (выигравший квадрат).

– Веса нейрона с нейроном-победителем и выигравшего квадрата обновляются, чтобы быть ближе к входному вектору.

– Это повторяется в несколько эпох до тех пор, пока SOM не будет полностью сходиться и не сформирует кластеры.

На этапе обучения самоорганизующаяся карта Кохонена активизируется с помощью подаваемых на неё данных, чтобы она могла построить оптимальную топологическую карту эталонов. Обучение карты позволяет ей создать некоторую нормальную границу, представляющую собой закономерности и структуры данных.

На рис. 2 представлена визуализация результатов обучения нейронной сети типа SOM. На графике каждый квадрат представляет собой нейрон, а его цвет обозначает расстояние от этого нейрона до ближайшего образца данных. Чем темнее цвет, тем меньше расстояние до ближайшего образца данных.

Маркеры на графике отображают образцы данных, которые были переданы в обученную нейронную сеть. Маркеры разного вида и цвета представляют разные метки данных. Каждый маркер размещается на позиции победителя-нейрона, который наиболее близок к данному образцу данных. Таким образом, на графике можно наблюдать, как образцы данных группируются вокруг близких похожих нейронов. Кластеры образцов данных отображаются цветом и формой маркеров. В целом, график помогает визуализировать взаимосвязи между данными и нейронами.

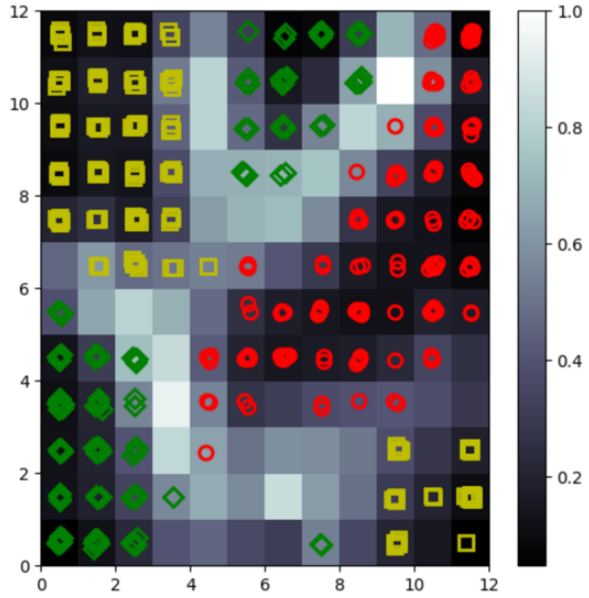


Рис. 2. График карты расстояний SOM (U-матрица)

После визуализации карты расстояний были вычислены две метрики: топографическая ошибка и ошибка квантования. Ошибка квантования возникает при применении самоорганизующейся карты Кохонена в процессе сжатия и кластеризации данных. Средняя ошибка квантования (Mean Quantization Error) является средним значением расстояний между входным образцом и его соответствующим весовым вектором на карте. В данном случае, значение ошибки квантования составила 0,45. Топографическая ошибка (Topographic Error) – это метрика, которая оценивает, насколько хорошо топология (отношения между соседними точками) сохраняется в преобразованном пространстве при квантизации данных. Значение равное 0,3 указывает на то, что приблизительно 30% нейронов на карте Кохонена не соблюдают топологическую структуру данных. Чем ближе значение ошибок к нулю, тем лучше соблюдается топологическая структура данных на карте Кохонена.

После обучения самоорганизующаяся карта Кохонена используется для классификации новых данных. Каждый новый объект проецируется на карту, и его сходство с каждым нейроном вычисляется и сравнивается с использованием метрики для выявления аномалий (threshold).

Обычно, для определения порогового значения выявления аномалий используются различные статистические методы, такие как методы на основе стандартного отклонения или квантилей.

Для вычисления порога выявления аномалий, в данном случае, используется среднее значение ошибок квантования np.mean(quantization_errors) и стандартное отклонение np.std(quantization_errors) этих ошибок. Сначала вычисляется среднее значение ошибок квантования, затем к нему добавляется стандартное отклонение. Таким образом, порог устанавливается на уровне среднего значения ошибок квантования плюс одно стандартное от-

клонение. Такой порог может быть использован для определения аномальных значений, которые имеют ошибку квантования, превышающую этот порог.

Если сходство объекта с нейроном больше заданного порога, считается, что объект является аномалией. Таким образом, самоорганизующаяся карта Кохонена выполняет функцию порогового детектора, определяющего, является ли новый объект нормальным или аномальным.

На рис. 3 пунктирная линия представляет порог, который составляет 0,65, выбранный для обозначения выбросов. Все значения, превышающие данный порог, будут относиться к аномальным. Видно, что большинство образцов имеют низкую ошибку квантования, а ошибки выше порога встречаются гораздо реже.

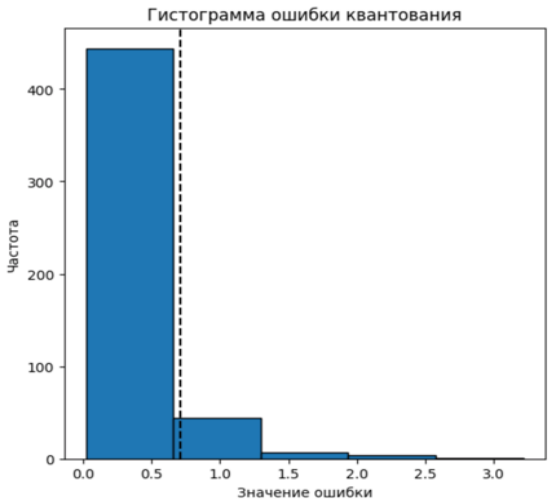


Рис. 3. Гистограмма определения порогового значения

На рис. 4 изображен график выделения аномальных точек данных с помощью синих звёздочек.

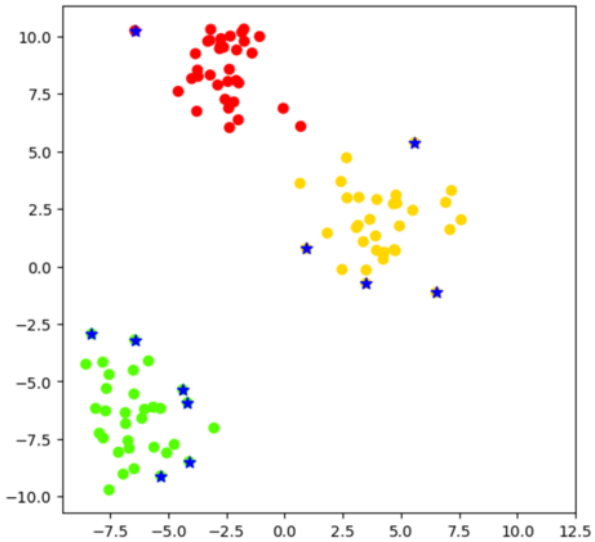


Рис. 4. График выделения аномальных точек данных

Это означает, что было обнаружено 11 аномальных образцов, и их индексы в массиве указаны в [4 9 25 29 36 44 45 52 78 81 84]. Аномальные индексы [4 9 25 29 36 44 45 52 78 81 84] представляют собой индексы точек данных в массиве, которые были классифицированы как аномалии на основе порогового значения threshold. Это означает, что точки данных с указанными индексами имеют более высокую ошибку квантования, чем пороговое значение. В данной работе аномальные индексы указывают на те точки данных, которые далеко отклоняются от среднего поведения или признаковых шаблонов, обнаруженных моделью SOM. Эти аномальные индексы могут быть полезны для дальнейшего анализа и исследования выбросов или необычного поведения в данных.

После обучения картой SOM на тестовых данных также был применен метод DBSCAN (Density-Based Spatial Clustering of Applications with Noise). DBSCAN использует плотностную основу, основываясь на близости точек данных, чтобы определить кластеры. Он ищет области высокой плотности, разделяя их от областей низкой плотности и шумовых точек. Аномалии могут быть точками, которые не принадлежат ни к одному кластеру или являются выбросами. На рис. 5 изображен график выделения аномальных точек на тестовых данных методом DBSCAN после обучения картой SOM.

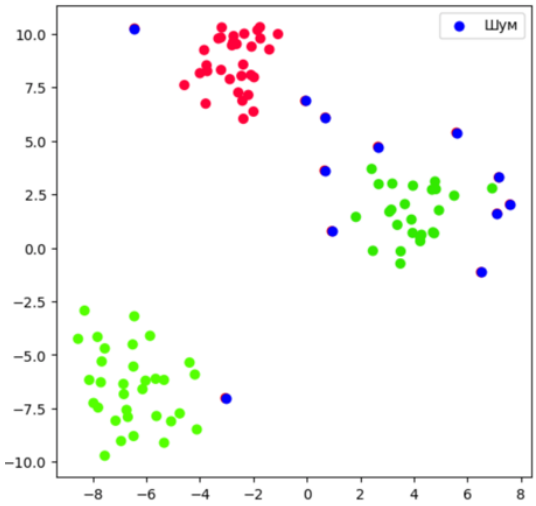


Рис. 5. График выделения аномальных точек данных методом DBSCAN после обучения картой SOM

На графике было выделено 3 кластера и 12 точек шума в виде синих точек. С определением кластеров метод справился, но определил на одну больше шумовую точку и если сравнить с рис. 4, то можно обратить внимание, что некоторые синие точки совпадают, а некоторые нет. Это говорит о том, что DBSCAN руководствуется немного другими параметрами для определения аномальных точек.

Если применить DBSCAN к тестовым данным без обучения картой SOM (рис. 6), то получим результат получения аномальных точек в количестве 7, что меньше значения, полученного ранее.

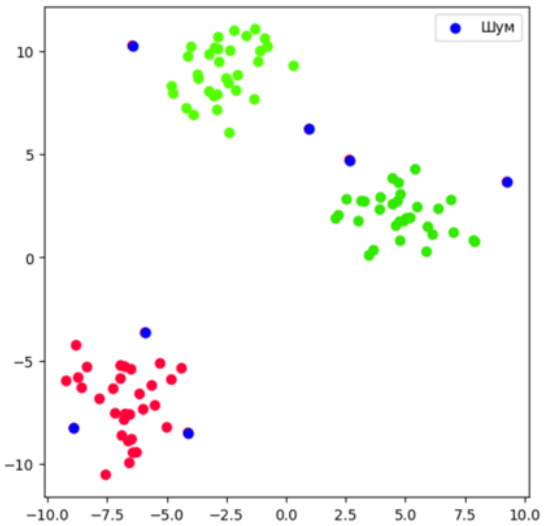


Рис. 6. График выделения аномальных точек данных методом DBSCAN

Сравнение метода самоорганизующейся карты Кохонена (SOM) с методом кластеризации DBSCAN в обнаружении аномалий может быть полезным для понимания и выбора подхода, наиболее соответствующего данным и поставленным задачам. Оба метода имеют свои преимущества и ограничения.

DBSCAN требует заранее заданного значения радиуса и минимального количества точек для формирования кластера. Это ограничивает его способность адаптироваться к изменениям в данных и обнаруживать ранее неизвестные аномалии.

SOM же адаптируется к структуре данных и может обнаруживать новые аномалии, не обучаясь заново. Кроме того, этот метод не требует предварительной разметки данных аномалий, что упрощает процесс развертывания системы обнаружения аномалий в реальных условиях. Однако он тоже требует тщательной настройки параметров и может быть требователен к ресурсам.

DBSCAN прост в реализации и может эффективно обнаруживать выбросы в данных, но он менее гибок в адаптации к изменениям и может быть ограничен в обработке больших наборов данных. Выбор метода зависит от специфики данных и поставленной задачи.

В заключение, использование самоорганизующейся карты Кохонена в качестве порогового детектора позволяет эффективно обнаруживать аномалии, вызванные кибератаками, в процессах автоматизированной системы управления технологическими процессами. Этот подход обладает рядом преимуществ, таких как адаптивность к изменениям данных и возможность работы без предварительной разметки аномалий. В дальнейшем исследовании можно улучшить этот метод и применить его на практике для повышения кибербезопасности АСУ ТП.

Библиографический список

1. Y. Dong, R. Wang and J. He, Real-Time Network Intrusion Detection System Based on Deep Learning, 2019 IEEE 10th International Conference on Software Engineering and Service Science (ICSESS), 2019, P. 1-4, doi: 10.1109/ICSESS47205.2019.9040718.

2. Farzan, Ali. Intrusion Detection System Using Self Organizing Map Algorithms. 2014. 3. 585.

3. Vita Santa Barletta, Danilo Caivano , Antonella Nannavecchia and Michele Scalera. A Kohonen SOM Architecture for Intrusion Detection on In-Vehicle Communication Networks. Appl. Sci. 2020, 10(15), 5062; https://doi.org/ 10.3390/app10155062.

4. Subarna Shakya, Bisho Raj Kaphle. Intrusion Detection System Using Back Propagation Algorithm and Compare its Performance with Self Organizing Map. Journal of Advanced College of Engineering and Management, Vol. 1, 2015. DOI: 10.3126/jacem.v1i0.14930.

5. Казаков Ф.А. Подходы к построению нейросетевого анализатора трафика / Ф.А. Казаков, А.В. Шнайдер // Вестник современных исследований. – 2021. № 1-6(39). – С. 15–17. – EDN JVYUZB.

6. Гамзаев Р.Г. Применение машинного обучения для поиска аномалий сетевого трафика с целью выявления и предупреждений инцидентов информационной безопасности / Р.Г. Гамзаев, Г.К. Масков, В.М. Моргунов // Методы и технические средства обеспечения безопасности информации. – 2023. № 32. – С. 129–131. – EDN AJSSAQ.

УДК 004.021

ПРИМЕНЕНИЕ КОНЦЕПЦИИ Q-ДЕТЕРМИНАНТА К МЕТОДАМ МИНИМИЗАЦИИ БУЛЕВОЙ ФУНКЦИИ, МОДЕЛИРУЮЩЕЙ ЗАВИСИМОСТИ В СИСТЕМАХ ПОДКЛЮЧЕННЫХ ТРАНСПОРТНЫХ СРЕДСТВ ПРИ ИССЛЕДОВАНИИ ИХ ЗАЩИЩЕННОСТИ

М.П. Соколов, Н.Д. Зюляркина
Научный руководитель: докт. физ.-мат. наук, доц. Н.Д. Зюляркина
Южно-Уральский государственный университет,
г. Челябинск

Задача минимизации булевой функции довольно часто возникает при исследовании различных информационных систем на предмет их защищенности, в частности систем подключенных транспортных средств. В статье рассмотрены методы минимизации Куайна и Куайна – Мак-Ласки. Изучена эффективность их распараллеливания на основе концепции Q-детерминанта. Было

показано, что метод Куайна допускает более эффективное распараллеливание.

Ключевые слова: информационная безопасность, подключенные транспортные средства, Q-детерминант алгоритма, Q-эффективная реализация алгоритма, Q-эффективная программа, параллельная программа, минимизация булевой функции, метод Куайна, метод Куайна – Мак-Ласки.

С увеличением количества цифровых устройств в современных транспортных средствах и появлением возможностей для связи между транспортным средством и внешним миром с использованием беспроводных интерфейсов возрос риск взлома подсистем транспортного средства, что может привести к авариям с серьезными последствиями. Чтобы предотвратить взлом, необходимо обеспечить достаточный уровень информационной безопасности в сети подключенного транспортного средства, в которой необходимо выявить наиболее уязвимые узлы сети [1].

В статье [2] предложен способ моделирования уязвимости узлов информационной сети транспортного средства для построения оптимальной конфигурации защитных средств, который заключается в применении математического моделирования с использованием графов И/ИЛИ, а также представлении набора защитных мер и узлов сети транспортного средства в виде гиперграфов. Операционные зависимости в подсистеме представлены в виде логических зависимостей с соединениями типа И/ИЛИ. Для оценки защищенности узла в модели подключенных транспортных средств составляется специальная функция алгебры логики, которая записывается в конъюнктивной нормальной форме. Чтобы оценить защищенность узла наиболее быстрым способом, формулу следует минимизировать. Для выявления наиболее уязвимых узлов в этих сетях довольно часто используется алгоритм Дейкстры и его модификации.

С увеличением количества узлов в сети подключенных транспортных средств возрастает вычислительная сложность алгоритма минимизации конъюнктивной нормальной формы, представляющей операционные зависимости сети. Также возрастает вычислительная сложность алгоритмов на графах, используемых для выявления наиболее уязвимых узлов. Отметим, что потребность минимизации функций алгебры логики возникает при решении многих задач, в частности, при минимизации цифровых устройств [3].

Для реализации таких алгоритмов целесообразно использовать параллельные вычисления. Использование ресурса внутреннего параллелизма алгоритмов с применением концепции Q-детерминанта позволяет достигнуть максимального распараллеливания любого алгоритма [4]. С помощью данного подхода можно для любого алгоритма, допускающего распараллеливание, определить его максимально параллельную реализацию.

Концепция Q-детерминанта дает возможность исследовать ресурс внутреннего параллелизма алгоритма, а именно: получать все возможные реализации алгоритма, в том числе Q-эффективную. В рамках данной концепции Q-эффективной называется реализация алгоритма, которая в полной мере использует ресурс параллелизма алгоритма. Такая реализация также носит название максимально быстрой реализации алгоритма, так как все операции выполняются тогда, когда они готовы к выполнению. К тому же представление алгоритма в форме Q-детерминанта позволяет проводить анализ характеристик ресурса параллелизма алгоритма: высоту и ширину, которые характеризуют число тактов работы и число процессоров вычислительной системы соответственно, необходимых для выполнения Q-эффективной реализации.

Одним из главных понятий концепции Q-детерминанта является представление алгоритма в форме Q-детерминанта. В основе данной концепции лежит преобразование последовательности действий алгоритма в Q-термы – выражения над множеством арифметических или логических переменных и множеством операций над этими переменными. Множество Q-термов алгоритма называется Q-детерминантом этого алгоритма. Пусть алгоритм имеет N выходных данных $y_i (i = 1, \dots, N)$, тогда общий вид представления этого алгоритма в форме Q-детерминанта: $y_i = f_i$ $(i = 1, \dots, N)$, где f_i – Q-терм, описывающий вычисление y_i. Вычисление полученных Q-термов и есть реализация алгоритма, представленного в форме Q-детерминанта. Q-эффективная реализация алгоритма заключается в том, что Q-термы вычисляются одновременно, а все входящие в них операции выполняются по мере готовности.

Метод проектирования параллельных программ, использующих концепцию Q-детерминанта, включает в себя: построение Q-детерминанта алгоритма, описание плана выполнения Q-эффективной реализации алгоритма, разработку программы для выполнения Q-эффективной реализации алгоритма, если она выполнима. Программа называется Q-эффективной, если она создается с использованием описанного метода проектирования параллельных программ.

Данная работа посвящена изучению Q-эффективной реализации алгоритма, реализующего минимизацию логической формулы, представляющей операционные зависимости в информационной сети подключенного транспортного средства. Исследование дополняет работы [5, 6], посвященные изучению Q-эффективной реализации алгоритма Дейкстры, который можно использовать для выявления наиболее уязвимых узлов в сети.

Существуют различные методы минимизации функции алгебры логики. Согласно концепции Q-детерминанта, из множества алгоритмов, решающих одну и ту же алгоритмическую проблему, большим ресурсом внутреннего параллелизма обладает алгоритм с большей шириной, а наименьшее время выполнения будет иметь алгоритм с меньшей высотой [4]. Ис-

ходя из этих критериев, необходимо выбрать оптимальный алгоритм, реализующий минимизацию логической формулы, для создания Q-эффективной программы. Для этого рассмотрим основные методы минимизации логической формулы (подробное описание этих методов и связанных с ними понятий можно найти в пособиях [7, 8]). Отметим, что минимизация конъюнктивной нормальной формы сводится к минимизации дизъюнктивной нормальной формы.

Методы минимизации логической формулы включают в себя два этапа: нахождение всех простых импликант и выбор минимального покрытия.

Карта Карно – графический способ представления булевых функций с целью их удобной и наглядной ручной минимизации. Специфика метода делает его не оптимальным с точки зрения параллельной программной реализации.

В методе Куайна простые импликанты получаются путем перемножения дизъюнкций в СКНФ, с последующим удалением повторяющихся и нулевых конъюнкций и применения операции поглощения.

Метод Куайна – Мак-Класки является усовершенствованным методом Куайна. Специфика метода Куайна – Мак-Класки по сравнению с методом Куайна в сокращении количества попарных сравнений. Сокращение достигается за счёт исходного разбиения термов на группы с равным количеством единиц (нулей). Это позволяет исключить сравнения, заведомо не дающие склеивания.

Аналитический анализ характеристик ресурса параллелизма алгоритмов, реализующих методы Куайна и Куайна – Мак-Класки, позволяет сделать вывод: так как усовершенствование метода Куайна – Мак-Класки заключается в сокращении количества попарных сравнений с помощью группировки, этот алгоритм имеет большую высоту и меньшую ширину за счет добавления в алгоритм группировки и сокращения количества операций соответственно. Следовательно, Q-эффективная программа, реализующая метод Куайна – Мак-Класки на параллельной вычислительной системе, будет использовать меньшее количество вычислителей и иметь большее время выполнения. Исходя из этого, согласно концепции Q-детерминанта, оптимальным алгоритмом для минимизации логической формулы является алгоритм, реализующий метод Куайна.

Второй этап минимизации логической формулы заключается в нахождении минимального покрытия носителя функции простыми импликантами. Пусть $f: Z_2^n \to Z_2$ – функция алгебры логики. Её носитель – это множество наборов из Z_2^n, на которых f принимает значение 1. Импликантная матрица функции f – это матрица, строки которой заиндексированы простыми импликантами функции f (или соответствующими им максимальными гранями носителя функции f), а столбцы заиндексированы наборами из носителя. Элемент a_{ij} этой матрицы равен 1, если набор j покрывается

гранью, соответствующей простой импликанте i. В противном случае $a_{ij} = 0$. Задачу нахождения минимального покрытия можно решить как с помощью перебора всех сочетаний импликант, так и с помощью различных методов, которые также включают в себя перебор потенциально большого, при большом количестве импликант, множества. Примером такого метода является метод Петрика. Из аналитического анализа характеристик ресурса параллелизма алгоритма поиска минимального числа импликант можно сделать вывод: алгоритм, реализующий перебор наборов импликант, начиная с минимального количества импликант в наборе, является алгоритмом с наименьшей высотой и наибольшей шириной, следовательно, согласно концепции Q-детерминанта, иметь наибольший ресурс параллелизма и наименьшее время выполнения Q-эффективной программы на параллельных вычислительных системах с достаточным количеством вычислителей. Очевидно, что количество возможных наборов импликант равно 2^n, где n – количество импликант в импликантной матрице.

Отметим, что аналитическая оценка эффективности распараллеливания программы для минимизации функции алгебры логики будет связана с размером импликантной матрицы.

Рассмотренные в статье методы на втором этапе идентичны, а на первом этапе имеют различия. При этом, метод Куайна является оптимальным выбором для применения параллельных вычислений, так как он имеет большую ширину и меньшую высоту на первом этапе.

Библиографический список

1. Barinov A., Davydkin N., Sharova D. and Skurlaev S., Prioritization methodology of computing assets for connected vehicles in security assessment purpose, in 2019 12th CMI Conference on Cybersecurity and Privacy (CMI), 2019, P. 1–6, DOI: 10.1109/CMI48017.2019.8962145.

2. Sheviakov I.A., Barinov A.E., Modeling the Vulnerabilities of Information Network Nodes of a Connected Vehicle Using AND/OR Graphs, 2022 Ural-Siberian Conference on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT), Yekaterinburg, Russian Federation, 2022, P. 313–316, DOI: 10.1109/USBEREIT56278.2022.9923368.

3. Смагин А.А., Шиготаров А.В. Применение методов минимизации булевых функций для оптимизации цифровых устройств, 2009, Ульяновский государственный университет, Ульяновск, 7 л.

4. Aleeva V.N., Aleev R.Z., Investigation and Implementation of Parallelism Resources of Numerical Algorithms, ACM Transactions on Parallel Computing, 2023, Vol. 10. №. 2.

5. Соколов М.П., Баринов А.Е., Зюляркина Н.Д. Использование концепции Q-детерминанта для распараллеливания вычислений при обеспечении информационной безопасности в системах подключенных транспортных средств, Безопасность информационного пространства, сборник научных трудов XXI Все-

российской научно-практической конференции студентов, аспирантов и молодых ученых, Екатеринбург, 2023. С. 188–190.

6. Sokolov M.P., Manatin P.A., Zyulyarkina N.D., The Parallelization of Computations for Ensuring Information Security in Connected Vehicle Systems Using Q-Effective Programming: The Example of Dijkstra's Algorithm, in 2023 International Russian Smart Industry Conference (SmartIndustryCon), 2023, DOI: 10.1109/SmartIndustryCon57312.2023.10110745.

7. Савельев А.Я. Основы информатики. – Москва: Издательство МГТУ им. Н.Э. Баумана, 2001.

8. Яблонский С.В. Введение в дискретную математику: учебное пособие для вузов. – 6-е изд. – М.: Высшая школа, 2010. – 384 с.

УДК 004

ОБНАРУЖЕНИЕ ПОПЫТОК ПОДБОРА ЛОГИНА С ПОМОЩЬЮ ИНСТРУМЕНТОВ МАШИННОГО ОБУЧЕНИЯ

М.М. Лихота
Научный руководитель: канд. техн. наук А.С. Коллеров
Уральский федеральный университет имени первого Президента России Б.Н. Ельцина, г. Екатеринбург

Исследованы события неудачных попыток авторизации пользователей в ОС Windows. Исходные данные проанализированы и размечены. Разработана методика обнаружения атаки подбора логина. Рассчитано сходство Джаро-Винклера и среднее расстояние Левеншнтейна. Обучена модель Sequmental с сетью LSTM для обнаружения попыток подбора логина. Рассчитаны основные метрики модели. Предложены сбособы улучшения результата.

Ключевые слова: подбор логина, машинное обучение, компьютерная атака, подбор пароля, учетные записи, пользователи, алгоритм обнаружения.

Считалось, что способ авторизации через пароль невозможно обойти, так как реквизиты для входа знает только «нужный» человек. Но с развитием информационных технологий и социальной инженерии злоумышленники научились перебирать или угадывать реквизиты пользователя. Если на информационном ресурсе есть поля для авторизации, и нет никаких ограничений на количество неуспешных попыток, то, только зная имя пользователя (login), пароль длинной менее, чем 5 символов возможно подобрать за несколько часов, если даже ни минут.

Существуют и обратные ситуации, когда злоумышленник понимает, что на информационном ресурсе с высокой вероятностью используются так называемые «слабые» пароли (пароли, которые легко подобрать или он имеет какой-то смысл в реальной жизни).

Атака методом перебора логина или так называемый PasswordSpraying заключается в том, что злоумышленник начинает процесс перебора уже с известным или «слабым» паролем. Согласно исследованию NordPass самыми популярными паролями являются «password», «123456», «123456789» [1]. Они могут быть использованы в первую очередь.

Атака методом перебора логина нацелена на системы, в которых количество пользователей измеряется сотнями и тысячами. К информационным ресурсам, которые имею такое количество пользователь, относятся вебсайты, контроллеры домена и другие объекты, на которых необходима авторизации.

В базе данных угроз ФСТЭК есть примеры действующих средств защиты, которые уязвимы к атакам таким методом, например, BDU:2021-04282, BDU:2021-02289, BDU:2021-02288, BDU:2021-01686, BDU:2020-03276, BDU:2020-03067, BDU:2020-01340 BDU:2020-0097 [2]. Реестр уязвимостей постоянно пополняется.

Также в международных базах уязвимостей (CVE (англ. Common Vulnerabilities and Exposures)) постоянно появляются новые уязвимости, связанные с атакой «грубой силой», например, CVE-2022-0828, CVE-2022-0652, CVE-2022-26519, CVE-2022-26314 [3].

Обнаружение атаки методом перебора логина не является тривиальной задачей. Обнаружение усложняет тот факт, что атаки могут быть продолжительными и кратковременными, по четкому словарю или смешанному. Целью данной статьи является показать, что методы машинного обучения помогут решить проблему обнаружения подобных атак.

В своей статье буду использовать обучение с учителем на уже размеченных данных. Предполагается решить задачу классификации и кластеризации данных для достижения удовлетворительных результатов. Результатом работы модели будут предсказания. Если по параметрам событие относится к цепочке перебора, то модель ставит «1», если нет – «0».

Исходными данными будут являться события Windows. События 4625 [4] из журнала Security отображают неудачные попытки входа в систему под конкретным пользователем. В самих событиях меня интересует только время и дата попытки и название учетной записи. Также можно использовать и другие поля, например источник попытки входа, станция, на которую пытались войти, код ошибки.

При разметке исходных данных цепочкой подборов будем считать три и более попытки входа подряд, в которых либо фигурируют похожие имена учетных записей (имеется схожесть в построении логина или используются одинаковые спецсимволы), либо подборы происходят за малый пери-

од времени (от 1 секунды до нескольких минут в зависимости от количества подборов).

Схожесть имен будем определять двумя способами. Первый способ, среднее расстояние Левенштейна, второй способ – сходство Джаро-Винклера. Эти способы помогут определить цепочку схожих логинов. Например, цепочку логинов user01, user02, user03. Для них сходство Джаро-Винклера и среднее расстояние Левеншнтейна будут максимальными.

Вторым важными признаком является временная разница между последовательными неудачными попытками. Необходимо вычислить среднюю временную разницу между предыдущими неудачными попытками. То есть, если у события X большая временная разница (например, более 10 минут) по сравнению с попытками входа Y, Z, U, W, которые произошли перед X, то значение параметра будет больше, чем в случаи, когда разница во времени, например, 10 секунд. Такой подход позволит определять подборы внутри одной цепочки, так как временная разница в них минимальна. Из минусов данного подхода можно выделить то, что такой параметр не может однозначно определить перебор, так как возможен случай, при котором несколько пользователей примерно в одно время неудачно авторизовались (например, в 8 утра).

Всего в реестре имеется 13753 записи. Мной было создано специальное поле «Подбор по словарю». Оно имеет два значения – «0» и «1». Если поле имеет значение «0», то это значит, что данное событие не относится к цепочке переборов логинов по уже указанным выше параметрам.

На рисунке (рис. 1) представлено начало таблицы. Здесь видно, что все логины не относятся к цепочке переборов, поэтому для них установлено значение «0».

A	D	E
Дата и время возникновени	**Пользователь**	**Подбор по словарю**
01.05.2010 2:48:24	economic	0
01.05.2010 3:48:34	economic	0
01.05.2010 4:21:50	oktshzt	0
01.05.2010 4:48:48	economic	0
01.05.2010 5:48:50	economic	0
01.05.2010 6:48:58	economic	0
01.05.2010 7:49:13	economic	0
01.05.2010 9:19:16	haustova-oa	0
01.05.2010 9:42:56	tu02	0

Рис. 1. Первые девять строк исходных данных

На рис. 2 представлен пример перебора логинов. Здесь четко видно, что имена пользователей имею общую структуру и время между переборами менее 10 секунд. Данная аномалия схожа с автоматизированным перебором логинов, поэтому для каждого имени пользователя в поле «Подбор по словарю» устанавливалось значение «1».

22.06.2010 0:24:52	ra_alex@example.ru	1
22.06.2010 0:25:03	ra_bykov@example.ru	1
22.06.2010 0:25:07	ra_gorod@example.ru	1
22.06.2010 0:25:08	ra_elan@example.ru	1
22.06.2010 0:25:08	ra_dubov@example.ru	1
22.06.2010 0:25:08	ra_danil@example.ru	1
22.06.2010 0:25:11	ra_cher@example.ru	1
22.06.2010 0:25:16	ra_kams_so@example.ru	1
22.06.2010 0:25:25	ra_ilov@example.ru	1
22.06.2010 0:25:25	ra_frol@example.ru	1
22.06.2010 0:25:25	ra_kalach@example.ru	1

Рис. 2. Пример цепочки переборов

Таким образом, данные проанализированы и готовы к предобработке. При нанесении разметки я также мог совершать ошибки, но считаем, что погрешность чрезвычайно мала.

После некоторых преобразований, а именно переименовании столбцов и проверки целостности данных, таблица имела вид указанный на рис. 3.

```
Int64Index: 13753 entries, 0 to 13752

Data columns (total 3 columns):

 #   Column     Non-Null Count  Dtype

---  ------     --------------  -----

 0   timestamp  13753 non-null  float64

 1   username   13753 non-null  object

 2   target     13753 non-null  int64

dtypes: float64(1), int64(1), object(1)

memory usage: 429.8+ KB
```

Рис. 3. Исходные поля и типы данных

Для обучения модели понадобятся дополнительные поля. Два поля связаны с именем пользователя, одно связано со временем. Чтобы оценить как конкретно имя пользователя связано с «соседними» имена создаю два новых значения – lvs (среднее расстояние Левенштейна) и jw (сходство Джаро-Винклера). «Соседними» именами являются некоторое количество любых X имен, которые во временном потоке находятся раньше, чем исходное имя. Для моей модели я выбрал значение X=5. В компьютерном коде данное значение присваивается полю BACKSHIFT (рис. 4).

```
BACKSHIFT = 5
```

```
from modules import relative_levenshtein_distance, jaro_winkler_similarity, pad_to_34, load_custom_ords
import re
import numpy as np

data['username'] = data.username.fillna('')
data['username'] = data.username.apply(lambda x: x.lower())
data['username'] = data.username.apply(lambda x: re.sub(r' ', r'', x))

weights = [x/10 for x in range(1, BACKSHIFT+1)]

for i in range(data.shape[0]):
    if i == 0:
        data.loc[data.index == 0, 'lvs'] = 0
        data.loc[data.index == 0, 'jw'] = 0

        continue

    start = max(0, i-BACKSHIFT)
    lvs_distances = list()
    jw_similarities = list()

    for j in range(start, i):
        lvs_distances.append(relative_levenshtein_distance(data.iloc[j].username, data.iloc[i].username))
        jw_similarities.append(jaro_winkler_similarity(data.iloc[j].username, data.iloc[i].username))

    data.loc[data.index == i, 'lvs'] = np.average(lvs_distances, weights=weights[-len(lvs_distances):])
    data.loc[data.index == i, 'jw'] = np.average(jw_similarities, weights=weights[-len(jw_similarities):])
```

Рис. 4. Вычисление среднего расстояния Левенштейна и сходства Джаро-Винклера

Для вычисления поля «time_delta» используется дата и время предыдущих пяти событий (BACKSHIFT=5), только используется вычисление средней разницы времени между предыдущими событиями (рис. 5).

```
from datetime import datetime

for i in range(data.shape[0]):
    if i == 0:
        data.loc[data.index == i, 'time_delta'] = 0

        continue

    start = max(0, i-BACKSHIFT)
    time_deltas = list()

    for j in range(start, i):
        time_deltas.append(data.iloc[i].timestamp - data.iloc[j].timestamp)

    mean_time_delta = np.average(time_deltas, weights=weights[-len(time_deltas):])

    data.loc[data.index == i, 'time_delta'] = mean_time_delta
```

Рис. 5. Дополнительное поле по времени

Итоговая таблица входных данных имеет вид, показанный на рис. 6.

```
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   timestamp   13753 non-null  float64
 1   username    13753 non-null  object
 2   target      13753 non-null  int64
 3   lvs         13753 non-null  float64
 4   jw          13753 non-null  float64
 5   time_delta  13753 non-null  float64
dtypes: float64(4), int64(1), object(1)
```

Рис. 6. Итоговая таблица входных данных

Перед началом обучения необходимо проверить полученные данные на наличие аномалий. Сначала оцениваю параметр time_delta. По графику видно, что среди всех значений поля «time_delta» абсолютно большинство лежит в диапазоне от 0 до 10000, что является приемлемым результатом (рис. 7).

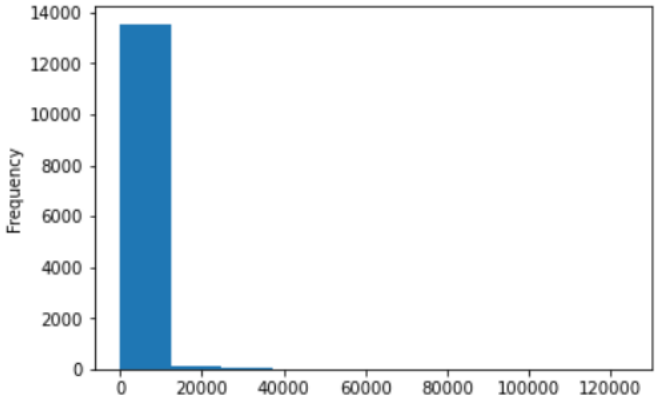


Рис. 7. Распределение по полю time_delta

Далее важно оценить количество исходных событий за определенные временные периоды. Для этого создал два массива. Первый назвал «neg», в нём все события, которые на этапе разметки были размечены как ложные (имя учетной записи не относится к подбору). Второй называется «pos». В нём находятся все события, которые относятся к подбору логина (рис. 8).

```
import matplotlib.pyplot as plt

data['date'] = data.timestamp.apply(lambda x: pd.to_datetime(x, unit='s').date())

fig = plt.figure(figsize=(15, 6))

idx = pd.date_range(data.date.min(), data.date.max())

neg = pd.Series({k: v for k, v in sorted(data.loc[data.target == 0].date.value_counts().items())})
neg.index = pd.DatetimeIndex(neg.index)
neg = neg.reindex(idx, fill_value=0)

pos = pd.Series({k: v for k, v in sorted(data.loc[data.target == 1].date.value_counts().items())})
pos.index = pd.DatetimeIndex(pos.index)
pos = pos.reindex(idx, fill_value=0)

neg_plot = plt.bar(range(0, data.date.nunique()), neg, width=0.35, label='Negative samples')
pos_plot = plt.bar(range(0, data.date.nunique()), pos, width=0.35, bottom=neg, label='Positive samples')
```

Рис. 8. Бинарное распределение событий

Результат анализа представлен на скриншоте ниже (рис. 9).

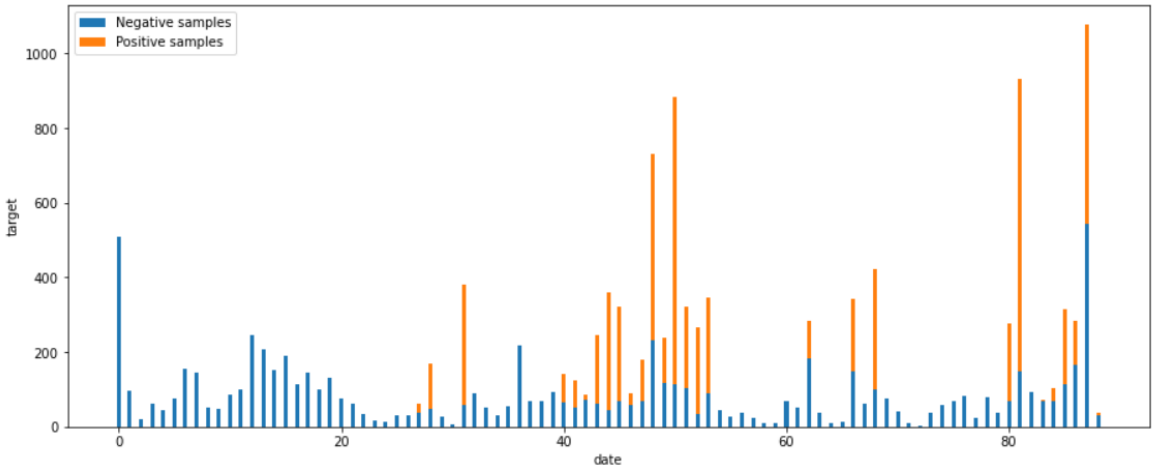


Рис. 9. График распределения событий по времени

Так как самое важно поле, с точки зрения результата работы модели, это поле «target», то мне необходимо оценить, как созданные мной поля («lvs», «jw», «time_delta») соотносятся с разметкой. У событий, которые относятся к цепочке подборов логина, имеют значение поля jw приблизительно 0.75, значение поля lvs около 0.4 и значение time_delta менее 500 (рис. 10).

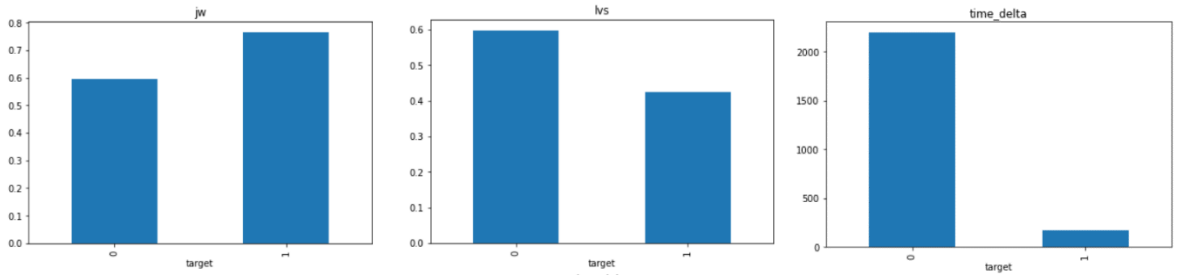


Рис. 10. Зависимость дополнительных метрик и разметки

Все предобработанные данные необходимо разделить на данные для обучения и тренировки (train – training) и данные для проверки (val – validation) (рис. 11). Основываясь на графике активности, был выбран объём данных для тренировки равный половине от общего датасета. Такая выборка позволит модели изучить как набор данных без аномалий, так и данные размеченные как подборы.

```
from sklearn.model_selection import train_test_split

X_train, X_val, y_train, y_val = train_test_split(X, y, test_size=0.5, shuffle=False)

assert all(np.hstack([y_train, y_val]) == y)
```

Рис. 11. Распределение данных

Для данной задачи лучше всего подойдет вид модели «Sequential». Также для настройки модели использована сеть с долгой краткосрочной памятью или LSTM – это разновидность рекуррентной нейронной сети (RNN), которая эффективна для прогнозирования длинных последовательностей данных за определенный период времени.

```
from tensorflow.keras.optimizers import Nadam
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense, Dropout
from tensorflow.keras.metrics import AUC
from tensorflow.keras.callbacks import EarlyStopping

BATCH_SIZE = 64

model = Sequential()

model.add(LSTM(32, return_sequences=True, input_shape=(X_train.shape[1], 1), activation='relu'))
model.add(LSTM(16, return_sequences=False, activation='relu'))

model.add(Dense(1, activation='sigmoid'))

early_stop = EarlyStopping(patience=5, restore_best_weights=True)

model.compile(optimizer=Nadam(clipnorm=1, clipvalue=0.5, learning_rate=0.0001), loss='binary_crossentropy', metrics=[AUC()])
model.summary()
```

Рис. 12. Обучение модели

Для оценки результатов работы модели была выбрана метрика «Точность отзыва» или же «Precision-recall». Точность (precision) – это мера релевантности результатов, а полнота (recall) – это мера того, сколько действительно релевантных результатов возвращается [5].

Так как целью работы модели является бинарная классификация, необходимо сразу разграничить классы:

– TruePositive (TP) (истинно положительные) – это вариант, при котором по разметке для определенного события установлено «1», и модель для этого же события предсказывает (predict) «1»;

– TrueNegative (TN) (истинно отрицательное) – это вариант, при котором по разметке для определенного события установлено «0», и модель для этого же события предсказывает (predict) «0»;

– FalsePositive (FP) (ложно положительное) – это вариант, при котором по разметке для определенного события установлено «1», и модель для этого же события предсказывает (predict) «0»;

– FalseNegative (FN) (ложно отрицательно) – это вариант, при котором по разметке для определенного события установлено «0», и модель для этого же события предсказывает (predict) «1».

По текущим результатам () можно определить, что:

– Метрика precision для «0» показывает отношение количества «0», которые определила модель, к количеству «0», которые были определены в разметке.

– Метрика recall для «0» показывает отношение количество «0», которые были определены в разметке, к количеству «0», которые определила модель.

– Метрика precision для «1» показывает отношение количества «1», которые определила модель, к количеству «1», которые были определены в разметке.

– Метрика recall для «1» показывает отношение количество «1», которые были определены в разметке, к количеству «1», которые определила модель.

Значение «accuracy» принимаем за итоговую точность первого варианта модели – 77% (рис.13). Данный результат выше ожидаемого и может считаться успешным, но проанализировав ошибки модели можно найти способы улучшить точность.

```
from sklearn.metrics import classification_report

print(np.unique(y_pred, return_counts=True))
print(classification_report(y_pred, y_val))
```

```
(array([0., 1.], dtype=float32), array([3114, 3763]))
              precision    recall  f1-score   support

         0.0       0.75      0.72      0.74      3114
         1.0       0.78      0.80      0.79      3763

    accuracy                           0.77      6877
```

Рис. 13. Результаты обучения модели

Для того, чтобы повысить точность модели необходимо в первую очередь определить, почему модель ошибается. В первую очередь необходимо изучить случаи, где разметка содержит значение «1», но модель предсказывает «0» (рис. 14).

	timestamp	username	targ	lvs	jw	time_de	predict
6896	2010-06-20 02:46:06	o_akulina	1	0,81952381	0,55005291	1309,1	0
6897	2010-06-20 02:46:06	o_antonova	1	0,717142857	0,596812169	428,8	0
6931	2010-06-20 03:01:45	o_bolgova@example.ru	1	0,579285714	0,648963166	283,2	0
6968	2010-06-20 03:17:35	o_bolgova@example.ru	1	0,688690476	0,610087535	331,6	0
7006	2010-06-20 05:51:29	m_arsenov	1	0,891111111	0,506560847	3636,3	0
7007	2010-06-20 05:52:13	m_gerok	1	0,716507937	0,519312169	1853,6	0
7008	2010-06-20 05:52:13	m_gurov	1	0,523809524	0,712063492	649,6	0
7018	2010-06-20 06:07:09	m_arsenov	1	0,750549451	0,503388278	572	0
7019	2010-06-20 06:07:45	m_gerok	1	0,698778999	0,595102768	343,3	0

Рис. 14. Анализ ошибок модели

Среди событий в реестре видно, что возникают цепочки событий, состоящие из 2–4 событий подряд. Такая ошибка связана с тем, что при вычислении параметров jw, lvs и time_delta модель задействует предыдущие 5 событий, и первые несколько событий в цепочке переборов отмечаются как «0». Данную ошибку можно частично исправить, запустив последовательность событий в обратном порядке, то есть начать анализировать датасет с конца. Повышение точности произойдет вследствие того, что модель для первых событий по времени рассчитает значения дополнительных полей относительно новых событий. То есть, если в цепочке X,Y,W,R,Z событий на данный момент модель ошибается в первых трёх (X,Y,W), но точно предсказывает следующие (R, Z и т.д.), то при обратно анализе, модель рассчитает дополнительные параметры для событий X,Y,W относительно событий R, Z и пометит события как истинные.

Внеся дополнения, указанные выше, удалось получить итоговые метрики, показывающие результаты на 2–3% выше, чем при первом результате (рис. 15). Особенно метрики увеличились для класса «1».

```
from sklearn.metrics import classification_report

print(np.unique(y_pred, return_counts=True))
print(classification_report(y_pred, y_val))
```

```
(array([0., 1.], dtype=float32), array([2906, 3971]))
              precision    recall  f1-score   support

         0.0       0.75      0.76      0.75      2906
         1.0       0.82      0.81      0.82      3971

    accuracy                           0.79      6877
```

Рис. 15. Итоговые результаты модели

В результате обучения модель научилась определять попытки подбора логина с точностью 79% на данных для валидации.

Заключение. На основании результатов тестирования можно сделать вывод, что модель достаточно точно определяет попытки перебора логина. Также можно сымитировать новые способы перебора и тем самым увеличить точность предсказания. Дополнительными полями для обнаружения перебора могут быть значения источника попытки (IP адрес) и причины неудачной авторизации (учетная запись отключена, неверный пароль или учетная запись заблокирована).

Обученную модель можно интегрировать в современные средства защиты. Она сможет обнаруживать атаки в реальном времени, что позволит специалистам информационной безопасности быстрее предпринять необходимые действия для устранения угрозы. Также такую модель можно использовать для расследования уже случившихся атак. Таким способом можно определить потенциальную «точку входа» злоумышленника в систему.

Библиографический список

1. Официальный сайт компании Nordpass. Статья «Самые популярные пароли». [Электронный ресурс] URL: https://nordpass.com/most-common-passwords-list/ (дата обращения: 25.10.2023).

2. Официальный сайт ФСТЭК. База данных уязвимостей. [Электронный ресурс] URL: https://bdu.fstec.ru/vul (дата обращения: 25.10.2023).

3. Официальный сайт международной компании Mitre. Раздел существующих уязвимостей. [Электронный ресурс] URL: https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=brute+force+rdp (дата обращения: 25.10.2023).

4. Официальный сайт Microsoft. Событие 4625 F: учетной записи не удалось войти в систему. [Электронный ресурс] URL: https://learn.microsoft.com/ru-ru/windows/security/threat-protection/auditing/event-4625 (дата обращения: 25.10.2023).

5. Документация продукта Scikit-Learn [Электронный ресурс] URL: https://scikit-learn.org/stable/auto_examples/model_selection/plot_precision_recall.html (дата обращения: 25.10.2023).

УДК 004.89+004.056

ИССЛЕДОВАНИЕ МЕТОДОВ И АЛГОРИТМОВ ОБНАРУЖЕНИЯ КОМПЬЮТЕРНЫХ ИНЦИДЕНТОВ

А.С. Грибачёв, В.В. Кальщиков
Научный руководитель: кандидат физ.-мат. наук, доц. А.Н. Ручай
Челябинский государственный университет,
г. Челябинск

В настоящее время серьезную угрозу для страны представляют кибератаки на объекты критической информационной инфраструктуры. Поэтому превентивная защита, своевременная реакция на данные виды атак, а также принятие мер по противодействию киберугрозам является приоритетной задачей. Цель данной работы состоит в изучении типов компьютерных угроз, методов их обнаружения и способов противодействия. В статье приводятся пути решения проблем выявления киберугроз.

Ключевые слова: информационная безопасность, кибербезопасность, критическая информационная инфраструктура, машинное обучение, обнаружение атак.

Одной из основных задач государства является обеспечение национальной безопасности [1]. Кибератаки в последнее время стали одной из основных проблем для национальной безопасности, а защита компьютерных систем от таких атак приоритетом с номером один [2]. Активное внедрение технологии интернета вещей в жизнь современного человека, цифровизации различных бизнес-процессов учреждений и предприятий, все это упрощает и облегчает существование и функционирование в различных сферах деятельности общества: медицина, сельское хозяйство, экономика и многие другие. Однако, это также является благоприятной средой для киберпреступности, что подтверждается каждодневным ростом кибератак на объекты критической информационной инфраструктуры. Кибератака – это любое действие, связанное с незаконным проникновением в компьютерную систему, путем обхода системы защиты. Выявление кибератаки – обнаружение факта неавторизованного доступа в систему [3]. Статья посвящена кибератакам и методам их обнаружения, а также представлен альтернативный способ построения системы обнаружения вторжения.

Действия киберпреступников направлены на получение несанкционированного доступа к информационным системам и каналам связи инфраструктуры для перехвата управлением, либо получения данных, имеющих определенную ценность [4]. Такие действия подразделяются на два вида: целевые и распределенные кибератаки. Одни нацелены на определенную

компанию или отрасль, такой вид атаки подразумевает получение доступа к ресурсам инфраструктуры и минимизация риска обнаружения киберпреступника в системе, то есть злоумышленник может находиться в сети долгое время, до момента его обнаружения. Для реализации такого типа атак необходимы автоматизированные инструменты и высокоспециализированные хакеры. Другие направлены на огромное количество информационных систем компаний, для таких атак применяются специальные роботизированные сети.

Существуют различные типы атак на критическую информационную инфраструктуру:

1. Боты. Происходит имитация поведения человека, однако робот выполняет задачи быстрее самого пользователя.

2. Атака «грубой силой» – представляет собой метод получения доступа в информационную систему путем взлома учетных записей.

3. Отказ в обслуживании - эти атаки нагружают систему большим количеством запросов, в результате происходит снижение пропускной способности, а система недоступной.

4. Фишинг – использование почтовых рассылок, замаскированных под обычные сообщения компании.

5. Атака через посредника. При таком типе атаки в момент передачи сообщений происходит утечка данных третьей стороне [5-9].

Современные системы обнаружения кибератак основываются на сборе данных о трафике сетевых соединений и журналируемых событиях серверов и ключевых компьютеров. Такие системы проводят наблюдение и анализируют события, которые происходят в информационной инфраструктуре, а также позволяют отслеживать различные сетевые атаки, такие как проникновение в сеть, отказ в обслуживании и сканирование портов. Анализ различных системных характеристик, либо отслеживание входящего/исходящего трафика позволяет обнаружить вредоносные действия [10-11].

Методы обнаружения компьютерных угроз можно разделить на следующие блоки:

- Сигнатурный анализ базируется на рассмотрении содержимого исследуемого объекта сигнатур уже известных угроз. Сигнатура атаки – характерные признаки угрозы, которые используются в целях ее обнаружения. Сравнение содержимого исследуемого объекта заключается в сравнении по контрольным суммам. Данный способ существенно снижает размер записей в базах и позволяет сохранить корректность обнаружения угроз.
- Метод эмуляции исполнения. Применяется для обнаружения полиморфных и шифрованных вирусов. Заключается в применении специальной программной модели процессора и среды исполнения программ (эмулятора). Эмулятор оперирует с защищенной областью памяти (буфером

эмуляции). При этом инструкции не передаются на центральный процессор для реального исполнения.

- Эвристический анализ основывается на наборе эвристик о характерных признаках вредоносного и, наоборот, безопасного исполняемого кода. Каждый признак кода имеет определенный вес. На основании суммарного веса, характеризующего содержимое объекта, эвристический анализатор вычисляет вероятность содержания в нем неизвестного вредоносного объекта. Если эта вероятность превышает некоторое пороговое значение, то выдается заключение о том, что анализируемый объект является вредоносным.
- Метод поведенческого анализа позволяют анализировать последовательность действий всех процессов в системе. При обнаружении признаков поведения вредоносной программы действия приложения блокируются.
- Метод машинного обучения. Данный метод используется для обнаружения угроз, которые отсутствуют в вирусных базах. Преимущество метода – распознавание угроз на основе их характеристик. Основывается на классификации кибератак согласно определенным признакам. Метод машинного обучения позволяет экономить ресурсы операционной системы, так как не требует исполнения кода для выявления угроз [12].

В качестве пути решения проблем с обнаружением киберугроз, предлагается разработка альтернативного способа построения системы обнаружения вторжения, который может позволить улучшить эффективность обнаружения кибератак.

Авторами разрабатывается система обнаружения компьютерных атак на основе машинного обучения. Для достижения поставленной цели произведен выбор набора данных для обучения системы обнаружения компьютерных атак, таким набором стала база данных компьютерных угроз на реальных объектах критической информационной инфраструктуры.

Предварительно проведена обработка данных, оценена значимость и отобраны наиболее значимые признаки. После определения рабочей модели будет произведена ее настройка и обучение, а также последующее тестирование и апробация результатов.

Заключение. В данной работе рассмотрены различные типы компьютерных угроз, а также методы их обнаружения. Важными составляющими методов обнаружения аномалий поведения информационных систем являются анализ последовательности действий всех процессов и классификация угроз по определенным характеристикам и признакам. Для повышения эффективности важно достижение оптимальных значений достоверности, точности и снижения времени принятия решений, что возможно лишь с применением глубокого машинного обучения. Одним из возможных путей решения проблемы обнаружения угроз - применение систематизации дан-

ных угроз, что позволит понять технологию обнаружения этих атак и разработать метод их обнаружения.

Библиографический список

1. Доктрина информационной безопасности Российской Федерации. Указ Президента Российской Федерации от 5 декабря 2016 г. № 646.

2. Хлопов О.А. «Проблемы кибербезопасности и защиты критической информационной инфраструктуры». Political Sciences. The scientific heritage No 45 (2020).

3. Серёдкин С.П. Особенности кибератак на объекты критической информационной инфраструктуры в современных условиях // «Информационные технологии и математическое моделирование в управлении сложными системами»: электрон. науч. журн. – 2022. – №4(16). – С. 56–66. – DOI: 10.26731/2658-3704.2022.4(16). – Режим доступа: http://ismm-irgups.ru/toma/416-2022, свободный. – Загл. с экрана. – Яз. рус., англ. (дата обращения: 23.10.2023).

4. Шаньгин В.Ф. Защита информации в компьютерных системах и сетях. / В.Ф. Шаньгин, Москва: ДМК Пресс, 2012. – 592 с.: ил.

5. Путято М.М., Евглевский В.Ю., Макарян А.С., Володин И.В. Исследование механизмов социальной инженерии и анализ методов противодействия // Научные труды КубГТУ, 2021, № 2. С. 57–68.

6. Афанасьева Н.С., Елизаров Д.А., Мызникова Т.А. Классификация фишинговых атак и меры противодействия им. Инженерный вестник Дона, №5 (2022) ivdon.ru/ru/magazine/archive/n5y2022/7641.

7. Баженов А.С. Обзор DDoS атак на IoT устройства // Наука настоящего и будущего. 2019. Т. 1. С. 122–125.

8. Савченко Е.В., Ниссенбаум О.В. Ботнет-атаки на устройства интернета вещей // Математическое и информационное моделирование. сборник научных трудов, электронный ресурс. Тюмень, 2018. С. 347–356.

9. Ручай А.Н., Токарев И.В., Грибачёв А.С. Методы машинного обучения и искусственного интеллекта в сфере информационной безопасности: анализ современного состояния и перспективы развития. Вестник УрФО № 4(46) / 2022, С. 76–87. DOI: 10.14529/secur220409.

10. Крейсат А., Гондал И., Вамплью П. и др. Обзор систем обнаружения вторжений: методы, наборы данных и проблемы. Кибербезопасность 2, 20 (2019). https://doi.org/10.1186/s42400-019-0038-7

11. Rafal Kozik, Michal Choraś, Rafal Renk, Witold Holubowicz. A Proposal of Algorithm for Web Applications Cyber Attack Detection. 13th IFIP International Conference on Computer Information Systems and Industrial Management (CISIM), Nov 2014, Ho Chi Minh City, Vietnam. P.680–687, ff10.1007/978-3-662-45237-0_61ff. ffhal-01405662.

12. Ван Г., Хао Дж., Ма Дж. и Хуанг Л. Новый подход к обнаружению вторжений с использованием искусственных нейронных сетей и нечеткой кластеризации, Приложение Expert Syst, Т. 37, № 9, С. 6225–6232, 2010/09/01/ 2010

13. Методы обнаружения угроз. // URL: https://cdn-download.drweb.com/pub/drweb/windows/server/12.0/documentation/html/ru/index.html?intro_detectionmethods.html (дата обращения: 10.10.2023).

УДК 004.891+ 004.912+ 007.51

ИСПОЛЬЗОВАНИЕ ГЕНЕРАТИВНЫХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ СОЗДАНИЯ КОДА ВСПОМОГАТЕЛЬНЫХ МОДУЛЕЙ СРЕДСТВ ЗАЩИТЫ ИНФОРМАЦИИ

А.А. Аверьянов
Научный руководитель: доц. В.Ю. Бердюгин
Южно-Уральский государственный университет,
г. Челябинск

В статье рассматриваются примеры применения больших языковых моделей для разработки технических и организационных мер защиты информации. Представлена теоретическая справка о генеративных нейронных сетях, объясняется, как они функционируют и что входит в их состав. Особое внимание уделяется использованию этих моделей для автоматизации разработки программного обеспечения, например, в написании кода для усиления кибербезопасности, включая алгоритмы шифрования. Кроме того, статья содержит графические иллюстрации, демонстрирующие влияние использования нейросетей на процесс разработки кода для защиты информации, а именно улучшение эффективности и сокращение времени разработки.

Ключевые слова: безопасность, большие языковые модели, генеративные нейросети, защита информации, информация, организационная защита, программный код.

Современные модели генеративных нейросетей, в частности, большие языковые модели, открывают новые горизонты в области защиты информации. Они не только способны генерировать код, повышающий безопасность программных продуктов, но и могут быть использованы при разработке и анализе организационных мер защиты. Эти модели способствуют оптимизации процессов путем создания настроенных политик безопасности, руководств по эксплуатации систем, а также документов, регламентирующих реакцию на инциденты и управление ими. Важно отметить, что применение нейросетей не ограничивается лишь автоматизацией существующих задач, но и включает в себя идентификацию уязвимостей и предсказание потенциальных угроз, что становится возможным благодаря способности моделей обрабатывать и анализировать большие объемы дан-

ных. Таким образом, Цель данной статьи – демонстрация того, как нейросети могут улучшить процессы разработки программных систем защиты от угроз и предсказания потенциальных уязвимостей, благодаря их способности генерировать безопасный код. А также показать, что системы защиты на основе нейронных сетей будут незаменимы для выявления аномалий в большом количестве событий информационной безопасности [1].

Теоретическая справка. Генеративные нейросети относятся к категории алгоритмов и моделей машинного обучения, которые способны создавать новый контент: текст, изображения, музыку, речь или другие данные, схожие с данными, на которых они были обучены. Основная черта генеративных моделей – их способность выявлять и изучать закономерности и структуры в исходных данных, чтобы затем генерировать новые данные, сохраняя при этом признаки оригинала.

Примеры генеративных моделей включают:

1. Генеративно-состязательные сети (Generative Adversarial Networks, GANs): это пара нейронных сетей – генератор, который создает данные, и дискриминатор, который оценивает их. Они «соперничают» друг с другом, что позволяет создавать высококачественные, реалистичные данные.

2. Вариационные автокодировщики (Variational Autoencoders, VAEs): это тип нейронных сетей, который сжимает данные в меньшее представление, а затем восстанавливает их, позволяя генерировать новые данные, которые похожи на исходные.

3. Модели Transformer (Generative Pre-trained Transformer): они обучаются на огромных корпусах текстов и могут генерировать согласованный и связанный текст, который подчас бывает неотличим от текста, написанного человеком.

Модели типа Transformer как раз и называются «Большие языковые модели» (Large Language Models, далее – LLM). Они учатся понимать и использовать естественный язык, а также генерировать связные и согласованные тексты на различные темы и в разных стилях. В результате они могут выполнять широкий спектр задач: от написания статей и создания шаблонов документов, до ответов на вопросы и перевода текстов. Более того, подобные модели могут генерировать рабочий программный код, который можно использовать для целей программной защиты информации.

Применение LLM для создания кода защиты. Использование генеративных нейросетей для написания программного кода защиты информации, является передовым направлением в области кибербезопасности. Из информации выше понятно, что генеративные модели, как правило, наиболее известны своей способностью создавать контент, но они также могут быть обучены для генерации кода, который используется для разработки систем безопасности.

При создании программной системы защиты информации, в первую очередь, создаются правила для систем обнаружения вторжений (IDS), а

также правила для обнаружения атаки SQL-инъекции. Благодаря использованию LLM мы можем создать их автоматически, сформировав запрос в языковую модель.

После отправки запроса в нейросеть для написания этих правил, в течение 10–20 секунд, получим следующий результат:

Правило для обнаружения сканирования портов

alert tcp any any -> $HOME_NET any (msg:"Подозрительное сканирование портов"; flags:S; threshold: type both, track by_src, count 5, seconds 60; classtype: attempted-recon; sid:1000001; rev:1;)

В этом правиле:

- alert tcp указывает, что правило касается трафика TCP.
- any any -> $HOME_NET any означает, что правило применяется к любому трафику TCP, направленному к любому порту в защищаемой сети.
- flags:S фильтрует пакеты с флагом SYN, что является признаком сканирования портов.
- threshold определяет порог, при котором генерируется предупреждение.
- classtype категоризирует тип атаки, attempted-recon означает попытку разведки.
- sid задает уникальный идентификатор правила.
- rev указывает номер версии правила.

Правило для обнаружения атаки SQL-инъекции

alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS $HTTP_PORTS (msg:"Возможная SQL-инъекция"; flow:to_server,established; content:"select"; nocase; content:"from"; distance:0; nocase; content:"where"; distance:0; nocase; pcre:"/(\%27)|(\')|(\-\-)|(\%23)|(#)/i"; classtype:web-application-attack; sid:1000002; rev:1;)

В этом правиле:

- Обрабатывается трафик, направленный к HTTP-серверам через соответствующие порты.
- content позволяет идентифицировать ключевые слова SQL-запросов в пакетах данных.
- nocase указывает на игнорирование регистра букв.
- pcre позволяет использовать регулярные выражения для более точного обнаружения атак SQL-инъекции.
- classtype категоризирует тип атаки, в данном случае – атака на веб-приложение.

Эти примеры показывают то, что LLM может сильно упростить создание базовых систем обнаружения вторжений. Для ручного написания кода из примеров выше, разработчик должен иметь углубленные знания в сетевых протоколах, базах данных, SQL и техниках кибербезопасности, а также умение работать с инструментами мониторинга сети и баз данных и

знание нескольких языков. Обучение для достижения такого уровня экспертизы может занять месяцы или даже годы.

В контрасте с этим, использование нейросетей позволяет ускорить процесс разработки. LLM могут быстро генерировать начальные версии программного кода, основываясь на обучающих данных и инструкциях пользователя. Это особенно полезно для стандартных задач, таких как обнаружение уязвимостей или мониторинг сетевого трафика. Пользователям этих систем, не обладающим глубокими знаниями в программировании, достаточно основного понимания задачи и умения взаимодействовать с LLM, чтобы начать работу.

Сравнение возможностей при разработке систем безопасности без использования нейросетей и с их использованием представлено на рис. 1. На нем мы можем увидеть, как меняется время, затрачиваемое на разработку, а также какая разница в требованиях к знаниям при этом подходе. Данный график основан на исследовании об эффективности использования нейросетей при разработке систем безопасности, проведенном на более чем 58 разработчиков разного уровня [2].

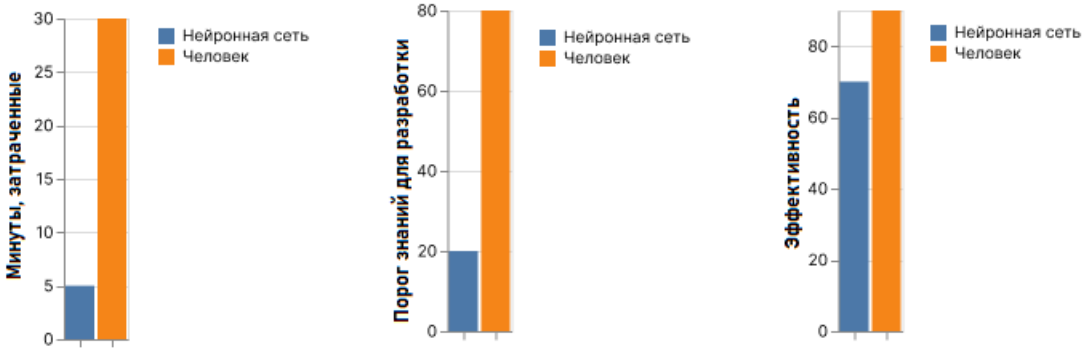


Рис. 1. Визуализация результатов разработки систем безопасности с использованием LLM

Разберем еще несколько примеров, чтобы доказать эффективность и вариативность данных систем, ведь автоматизация и простота использования – это единственные пути, которые смогут улучшить безопасность, и генеративные нейросети определенно займут весомую долю в этой сфере [3].

Генерация кода для шифрования и криптографических операций. Генеративные нейросети могут быть использованы для создания кода, который выполняет криптографические операции, такие как шифрование и дешифрование данных. Это может быть полезно для автоматизации процесса разработки безопасного программного обеспечения. Отправим запрос в LLM для создания алгоритма симметричного шифрования, который мы сможем применить в системе защиты.

На выходе получаем следующий код:

```
from Crypto.Cipher import AES
```

```
from Crypto.Random import get_random_bytes
from Crypto.Protocol.KDF import PBKDF2
from Crypto.Util.Padding import pad, unpad
import os
# Функция для создания ключа шифрования
def create_aes_key(password: str, salt: bytes):
    kdf = PBKDF2(password, salt, dkLen=32)  # AES ключ длиной 256 бит
    return kdf
# Функция для шифрования сообщения
def aes_encrypt(message: str, key: bytes):
    cipher = AES.new(key, AES.MODE_CBC)
    ct_bytes = cipher.encrypt(pad(message.encode('utf-8'), AES.block_size))
    return cipher.iv + ct_bytes  # Возвращаем IV + шифротекст
# Функция для расшифрования сообщения
def aes_decrypt(encrypted_message: bytes, key: bytes):
    iv = encrypted_message[:AES.block_size]  # извлекаем IV
    cipher = AES.new(key, AES.MODE_CBC, iv)
    pt = unpad(cipher.decrypt(encrypted_message[AES.block_size:]),
AES.block_size)
    return pt.decode('utf-8')
# Пример использования
password = 'секретный_пароль'
salt = get_random_bytes(16) # Соль для KDF
# Создание ключа
aes_key = create_aes_key(password, salt)
# Текст для шифрования
message = "Секретное сообщение"
# Шифрование
encrypted = aes_encrypt(message, aes_key)
print("Зашифрованное сообщение:", encrypted)
# Расшифровка
decrypted = aes_decrypt(encrypted, aes_key)
print("Расшифрованное сообщение:", decrypted)
```

В данном коде процесс шифрования и расшифровки осуществляется с использованием алгоритма AES, симметричного метода шифрования, где один и тот же ключ применяется как для шифрования, так и для расшифровки данных. Пользовательский пароль, в сочетании с криптографически безопасной солью, преобразуется в ключ шифрования с помощью функции PBKDF2, увеличивая тем самым защиту от атак по словарю и повышая стойкость ключа. Этот ключ затем используется для шифрования и последующей расшифровки сообщения, обеспечивая безопасный обмен данными.

Таким образом, на примере мы смогли еще раз убедиться, что LLM могут помочь не только с созданием базовых систем безопасности, но и с использованием сложных алгоритмов шифрования.

Стоит отметить, что LLM так же могут помочь в автоматизация создания политик безопасности, а также тестирование безопасности кода и его улучшение. В целом, гибкость нейросетей позволяет адаптировать их под любые задачи, что позволит специалистам тратить меньше времени на создание систем защиты безопасности, а также использовать передовые методы и технологии.

На рис. 2 представлен график эффективности специалиста по безопасности до и после использования LLM при разработке программ защиты. Исследования показывают, что в среднем эффективность разработки в этом случае вырастает на 23% на старте и становится тем больше, чем сильнее специалист привыкает к нейросети [2]. На временной точке 3 рис. 2 показан момент начала использования LLM в работе специалиста. Мы видим, что сразу после начала применения нейросети, резко возросла эффективность, которая становилась все больше и далее.

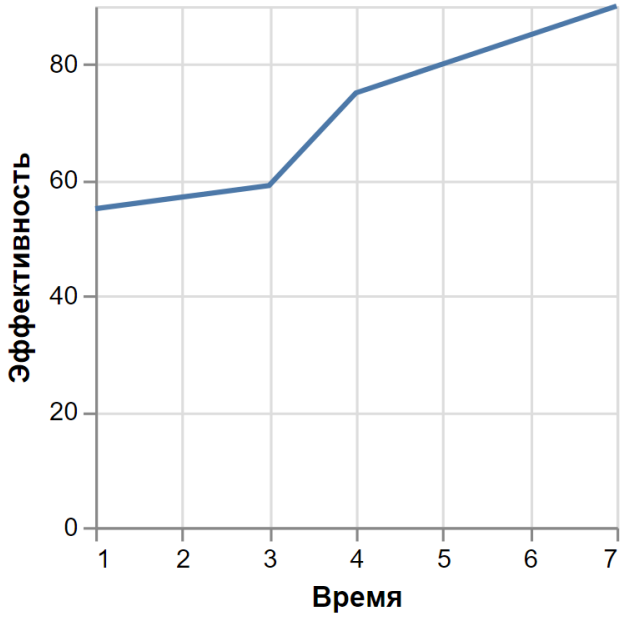


Рис. 2. Увеличение эффективности специалиста при разработке с использованием LLM

Заключение. В статье подчеркивается значительная роль больших языковых моделей (LLM) для генерации кода в информационной безопасности. Эти модели обладают способностью к обучению и адаптации, что делает их эффективными в разработке программного обеспечения для кибербезопасности. Они способствуют уменьшению времени и ресурсов для мониторинга киберугроз, написания программ, сокращают вероятность ошибок в коде и улучшают анализ данных.

Статья также акцентирует внимание на стратегической значимости интеграции LLM в защитные механизмы организаций. Применение этих тех-

нологий облегчает процессы разработки и документации в области кибербезопасности, делая их доступнее для широкого круга специалистов. Ожидается, что в ближайшем будущем мы можем ожидать еще большего расширения границ возможностей, предоставляемых нейросетями в борьбе с киберугрозами и в обеспечении информационной безопасности [4].

Библиографический список

1. Искусственный интеллект в информационной безопасности [Электронный ресурс]. – М.: Информационная безопасность, 2022. – Режим доступа: https://www.securityvision.ru/blog/iskusstvennyy-intellekt-v-informatsionnoy-bezopasnosti/, свободный – Загл. с экрана. (дата обращения: 16.10.2023).

2. Сандовал, Г., Пирс, Х., Нис, Т., Карри, Р., Долан-Гавитт, Б., Гарг, С. (2022). Влияние использования крупномасштабных языковых моделей-помощников на безопасность кода: Исследование среди пользователей. [Электронный ресурс]. – Режим доступа: https://ar5iv.labs.arxiv.org/html/2208.09727v1, свободный – Загл. с экрана. (дата обращения: 16.10.2023).

3. Палмер А. Искусственный интеллект трансформирует информационную безопасность [Электронный ресурс]. – Режим доступа: https://www.iksmedia.ru/articles/5682996-Iskusstvennyj-intellekt-v-informaci.html, свободный – Загл. с экрана. (дата обращения: 18.11.2023).

4. Будущее искусственного интеллекта в сфере безопасности [Электронный ресурс] – Режим доступа: https://www.secuteck.ru/articles/budushchee-iskusstvennogo-intellekta-v-sfere-bezopasnosti, свободный. – Загл. с экрана. (дата обращения: 21.11.2023).

УДК 004.896 + 004.056.5

ОПРЕДЕЛЕНИЕ ОПТИМАЛЬНОГО СПОСОБА АНСАМБЛИРОВАНИЯ МЕТОДОВ КЛАСТЕРИЗАЦИИ В ДВУХЭТАПНОЙ ПОВЕДЕНЧЕСКОЙ МОДЕЛИ ИНФОРМАЦИОННЫХ ПРОЦЕССОВ АСУ ТП, НАХОДЯЩЕЙСЯ В УСЛОВИЯХ ВОЗДЕЙСТВИЯ КИБЕРАТАК[1]

Д.А. Бухарев, А.Н. Соколов
Научный руководитель: канд. техн. наук, доц. А.Н. Соколов
Южно-Уральский государственный университет, г. Челябинск

Исследования в области кибербезопасности промышленных сетей выявили устойчивый рост кибератак на сфере промышленности. Большой объем обрабатываемой информации, требующий

[1] Исследование поддержано грантом Российского научного фонда (проект № 22-71-10095).

специальных инструментов для сбора и анализа, зачастую не позволяет оперативно обнаружить инциденты информационной безопасности. Для обнаружения кибератак в системах управления технологическими процессами (АСУ ТП) разработана двухэтапная поведенческая модель. На первом этапе модель автоматически определяет оптимальное число кластеров для агрегации нормальных и аномальных данных. На втором этапе используются методы кластерного анализа для выявления аномалий в данных. Эффективность модели подтверждена тестированием на сигналах с водоочистного полигона Secure Water Treatment, поддерживаемого iTrust.

Ключевые слова: информационная безопасность, иерархическая кластеризация, автоматизированная система управления технологическим процессом, кибератака, обнаружение аномалий.

На сегодняшний день промышленное производство в значительной степени опирается на системы автоматизированного управления технологическими процессами (АСУ ТП). Современные АСУ ТП осуществляют контроль над физическими устройствами, процессами и событиями, отражающими деятельность промышленных предприятий. Это улучшает эффективность управления и функционирование предприятий, однако такие системы подвержены кибератакам как внутри промышленной сети, так и вне ее.

В последние годы было зарегистрировано множество вредоносных атак, направленных на АСУ ТП промышленных предприятий, включая такие известные случаи, как Stuxnet [1], Triton [2] и BlackEnergy [3]. Нарушение нормального функционирования систем АСУ ТП может вызвать разрушительные последствия. Поэтому весьма актуальной является необходимость проведения исследований и разработки новых методов обеспечения киберфизической безопасности с целью защиты подобных систем.

Современные методы интеллектуальной обработки данных позволяют проводить анализ обширных объемов информации, характерных для промышленных сетей, и предоставлять точные решения с целью разработки неотложных мер для защиты сетей передачи данных на промышленных предприятиях от потенциально вредоносных информационных воздействий [4]. При помощи использования иерархического кластерного анализа [5–7] возможно выявление кластеров, содержащих в себе аномальные состояния технологического процесса, во время которых происходят кибератаки. Дальнейшее применение методов классификации разделяет полученные кластеры на аномальные и нормальные группы.

Предложен двухэтапный гибридный метод, объединяющий разнообразные подходы к кластеризации неразмеченных данных, с последующим сравнительным анализом эффективности различных методов кластерного анализа в контексте обнаружения аномалий в данных. Таким образом, был

выбран наиболее оптимальный метод кластеризации для обнаружения аномалий в неразмеченных данных.

Целью данной работы является определение оптимального способа ансамблирования выбранных ранее методов (метод изолированного леса [8], LOF [9], ECOD [10], OPTICS [11], COPOD [12]) при использовании предлагаемого двухэтапного метода для наиболее точного выделения аномальных и нормальных событий.

На первом этапе данные подвергаются агломеративному иерархическому кластерному анализу. Данный вид кластерного анализа отличается от других методов тем, что в результате работы алгоритма иерархической кластеризации мы получаем график дендрограммы, представленный в форме дерева. Такая структура позволяет определить некоторый уровень иерархии, на котором происходит обрезание дерева и формирование кластеров. Ранее было продемонстрировано, что данный метод может применяться для создания системы неперекрывающихся кластеров, отражающих как нормальные, так и аномальные состояния исследуемой системы. Этот этап считается преобразованием данных с целью снижения размерности, представляя собой фазу предварительной обработки данных. На втором этапе, данные, полученные после первого этапа, подвергаются обработке с использованием ансамбля неагломеративных методов кластеризации или классификации, что позволяет выделить класс аномалий в неразмеченных данных. В работе приведены три способа определения конечного класса точки данных как аномальной: по одному голосу, всем голосам, больше 50%.

Для сравнения различных способов ансамблирования предлагается рассчитывать следующие параметры:

– $\mathrm{R}_{nn} = \frac{N_{nn}}{N_n}$ – соотношение нормальных кластеров, не содержащих аномальные данные относительно общего числа нормальных кластеров (1);

– $\mathrm{R}_{aa} = \frac{N_{aa}}{N_a}$ – соотношение аномальных кластеров, не содержащих нормальные данные относительно общего числа аномальных кластеров (2).

Соотношения (1, 2) позволяет нам понять качество итоговой кластеризации и классификации, так как показывает, насколько четко разделены аномальные и нормальные данные в получившихся кластерах.

При построении модели применялся датасет, записанный на стенде водоочистного сооружения SWaT [13]. Датасет представляет из себя набор показаний входов и выходов 75 контроллеров, сенсоров и актуаторов водоочистного сооружения. В ходе записи датасета производились имитационные кибератаки, при которых несанкционированно подавались управляющие сигналы на входы контроллеров, а также происходила подмена значений на выходе сенсоров и контроллеров. Общее количество записей в датасете – 15000. Доля аномальных данных при этом составляла 14%.

Вычислительные эксперименты проводились для трех рассматриваемых методов ансамблирования. При этом варьировался уровень иерархии дендрограммы, то есть изменялось количество кластеров, центры которых подавались на вход алгоритмов классификации. Также проводилась оптимизация параметров применяемых в ансамбле методов для получения оптимального результата.

Результаты экспериментов приведены в табл. 1 и на рис. 1, 2 и 3. В табл. 1 указаны средние значения показателей R_{aa} и R_{nn} для рассматриваемых методов ансамблирования. На рис. 1, 2 и 3 представлено сравнение рассматриваемых методов относительно параметров R_{aa}, R_{nn} и среднего из двух показателей. По результатам видно, что применение метода при одном голосе за выделение точки как аномальной и метод при более 50% за выделение точки как аномальной обладают наибольшей точностью. В диапазоне количества кластеров от 800 до 2000 эти два метода обладают самыми высокими средними значениями показателей R_{aa} и R_{nn}. При этом метод выделения аномалий по одному голосу показал самую высокую точность при количестве кластеров 1200–0,72.

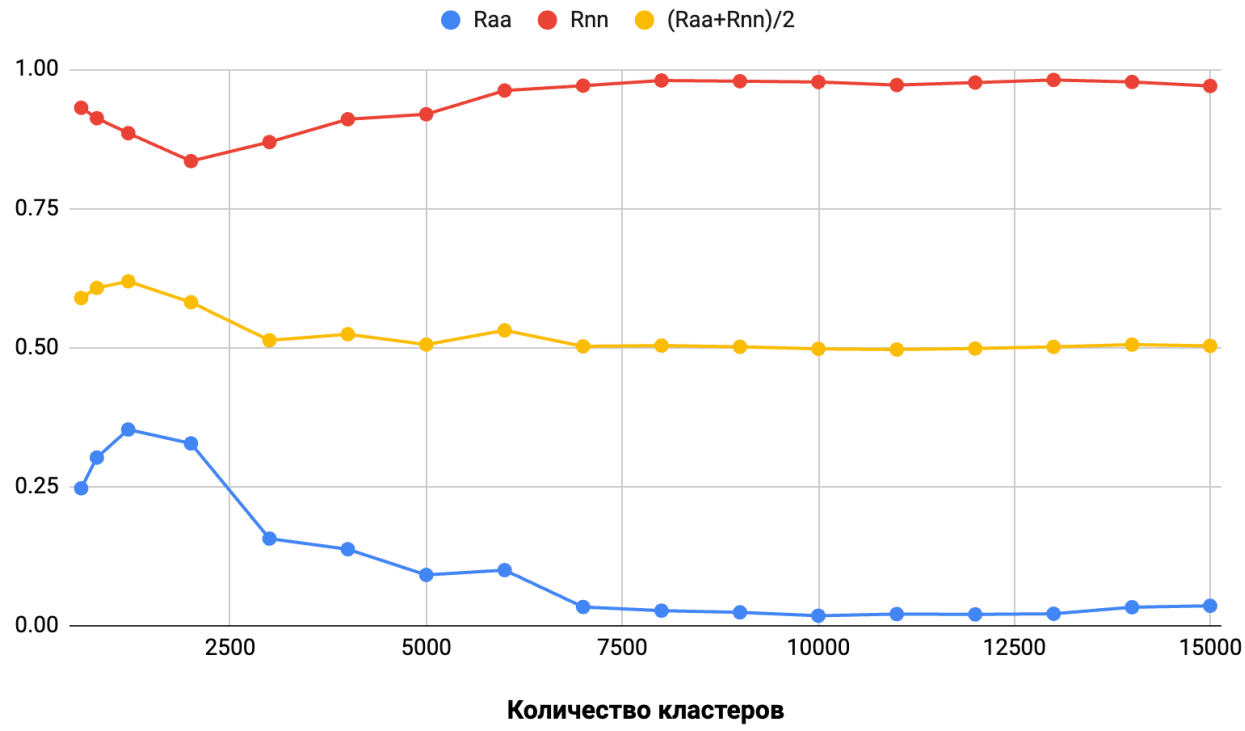


Рис. 1. Сравнение показателей R_{aa} и R_{nn} при методе выделения аномальной точки по всем голосам «за»

Таблица 1

Сравнение методов ансамблирования

Количество кластеров	$\frac{R_{nn}+R_{aa}}{2}$, все голоса	$\frac{R_{nn}+R_{aa}}{2}$, 1 голос	$\frac{R_{nn}+R_{aa}}{2}$, >50% голосов
600	0,59	0,67	0,59
800	0,61	0,70	0,63
1200	0,62	**0,72**	0,71
2000	0,58	0,70	0,66
3000	0,51	0,66	0,62
4000	0,52	0,68	0,63
5000	0,51	0,67	0,58
6000	0,53	0,67	0,60
7000	0,50	0,67	0,60
8000	0,50	0,68	0,63
9000	0,50	0,68	0,64
10000	0,50	0,67	0,64
11000	0,50	0,66	0,64
12000	0,50	0,68	0,65
13000	0,50	0,67	0,63
14000	0,51	0,66	0,61
14996	0,50	0,66	0,57

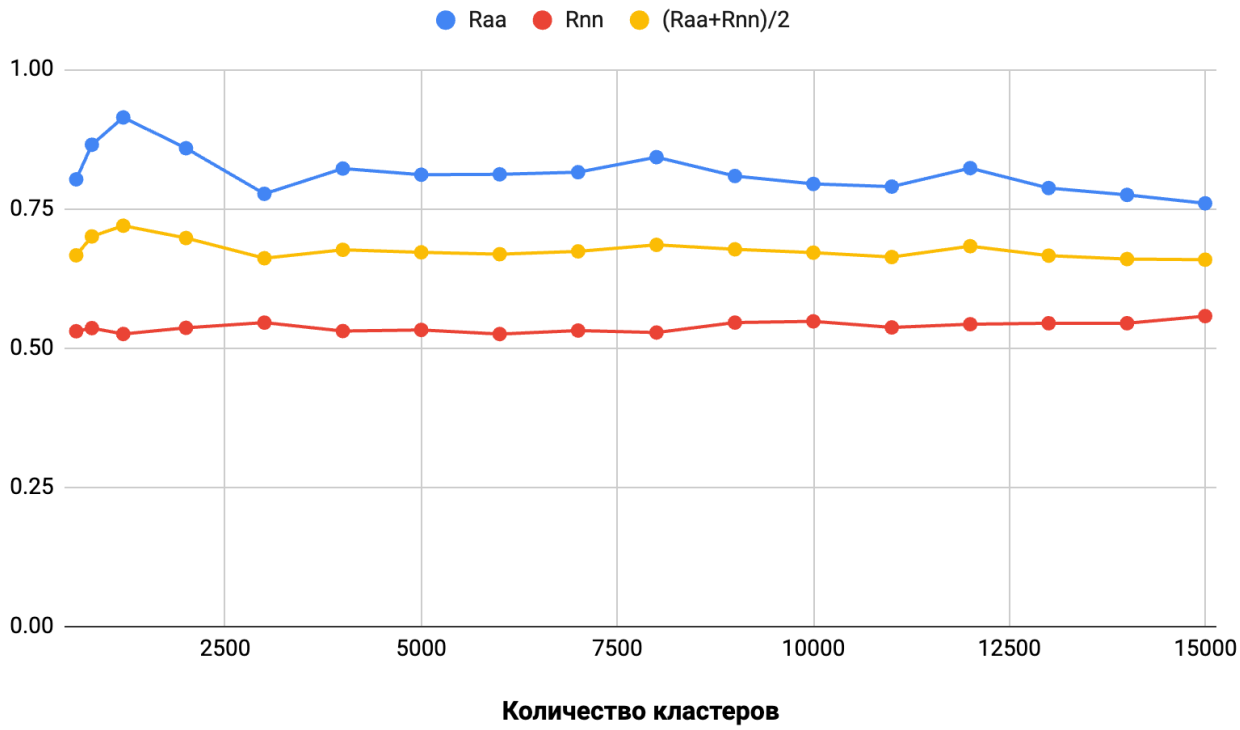


Рис. 2. Сравнение показателей R_{aa} и R_{nn} при методе выделения аномальной точки по одному голосу «за»

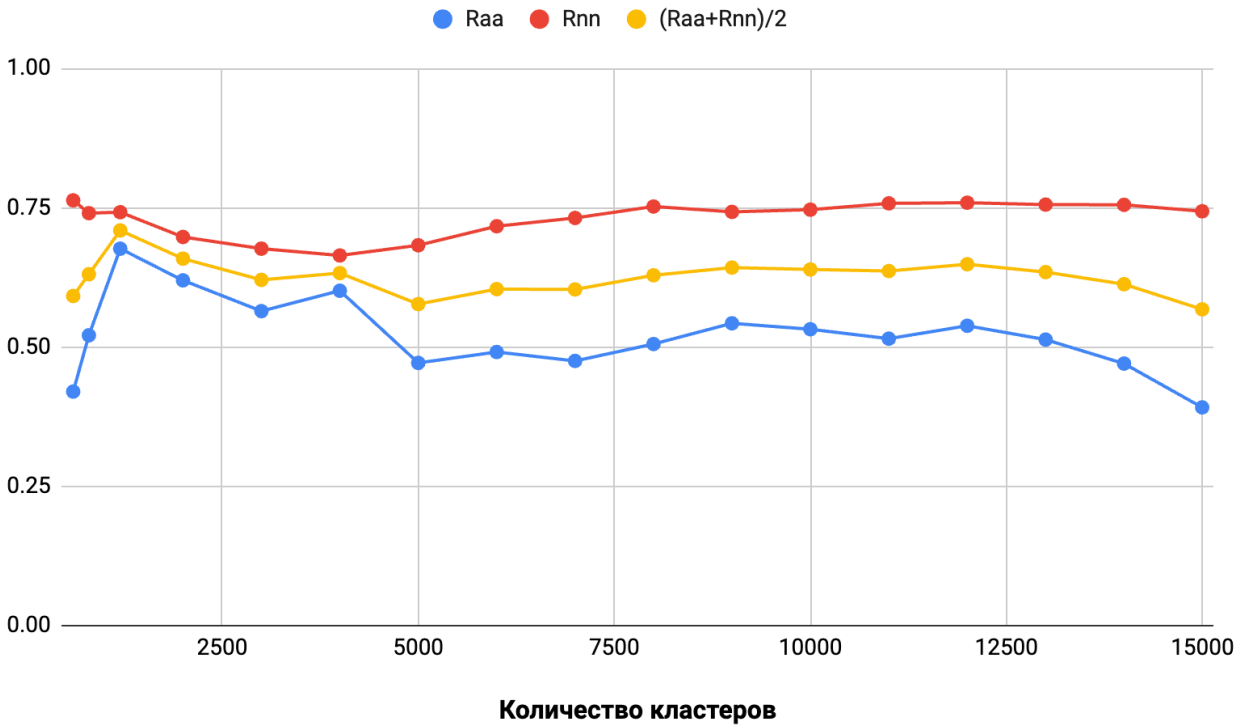


Рис. 3. Сравнение показателей R_{aa} и R_{nn} при методе выделения аномальной точки при более 50% голосов «за»

Заключение. В результате работы построена модель информационных процессов АСУ ТП, подвергающихся кибератакам, с использованием предлагаемого двухэтапного метода определения нормальных и аномальных состояний и трех методов ансамблирования различных неиерархических методов кластеризации или классификации. При проведении сравнительного анализа метод, при котором точка данных определяется как аномальная при получении более 50% голосов «за», оказался оптимальный при сравнении его на основе показателей R_{aa} и R_{nn}. Дальнейшая работа над методом будет посвящена оптимизации предлагаемого метода и сравнение его с другими методами, применяя общепринятые параметры качества моделей по обнаружению аномалий в данных.

Библиографический список

1. Falliere N. W32. Stuxnet dossier / N. Falliere, L.O. Murchu, E. Chien // White paper, Symantec Corp., Security Response. – 2011. – Т. 5, № 6. – С. 29.
2. Lee R.M. TRISIS: Analyzing Safety System Targeting Malware. DRAGOS, 2017.
3. Lee R.M., Assante M.J., Conway T. Analysis of the cyberattack on the Ukrainian power grid // SANS Industrial Control Systems. – 2016. – Т. 23. – С. 1–18.
4. Alguliyev R., Imamverdiyev Y., Sukhostat L. Cyber-physical systems and their security issues // Computers in Industry. – 2018.– Т. 100, – С. 212–223.

5. Murtagh, F. Algorithms for hierarchical clustering: an overview / F. Murtagh, P. Contreras // WIREs Data Mining Knowl. Discov. – 2012. – Т. 2. – №1. – С. 86–97.

6. Murtagh, F. Algorithms for hierarchical clustering: an overview, II / F. Murtagh, P. Contreras // WIREs Data Mining Knowl. Discov. – 2017. – Т. 7. – № 6. – С. 12–19.

7. Cohen-Addad, V. Hierarchical Clustering: Objective Functions and Algorithms / V. Cohen-Addad, V. Kanade, F. Mallmann-Trenn, C. Mathieu // Journ. ACM. – 2019. – Т. 66. – № 4. – С. 26–42.

8. F. T. Liu, K. M. Ting and Z. -H. Zhou, "Isolation Forest," 2008 Eighth IEEE International Conference on Data Mining, Pisa, Italy, 2008, P. 413-422, DOI: 10.1109/ICDM.2008.17.

9. Breunig M. M. и др. LOF // SIGMOD Rec. 2000. Т. 29. № 2. С. 93–104.

10. Z. Li, Y. Zhao, X. Hu, N. Botta, C. Ionescu, и G. H. Chen, «ECOD: Unsupervised Outlier Detection Using Empirical Cumulative Distribution Functions», arXiv, 2022, doi: 10.48550/ARXIV.2201.00382.

11. Kriegel H. и др. Density-based clustering // WIREs Data Min & Knowl. 2011. Т. 1. № 3. С. 231–240.

12. Z. Li, Y. Zhao, N. Botta, C. Ionescu, и X. Hu, «COPOD: Copula-Based Outlier Detection», arXiv, 2020, doi: 10.48550/ARXIV.2009.09463.

13. Goh, J. A Dataset to Support Research in the Design of Secure Water Treatment Systems / J. Goh, S. Adepu, K. Nazir Juneo, A. Mathur // CRITIS 2016: Critical Information Infrastructures Security. – 2017. – С. 88–99.

СОДЕРЖАНИЕ

Секция «Организационные, правовые, гуманитарные и социальные аспекты обеспечения информационной безопасности»

Секция «Прикладные и научные исследования в области защиты информации от утечек по техническим каналам»

Секция «Программно-аппаратные средства защиты информации и компьютерная безопасность»

Секция «Математические методы и анализ данных в обеспечении информационной безопасности»

Научное издание

БЕЗОПАСНОСТЬ ИНФОРМАЦИОННОГО ПРОСТРАНСТВА

Сборник трудов XXII Всероссийской научно-практической конференции студентов, аспирантов и молодых учёных

Составитель **Соколов** Александр Николаевич

Техн. редактор *А.В. Миних*
Дизайн обложки *А.С. Пановой*

Издательский центр Южно-Уральского государственного университета

Подписано в печать 06.02.2024. Формат 60×84 1/16. Печать цифровая.
Усл. печ. л. 18,13. Тираж 100 экз. Заказ 15/44.

Отпечатано с оригинал-макета заказчика в типографии Издательского центра ЮУрГУ.
454080, г. Челябинск, проспект Ленина, 76.