**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ**
**РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ**
**ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**
**«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**
**ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ»**

**КАФЕДРА ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ**
**И ПРОГРАММИРОВАНИЯ**

# ЦИФРОВЫЕ ТЕХНОЛОГИИ И ПРОБЛЕМЫ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

**Под редакцией Е.В. Стельмашонок, И.Н. Васильевой**

**ИЗДАТЕЛЬСТВО**
**САНКТ-ПЕТЕРБУРГСКОГО ГОСУДАРСТВЕННОГО**
**ЭКОНОМИЧЕСКОГО УНИВЕРСИТЕТА**
**2021**

**ББК 32.973.2**
**Ц75**

**Цифровые** технологии и проблемы информационной безопасности / Ц75 под ред. Е.В. Стельмашонок, И.Н. Васильевой. – СПб. : Изд-во СПбГЭУ, 2021. – 163 с.

ISBN 978-5-7310-5243-6

В монографии исследуются проблемы информационной безопасности, сопровождающие процессы внедрения новых цифровых технологий, и методы защиты информации. Рассматриваются подходы к выявлению злоупотреблений и деструктивных воздействий в компьютерных системах, вопросы моделирования и методические аспекты построения системы защиты информации на предприятии. Приводится анализ современных технологий и средств защиты информации. Излагается опыт применения образовательных технологий в области информационной безопасности.

Монография может быть полезна преподавателям, студентам, магистрантам, аспирантам, занимающимся исследованием данных проблем и специалистам в области защиты информации, а также ИТ специалистам и всем, кто интересуется вопросами информационной безопасности.

The monograph examines the problems of information security, accompanying the processes of introducing new digital technologies, so well as the mesures of protecting information. Approaches to identifying abuses and destructive influences in computer systems, modeling issues and methodological aspects of building an information security system at an enterprise are considered. The analysis of modern technologies and information security tools is provided. The experience of using educational technologies in the field of information security is presented.

The publication can be useful to teachers, students, undergraduates, graduate students who research these problems, to professionals in the field of information security, as well as IT professionals and anyone interested in questions of information security issues.

**ББК 32.973.2**

**Коллектив авторов:**
**Абдуллин Т.И.** (4.3), **Баев В.Д.** (2.10), **Буйневич М.В.** (1.4), **Бурзунов Д.Д.** (3.6), **Васильева И.Н.** (2.4, 2.5), **Галиуллина Э.Ф.** (3.3), **Гатчин Ю.А.** (1.5), **Гниденко И.Г.** (2.1), **Горохов Н.А.** (2.1), **Горсткин И.В.** (2.3), **Долженко А.Б.** (2.7), **Егорова И. В.** (2.1), **Еникеева Л.А.** (3.4), **Ефимов В.С.** (1.2), **Жиров А.Р.** (3.3), **Израилов К.Е.** (1.4), **Красильникова Е.В.** (3.2), **Красненков А.М.** (2.7), **Ласкус А.С.** (2.7), **Локнов А. И.** (4.1), **Майорова Е.В.** (3.2), **Мердина О.Д.** (2.2), **Морозов С.К.** (2.9–2.11), **Петров В. Г.** (3.1), **Полегенько А.М.** (2.6), **Простак А.А.** (2.1), **Родин В.Н.** (3.4), **Савельева Н.А.** (2.9), **Сальников В.** (3.3), **Сидоров Е.С.** (1.1, 1.3), **Смирнов Н.С.** (2.11), **Смирнова Т.В.** (2.4), **Соколовская С.А.** (3.2), **Солодянников А.В.** (1.2), **Стельмашонок В Л.** (3.5), **Стельмашонок Е.В.** (3.5), **Сухостат В.В.** (1.5, 3.3, 3.6), **Сясин Н.И.** (4.4), **Тверитин И.С.** (2.8), **Федоров Д.Ю.** (4.2), **Филиппов В.Д.** (1.1, 1.3), **Чернокнижный Г.М.** (4.3).

**Рецензенты**: профессор СПб университета МВД России, д-р техн. наук, проф. **Д.И. Якушев**; зав. кафедрой информатики СПбГЭУ заслуженный деятель науки РФ, д-р техн. наук, проф. **В.В. Трофимов**

**ISBN 978-5-7310-5243-6**

# ОГЛАВЛЕНИЕ

## Введение

Новые цифровые технологии стали движущей силой развития современного мира, прочно войдя во все сферы профессиональной деятельности и повседневной жизни человека. Однако наряду с инновационной составляющей процессы цифровизации несут в себе определенные угрозы, связанные, прежде всего, с проблемами информационной безопасности и обеспечения приватности, возможностью деструктивных воздействий и злоупотреблений в цифровой среде. Все это остро ставит вопросы обеспечения безопасности цифровых технологий, как на уровне отдельных компьютерных систем, так и на уровне организаций и общества в целом.

Первая глава монографии посвящена проблеме выявления злоупотреблений в компьютерных системах. Сделан акцент на вопросах контроля недекларированных возможностей, в частности, предложены подходы к анализу программного кода на наличие недекларированных возможностей, а также заимствований. Кроме того, рассмотрены модели выявления злоупотреблений в компьютерных сетях за счет анализа аномалий сетевого трафика.

Вторая глава отражает современные тенденции обеспечения различных аспектов безопасности цифровых технологий. Рассмотрены подходы к обеспечению информационной безопасности NoSQL баз данных, цифровых валют, веб-приложений, «умных вещей». Затронуты дискуссионные вопросы применения криптографической защиты запросов DNS. Рассмотрены средства и методы аутентификации, стеганографии, а также нейросетевые инструменты криптографической защиты информации. Кроме того, в эту главу включены вопросы, связанные с физической защитой информационных систем – построение систем контроля и управления доступом, мониторингом физического состояния сотрудников охраны и системам предотвращения аварийных ситуаций в инженерных сетях.

Обеспечение безопасности цифровых технологий служит, прежде всего, средством соблюдения законных интересов личности, отдельных хозяйственных субъектов, государства и общества в целом в информационной сфере. Важным требованием цифровизации становится соблюдение приватности и защита персональных данных субъектов информационных отношений. В третьей главе излагаются методические аспекты создания системы защиты информации в организациях. Описаны необходимые действия, которые должна предпринять организация для соблюдения отечественных и европейских требований к защите персональных данных. Даны подходы к разработке типовой модели нарушителя. Рассмотрены вопросы оценки экономической эффективности затрат на инфраструктуру

защиты информации предприятия. Кроме того, затронуты гуманитарные аспекты информационной безопасности.

Не менее важной задачей, по мнению авторов, является подготовка новых кадров, формирование будущих специалистов в области информационной безопасности. Формирование навыков обеспечения информационной безопасности в условиях постоянного развития цифровых технологий, появления как новых атак, так и методов защиты, невозможно без применения активных методов обучения. Эти вопросы нашли отражение в четвертой главе монографии, где излагается опыт использования современных образовательных технологий в сфере информационной безопасности, а также затрагиваются вопросы информационной безопасности при организации процессов дистанционного обучения.

Авторский коллектив монографии составляют преподаватели СПбГЭУ и других ВУЗов, действующие специалисты в области информационной безопасности, а также студенты старших курсов, обучающиеся по направлению «Информационная безопасность». Вклад каждого из авторов указан ниже.

- Абдуллин Т.И. – студент направления группы ИБ-1701 Санкт-Петербургского государственного экономического университета – п. 4.3.
- Баев В.Д. – студент Санкт-Петербургского государственного университета телекоммуникаций им. проф. М.А. Бонч-Бруевича – п. 2.10.
- Буйневич М.В. – профессор Санкт-Петербургского университета государственной противопожарной службы МЧС России, профессор Санкт-Петербургского государственного университета телекоммуникаций им. проф. М.А. Бонч-Бруевича, д.т.н., профессор – п. 1.4.
- Бурзунов Д.Д. – студент группы ИБ-1701 Санкт-Петербургского государственного экономического университета – п. 3.6.
- Васильева И.Н. – доцент Санкт-Петербургского государственного экономического университета, доцент Санкт-Петербургского университета МВД РФ, к.ф.-м.н., доцент − п. 2.4, 2.5.
- Галиуллина Э.Ф. – студент группы ИБ-1701 Санкт-Петербургского государственного экономического университета – п. 3.3.
- Гатчин Ю.А. – профессор Университета ИТМО, д.т.н., профессор – п. 1.5.
- Гниденко И.Г. – доцент Санкт-Петербургского государственного экономического университета, к.э.н., доцент – п. 2.1.
- Горохов Н.А. – студент группы ИБ-1701 Санкт-Петербургского государственного экономического университета – п. 2.1.
- Горсткин И.В. – студент группы ИБ-1701 Санкт-Петербургского государственного экономического университета – п. 2.3.

- Долженко А.Б. – студент группы ИБ-1701 Санкт-Петербургского государственного экономического университета – п. 2.7.
- Егорова И.В. – доцент Санкт-Петербургского государственного экономического университета, к.э.н., доцент − п. 2.1.
- Еникеева Л.А. – профессор Санкт-Петербургского государственного института кино и телевидения, д.э.н., профессор − п. 3.5.
- Ефимов В.С. – студент группы ИБ-1701 Санкт-Петербургского государственного экономического университета – п. 1.2.
- Жиров А.Р. – студент группы ИБ-1701 Санкт-Петербургского государственного экономического университета – п. 3.3.
- Израилов К.Е. – доцент Санкт-Петербургского государственного университета телекоммуникаций им. проф. М.А. Бонч-Бруевича, старший научный сотрудник лаборатории проблем компьютерной безопасности Санкт-Петербургского Федерального исследовательского центра Российской академии наук, к. т. н. – п. 1.4.
- Красильникова Е.В. – доцент Санкт-Петербургского государственного экономического университета, к.т.н., доцент − п. 3.2.
- Красненков А.М. – студент группы ИБ-1701 Санкт-Петербургского государственного экономического университета – п. 2.7.
- Ласкус А.С. – студент группы ИБ-1701 Санкт-Петербургского государственного экономического университета – п. 2.7.
- Локнов А.И. – старший преподаватель Санкт-Петербургского университета МВД России, майор полиции, к.т.н. – п. 4.1.
- Майорова Е.В. – доцент Санкт-Петербургского государственного экономического университета, к.т.н., доцент − п. 3.2.
- Мердина О.Д. – доцент Санкт-Петербургского государственного экономического университета, к.э.н., доцент − п. 2.2.
- Морозов С.К. – старший преподаватель Санкт-Петербургского государственного экономического университета − пп. 2.9 – 2.11.
- Петров В.Г. – заместитель директора УМЦ «ХимИнформЗащита» по учебно-методической работе, к.т.н., доцент − п. 3.1.
- Полегенько А.М. – начальник отдела криптографического и инженерно-криптографического анализа ООО «СТЦ», старший преподаватель Санкт-Петербургского государственного экономического университета, аспирант Университета ИТМО – п. 2.6.
- Простак А.А. – студент группы ИБ-1701 Санкт-Петербургского государственного экономического университета – п. 2.1.
- Родин В.Н. – доцент Санкт-Петербургского университета МВД России, к.т.н., доцент – п. 3.4.
- Савельева Н.А. – студент Санкт-Петербургского государственного университета телекоммуникаций им. проф. М.А. Бонч-Бруевича – п. 2.9.

- Сальников В. – студент группы ИБ-1701 Санкт-Петербургского государственного экономического университета – п. 3.3.
- Сидоров Е.С. – студент группы ИБ-1701 Санкт-Петербургского государственного экономического университета, инженер-испытатель 2-й категории ООО «Ассоциация специалистов по безопасности» (Санкт-Петербург) – п. 1.1, 1.3.
- Смирнов Н.С. – студент Санкт-Петербургского государственного университета телекоммуникаций им. проф. М.А. Бонч-Бруевича – п. 2.11.
- Смирнова Т.В. – студент группы ИБ-1701 Санкт-Петербургского государственного экономического университета – п. 2.4.
- Соколовская С.А. – доцент Санкт-Петербургского государственного экономического университета, к.э.н., доцент − п. 3.2.
- Солодянников А.В. – генеральный директор ООО «Ассоциация специалистов по безопасности» (Санкт-Петербург), к.т.н., доцент − п. 1.2.
- Стельмашонок В.Л. – доцент Санкт-Петербургского государственного экономического университета, к.э.н., доцент – п. 3.5.
- Стельмашонок Е.В. – зав. кафедрой вычислительных систем и программирования Санкт-Петербургского государственного экономического университета д.э.н., профессор – п. 3.5.
- Сухостат В.В. – доцент Санкт-Петербургского государственного экономического университета, доцент Университета ИТМО, к.т.н., к.пед.н., доцент − п. 1.5, 3.3, 3.6.
- Сясин Н.И. – доцент Санкт-Петербургского государственного экономического университета, к. б. н., доцент – п. 4.4.
- Тверитин И.С. – студент группы ИБ-1701 Санкт-Петербургского государственного экономического университета – п. 2.8.
- Федоров Д.Ю. – старший преподаватель Санкт-Петербургского государственного экономического университета – п. 4.2.
- Филиппов В.Д. – студент группы ИБ-1701 Санкт-Петербургского государственного экономического университета, инженер-испытатель 2-й категории ООО «Ассоциация специалистов по безопасности» (Санкт-Петербург) – п. 1.1, 1.3.
- Чернокнижный Г.М. – доцент Санкт-Петербургского государственного экономического университета, к.т.н., доцент − п. 4.3.

# ГЛАВА 1. ВЫЯВЛЕНИЕ УГРОЗ БЕЗОПАСНОСТИ И ЗЛОУПОТРЕБЛЕНИЙ В КОМПЬЮТЕРНЫХ СИСТЕМАХ

## 1.1. Разработка программного комплекса анализа исходных текстов программного обеспечения на отсутствие недекларированных возможностей

В соответствии с Федеральным законом от 27 декабря 2002 г. № 184-ФЗ «О техническом регулировании» (статья 5) и Постановлением Правительства Российской Федерации от 26 июня 1995 г. № 608 «О сертификации средств защиты информации», оборонная продукция, поставляемая по государственному оборонному заказу, продукция, используемая в целях защиты сведений, составляющих государственную тайну или относимых к охраняемой в соответствии с законодательством Российской Федерации иной информации ограниченного доступа, продукция, сведения о которой составляют государственную тайну, подлежат техническому регулированию, следовательно, требуют получения сертификата соответствия, что включает в себя проведение сертификационных испытаний. Сертификационные испытания, относящиеся к вышеперечисленным категориям, включают в себя контроль отсутствия недекларированных возможностей, регламентированный руководящим документом – «Защита от несанкционированного доступа к информации. Часть 1. Программное обеспечение средств защиты информации. Классификация по уровню контроля отсутствия недекларированных возможностей» [1].

Недекларированные возможности – функциональные возможности программного обеспечения, не описанные или не соответствующие описанным в документации, при использовании которых возможно нарушение конфиденциальности, доступности или целостности обрабатываемой информации. Реализацией недекларированных возможностей, в частности, являются программные закладки [2].

Программные закладки – преднамеренно внесенные в ПО функциональные объекты, которые при определенных условиях (входных данных) инициируют выполнение не описанных в документации функций ПО, приводящих к нарушению конфиденциальности, доступности или целостности обрабатываемой информации [2].

В настоящей работе применяются следующие сокращения:

- АСГ – Абстрактный семантический граф;
- АСД – Абстрактное синтаксическое дерево;
- КСД – Конкретное синтаксическое дерево;.
- НДВ – Недекларированные возможности;
- ПО – Программное обеспечение;

– РД НДВ – Руководящий документ. Защита от несанкционированного доступа к информации. Часть 1. Программное обеспечение средств защиты информации. Классификация по уровню контроля отсутствия недекларированных возможностей;
– ЯП – Язык программирования.

Все перечисленные сокращения являются общепринятыми.

Руководящий документ устанавливает четыре уровня контроля отсутствия НДВ (наиболее высокий – первый). Каждый уровень характеризуется определенным минимальным набором требований. Программный комплекс, описанный в данной статье, обеспечивает контроль отсутствия НДВ с 4-го уровня по 2-й. Учитывая, что требования к каждому уровню включают в себя требования ко всем предыдущим уровням, функционирование программного комплекса описано для наивысшего уровня из поддерживаемых – второго.

Данный параграф посвящен разработке инструмента анализа исходных текстов программного обеспечения на отсутствие НДВ в соответствии с руководящим документом [2]. В рамках этого направления можно выделить следующие задачи по разработке алгоритмов:

1. построения математической модели испытываемого ПО на основе исходного текста;
2. анализа полученной математической модели в соответствии с требованиями руководящего документа;
3. формирования исходных данных для ручного анализа.

Результатом работы программного комплекса является отчет, используемый специалистом испытательной лаборатории для формирования заключения о соответствии.

Руководящий документ устанавливает следующие требования к испытаниям объекта оценки на отсутствие НДВ:

1. Контроль целостности.

Результатами контроля исходного состояния ПО должны быть уникальные значения контрольных сумм, рассчитанных для каждого исследуемого исходного файла.

2. Статический анализ:

– контроль полноты и отсутствия избыточности на уровне файлов;
– контроль соответствия объектному (загрузочному) коду;
– контроль полноты и отсутствия избыточности на уровне процедур;
– контроль связей функциональных объектов по управлению;
– контроль связей функциональных объектов по информации;
– контроль информационных объектов различных типов;
– формирование перечня маршрутов выполнения функциональных объектов (процедур, функций);

- контроль полноты и отсутствия избыточности исходных текстов контролируемого программного обеспечения на уровне функциональных объектов (функций);
- синтаксический контроль наличия заданных конструкций в исходных текстах ПО из списка (базы) потенциально опасных программных конструкций;
- формирование перечня маршрутов выполнения функциональных объектов (ветвей);
- построение по исходным текстам контролируемого ПО блок-схем, диаграмм и т.п., и последующий сравнительный анализ алгоритма работы функциональных объектов (процедур, функций) и алгоритма работы, приведенного в «Пояснительной записке».

3. Динамический анализ:

- контроль выполнения функциональных объектов (процедур, функций);
- сопоставление фактических маршрутов выполнения функциональных объектов (процедур, функций) и маршрутов, построенных в процессе проведения статического анализа;
- анализ критических маршрутов выполнения функциональных объектов (процедур, функций) для заданных экспертом списков информационных объектов;
- контроль выполнения функциональных объектов (ветвей)
- сопоставление фактических маршрутов выполнения функциональных объектов (ветвей) и маршрутов, построенных в процессе проведения статического анализа [2].

С целью максимизации полноты исполнения вышеперечисленных требований предлагается анализировать не исходные тексты, а полную математическую модель, построенную на их основе. Таким образом, испытания состоят из следующих этапов, выполняемых последовательно.

1. Построение математической модели.

А. Лексический анализ исходного текста.

– Процедура лексеризации.

Этап лексеризации подразумевает разбиение исходного текста (потока символов) на лексемы [3]. Лексика языка описывается набором регулярных выражений и импортируется из отдельного модуля. Таким образом, лексический анализатор не зависит от ЯП исследуемого исходного текста, что позволяет расширять список ЯП, поддерживаемых анализатором. Входные данные лексера – файл с исходным текстом (рис. 1.1), выходные данные – поток лексем (рис. 1.2).

```
##comment
def reverse(string):
   return string[::-1]
a=input('input string: ')
b=int(input('input int:'))
if len(a)<b:
   print(reverse(a)*b)
else:
   for i in range(b):
      print(b)
```

Рисунок 1.1 – Упрощенный пример исходного текста для анализа

| file_id | lexem_id | string | start_pos | start_lineno | end_pos | end_lineno |
|---|---|---|---|---|---|---|
| 1 | 1 | ##comment | 0 | 1 | 9 | 1 |
| 1 | 2 | def | 0 | 2 | 3 | 2 |
| 1 | 3 | reverse | 4 | 2 | 11 | 2 |
| 1 | 4 | ( | 11 | 2 | 12 | 2 |
| 1 | 5 | string | 12 | 2 | 18 | 2 |
| 1 | 6 | ) | 18 | 2 | 19 | 2 |
| 1 | 7 | : | 19 | 2 | 20 | 2 |
| 1 | 8 | return | 4 | 3 | 10 | 3 |
| 1 | 9 | string | 11 | 3 | 17 | 3 |
| 1 | 10 | [ | 18 | 3 | 19 | 3 |
| 1 | 11 | : | 19 | 3 | 20 | 3 |
| 1 | 12 | : | 20 | 3 | 21 | 3 |
| 1 | 13 | - | 21 | 3 | 22 | 3 |
| 1 | 14 | 1 | 22 | 3 | 23 | 3 |
| 1 | 15 | ] | 23 | 3 | 24 | 3 |

Рисунок 1.2 – фрагмент потока лексем

– Процедура токенизации.

Токенизация – процедура преобразования потока лексем в поток токенов [4]. На этом этапе определяется тип каждой лексемы. Возможные типы импортируются из отдельного модуля, что позволяет абстрагировать анализатор от специфики ЯП. Входные данные токенизатора – поток лексем, выходные данные – поток токенов (пар вида «лексема – тип лексемы», рис. 1.3).

| file_id | token_id | lexem_id | token_type | token_type_code |
|---|---|---|---|---|
| 1 | 1 | 1 | COMMENT | 13 |
| 1 | 2 | 2 | KEYWORD_def | 17 |
| 1 | 3 | 3 | NAME | 25 |
| 1 | 4 | 4 | RBRACE_O | 10 |
| 1 | 5 | 5 | NAME | 25 |
| 1 | 6 | 6 | RBRACE_C | 11 |
| 1 | 7 | 7 | COLON | 8 |
| 1 | 8 | 8 | KEYWORD_return | 15 |
| 1 | 9 | 9 | NAME | 25 |

Рисунок 1.3 – фрагмент потока токенов

Б. Синтаксический анализ потока токенов

Синтаксический анализ — процесс разбора потока токенов в соответствии с синтаксисом ЯП, правила которого импортируются как отдельный модуль. Результатом синтаксического анализа является дерево разбора (конкретное синтаксическое дерево, КСД) — математическая модель исходного текста, учитывающая синтаксис ЯП и не учитывающая его семантику.

В. Семантический анализ дерева разбора

– Построение абстрактного синтаксического дерева (рис. 1.4)

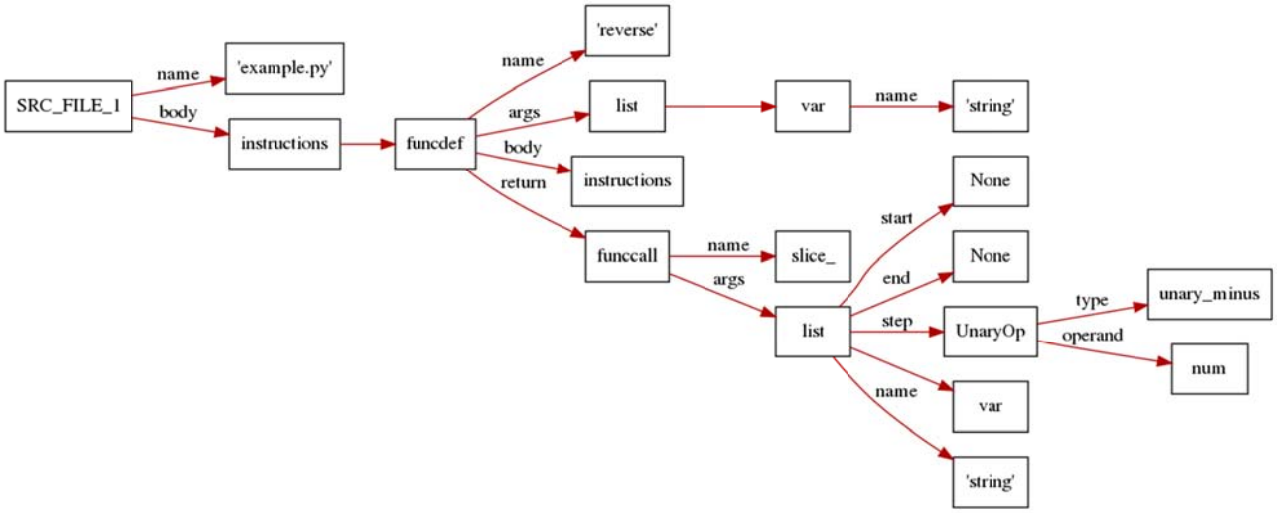



Рисунок 1.4 – Абстрактное синтаксическое дерево
(демонстрационный фрагмент)

В соответствии с семантикой языка осуществляется преобразование дерева разбора (КСД) в АСД путем абстрагирования от синтаксиса ЯП, удаления либо преобразования неинформативных сущностей. Правила

семантики языка импортируются из отдельного модуля в соответствии с архитектурой программного комплекса.

– Построение абстрактного семантического графа (рис. 1.5)

Производится анализ АСД-объекта в соответствии с семантикой ЯП. Добавляются дополнительные связи между узлами дерева, формируется граф, учитывающий структуру компонента ПО.

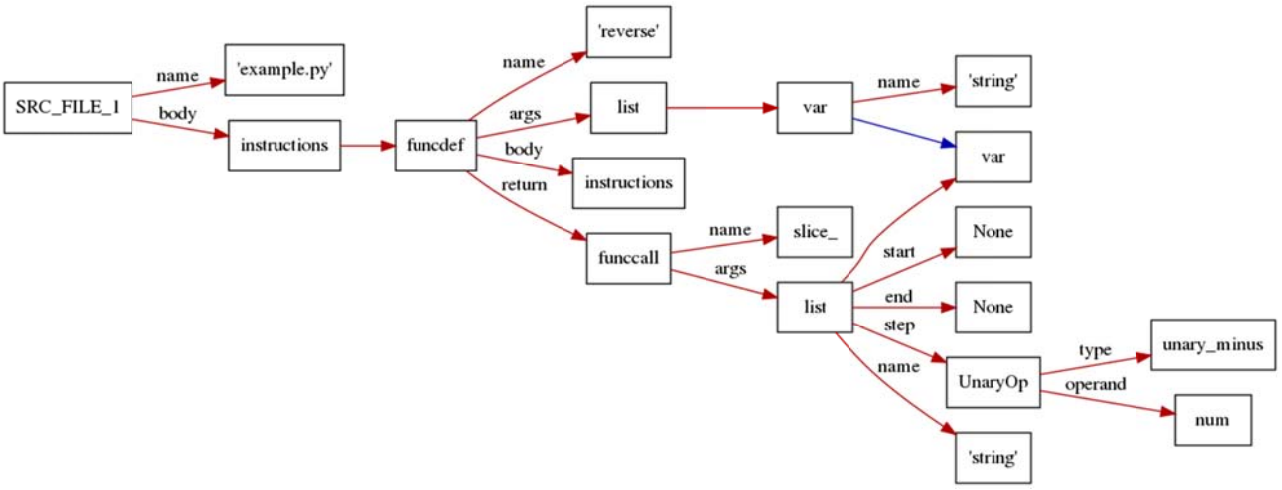



Рисунок 1.5 – Абстрактный семантический граф
(демонстрационный фрагмент)

Г. Создание единой модели ПО на основе моделей компонентов ПО, полученных на предыдущих этапах анализа (рис. 1.6).

Выполняются поиск и установка связей между узлами АСГ, построенных для каждого файла, входящего в состав анализируемого ПО. На этом этапе строится единая модель, представляющая граф, учитывающий полную структуру ПО и смысловую составляющую исходных текстов, но независимую от специфики ЯП, что уменьшает вероятность ошибок в работе программы, упрощает поддержку ПО и дает максимально широкие возможности для дальнейшего развития анализатора.

2. Статический анализ имеющейся математической модели.

А. Контроль отсутствия избыточности на уровне файлов и функциональных объектов.

Получение на основе полученного ранее графа списка файлов и функциональных объектов, не используемых в функционировании анализируемого ПО.

Б. Контроль связей функциональных объектов по управлению.

Составление списка возможных переходов между линейными участками потока управления исполнением программы в зависимости от ее управляющей структуры.

В. Контроль связей функциональных объектов по информации.

Составление списка возможных переходов между линейными участками потока управления исполнением программы в зависимости от ее информационной структуры.

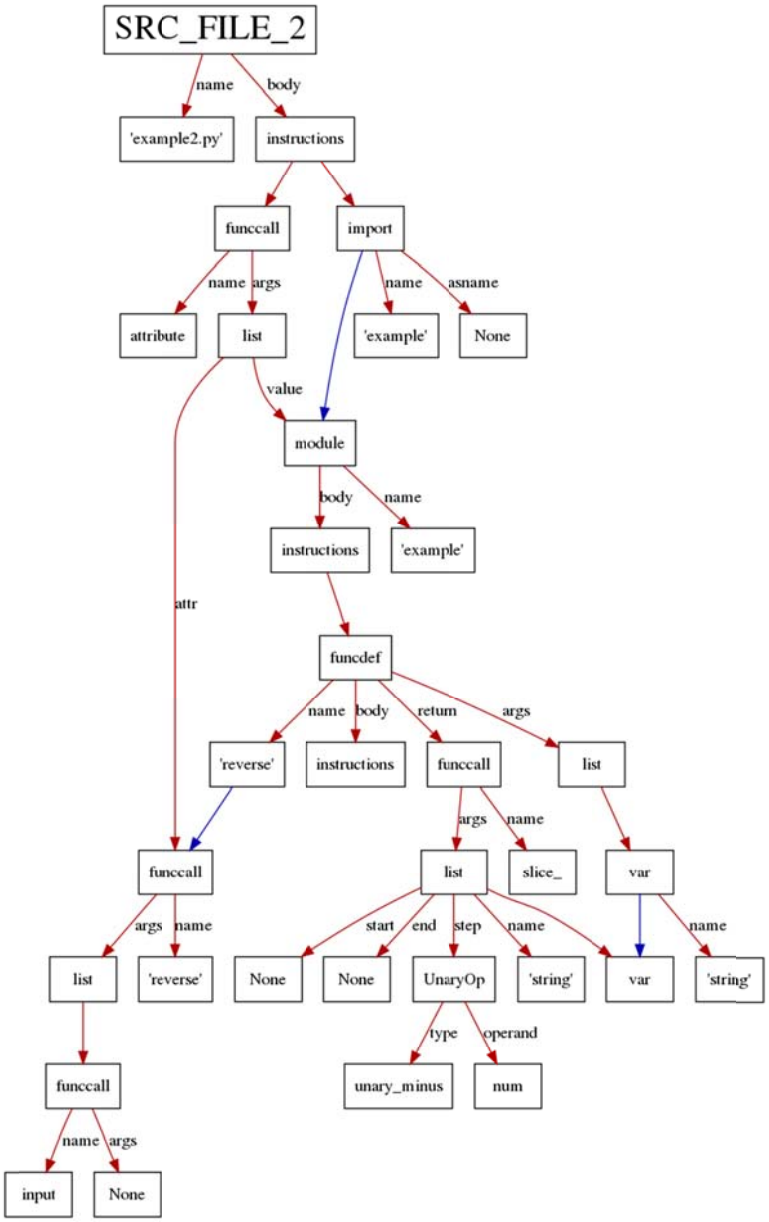



Рисунок 1.6 – Единая модель ПО, созданная из двух исходных файлов

Г. Формирование перечня маршрутов выполнения функциональных объектов

Перечень маршрутов создается путем сопоставления списков, полученных на двух предыдущих этапах.

Д. Построение модели маршрутов исполнения на основе перечня, сгенерированного на предыдущем этапе.

Модель маршрутов является графом, формируемым из общей модели ПО путем исключения объектов, не влияющих на поток исполнения.

3. Создание лабораторной версии ПО на основе статического анализа.

А. Для каждого поддерживаемого ЯП разработана функция-датчик, фиксирующая состояние компонентов среды тестирования.

Б. На основе перечня функциональных объектов исходные тексты ПО дополняются вызовами функции-датчика, таким образом, что каждое выполнение функционального объекта фиксируется в специально созданной структуре данных (будем называть ее «сенсораграмма»).

4. Динамический анализ лабораторной версии ПО

А. Исполнение лабораторной версии ПО.

На данном этапе экспериментальным путем составляется сенсораграмма.

Б. Контроль выполнения функциональных объектов

Построение списка фактических маршрутов выполнения путем анализа данных сенсораграммы.

В. Сопоставление фактических маршрутов выполнения функциональных объектов и маршрутов, построенных в процессе проведения статического анализа

Сравнение модели маршрутов, составленных на основе статического анализа и фактических маршрутов выполнения, полученных на предыдущем этапе.

5. Формирование отчета, содержащего результаты анализа, представленные в удобном для эксперта виде.

Полный отчет о работе анализатора содержит следующие данные:

- контрольные суммы файлов исходных кодов;
- перечень файлов, не используемых испытываемым по;
- перечень файлов, недостающих в испытываемом по;
- перечень функциональных объектов, не используемых испытываемым по;
- перечень функциональных объектов, недостающих в испытываемом по;
- перечень информационных объектов, не используемых испытываемым по;
- перечень информационных объектов, недостающих в испытываемом по;
- перечень допустимых маршрутов выполнения (полученных в результате статического анализа);
- перечень фактических маршрутов выполнения;
- результаты сопоставления фактических маршрутов выполнения с перечнем допустимых маршрутов.

Краткий отчет о работе анализатора содержит следующие данные:

- количество файлов, не используемых испытываемым по и их процентное отношение к общему количеству файлов исходных текстов;
- количество функциональных объектов, не используемых испытываемым по их процентное отношение к общему количеству функциональных объектов;
- количество информационных объектов, не используемых испытываемым по их процентное отношение к общему количеству информационных объектов;
- перечень допустимых маршрутов выполнения (полученных в результате статического анализа);
- результаты сопоставления фактических маршрутов выполнения с перечнем допустимых маршрутов.

Отчет вместе со всеми данными, полученными в ходе работы анализатора, предоставляется эксперту для ручного анализа с дальнейшим составлением заключения.

В результате проделанной работы разработан анализатор, отличающийся от ранее существовавших независимостью от ЯП, а также имеющий удобные возможности для расширяемости. В дальнейшем на основе разработанного программного комплекса возможна реализация 1-го уровня контроля в соответствии с РД НДВ.

Разработано ПО анализа исходных текстов ПО на отсутствие НДВ в соответствии с РД НДВ для осуществления контроля отсутствия НДВ с 4-го уровня по 2-ой, решением задач по построению математической модели испытываемого ПО на основе исходного текста, анализа полученной математической модели в соответствии с требованиями руководящего документа, формирования исходных данных для ручного анализа.

Предложен метод реализации анализатора исходных текстов программ, соответствующий руководящим документам ФСТЭК. Разработан ряд методов в подходе к анализу исходного текста программ вне зависимости от ЯП.

Список использованных источников:

1. О создании системы сертификации средств защиты информации Министерства обороны Российской Федерации по требования безопасности информации. Приказ Министра обороны Российской Федерации от 5 сентября 1996 г. № 058 [Электронный ресурс] – URL: https://zgt.mil.ru/Licenzirovanie-i-sertifikaciya/Sertifikaciya (дата обращения: 16.10.2020).
2. Защита от несанкционированного доступа к информации. Часть 1. Программное обеспечение средств защиты информации. Классификация по уровню контроля отсутствия недекларированных возможностей. Руководящий документ, утверждено решением председателя Государственной технической комиссии при Президенте Российской Федерации от 4 июня 1999г. № 114.

3. Ахо А., Ульман Дж. Теория синтаксического анализа, перевода и компиляции: Том 2 Компиляция /Prentice-Hall, Inc. Englewood Cliffs, N. J. 1973 – Москва: Изд-во «Мир», 1978. – 268 с.
4. Ахо А., Ульман Дж. Теория синтаксического анализа, перевода и компиляции: Том 1 Синтаксический анализ / Prentice-Hall, Inc. Englewood Cliffs, N. J. 1973 – Москва: Изд-во «Мир», 1978. – 283 с.
5. Автоматизированные системы. Защита от несанкционированного доступа к информации. Классификация автоматизированных систем и требования по защите информации. Руководящий документ, утверждено решением председателя Государственной технической комиссии при Президенте Российской Федерации от 30 марта 1992 г.
6. О техническом регулировании. Федеральный закон Российской Федерации от 27 дек. 2002 г. № 184-ФЗ, ред. от 28.11.2018 [Электронный ресурс] URL: http://www.consultant.ru/document/cons_doc_LAW_40241/ (дата обращения: 16.10.2020).
7. О сертификации средств защиты информации. Постановление Правительства Российской Федерации от 26 июня 1995 г. № 608, ред. от 21.04.2010 [Электронный ресурс] URL: http://www.consultant.ru/document/cons_doc_LAW_7054/ (дата обращения: 16.10.2020).

## 1.2. Сложности выполнения требований руководящего документа при проведении исследований на отсутствие недекларированных возможностей программного обеспечения

В процессе статического, динамического и других (например, классификация) подходов к выполнению контроля недекларированных возможностей формируется набор материалов. Этот набор, без экспертного анализа, имеет условно низкую ценность, т.к. для определения, например, функции удаления конфиденциальной информации, как программной закладки, необходимо быть уверенным, что эта функция не штатная и не заложена разработчиком согласно технического задания.

Как показывает практика, наиболее удачный вариант анализа – это изучение программистом-исследователем работы программы, путем целенаправленной работы с ней при условии наличия целого набора инструментов: отладчика, дизассемблера, узконаправленных анализаторов, – рассчитывающих определенные математические показатели вроде длины выделяемой памяти, частоты распределения конкретных конструкций, процессорное время выполнения, вероятность срабатывания и многое другое [1].

В рамках данной методики используется упрощенный вариант, направленный на реализацию требований ФСТЭК России В соответствии с методом выделяются такие сущности, как функции, переменные, про-

цедуры, ветви программы; затем их связи, строятся блок-схемы. Должно быть понятно, что это не совсем выполнение «требований» т.к. их выполнение, к обнаружению программных закладок исследователя не приближает. Но эти сведения могут быть собраны в процессе испытаний, например, «испытания одной кнопки». Инженер лаборатории рассматривает как эту кнопку и при каких условиях вызывает (по задумке разработчика) оператор, находит связанный с ней код, определяет место обработчика в общей структуре и затем пытается найти способ заставить программу выдавать результат срабатывания кнопки иной, нежели она имеет в задуманном программистами варианте.

При этом инженер строит схемы (чаще на листочке бумаги в стиле удобном ему для осознания, а не используя ГОСТ), связи и определяет входную/выходную информацию, может вставлять датчики в различные участки кода (но не во все сразу), расставляет мониторы слежения за внешней активностью приложения и выделяет конкретные элементы из такого журнала, который чаще всего слишком объемен. В итоге, если предположить, что исследования идут именно от общего к частному, то возможно допустить, что будут собраны сведения обо всех конструкциях (на самом деле это, конечно, не так по причине наличия очевидных кусков кода, не интересных с точки зрения безопасности).

В конечном счете набирается полное множество конструкций языка (правда, и тут руководящий документ не требует тотального разбора, ограничиваясь лишь функциональным программированием и не рассматривая объектно-ориентированное, и не говоря уже о более сложных парадигмах программирования). Так как создатель руководящего документа, профессор Л.Г. Осовецкий, не смог в 1997-1998 годах, формализовать процесс, описанный выше в качестве примера «испытания одной кнопки», то был сформирован последний пункт руководящего документа, подразумевающий отчетность в виде полных списков информационных, функциональных объектов, нечитаемых маршрутов выполнения, блок-схем и т.п.

В результате были созданы за последние время такие инструменты, как «АИСТ» («ЦБИ» для ФСТЭК России), «АК-ВС» (Эшелон) [2,3,4]. Наиболее основательно подошел «СИСТЕМПРОМ» со своим сертифицированным в Минобороны России анализатором, который успешно автоматически формирует отчеты. Стоимость инструментов необоснованно превышает цену любого компилятора (которые работают на тех же основаниях, только на выходе формируют работоспособный код) на порядки. При этом сами инструменты не компилируют, а создают огромные массивы информации о структурах, связях и результатах выполнения лабораторной версии, которые автоматически оценить (с точки зрения безопасности) невозможно из-за указанной в начале данного подраздела

примера с удалением тайны. Без человеческого вмешательства обойтись нельзя, что и определяется, как экспертный анализ.

В результате, именно эти массивы информации (порой миллионы связей в возможных маршрутах) инженеру лаборатории необходимо обобщить и выполнить анализ, после чего сделать вывод о соответствии требованиям по безопасности. Так, общество с ограниченной ответственностью «Ассоциация специалистов по безопасности» выполняет работы по проведению сертификационных испытаний на протяжении 5 лет, и за это время экспертами лаборатории были выявлены наиболее часто встречающиеся уязвимости программных комплексов, исполняемых на языках программирования C# и Java, а также предложены пути их устранения.

Характерной особенностью руководящего документа является отсутствие каких-либо эффективных предложений для обнаружения закладок, сформулированных в виде требований к работе исследователя. Общая концепция сводится к неизменности сертифицированного программного продукта после выпуска его из испытательной лаборатории [5].

Для поиска программных закладок предлагается:

1) посмотреть на предмет с «разных сторон» (бинарный код, исходный код, документация, блок-схемы с трассировкой);
2) измерить его некоторые параметры (количество переменных, функций, условных операторов и пр.);
3) разложить по полочкам его составляющие (функции отдельно, переменные отдельно);
4) связать межу собой разложенные по полочкам части (т.е. собрать программу назад путем составления маршрутов, связей);
5) лишние детали, оставшиеся после сборки, выкинуть («определить избыточность»);
6) зарисовать, как выглядят части в сборе (блок-схему, диаграмму) и определить не вырисовалось ли что-нибудь опасное [5].

С этого момента документ переходит в плоскость «философии безопасности», давая возможность исследователю проявить свои знания «по взлому» (которых, в общем-то может и не быть), но руководящий документ этого не учитывает. На этой стадии надо уметь отличить вредоносное удаление файла от штатного удаления, контролировать выполнение программы без отладчика и уметь наблюдать за прохождением защищаемой информации в программе, как за радиоактивным атомом в теле человека при диагностике раковых заболеваний.

Опыт работы показал, что для второй части инструменты пока не изобретены. Раскладывание «по полочкам» исходных текстов для десятков сотен и тысяч файлов носит бесполезный характер. А для того, чтобы находить программные закладки, надо в первую очередь их самим писать.

При этом требования руководящего документа выполняются почти всегда корректно, но всегда требуют решения Органа по сертификации о возможности использования предмета исследования, который был таким образом проконтролирован. В ряде испытаний обнаруживается, в первую очередь, избыточность как на уровне исполняемых файлов (*.exe, *.dll), так и исходных текстов. При этом далеко не вся избыточность может быть удалена т.к. большая часть – это динамические библиотеки тех покупных средств, которые использовались разработчиком (ОС Windows, СУБД MS SQL, измерительная аппаратура и т.п.). Избыточность на уровне исходных текстов также должна быть определена и устранена (включая избыточные функциональные объекты).

Руководящий документ, предписывающий проверки по контролю отсутствия НДВ, имеет ряд вышеописанных недоработок. В связи с этим, целесообразно было бы при поставке изделий на объекты заказчика формировать для каждого образца специальное задание на сертификацию, где будут предусмотрены требования к поиску программных закладок, исходя из назначения и технических характеристик всего изделия.

Список использованных источников:

1. Методический документ. Меры защиты информации в государственных информационных системах [Электронный ресурс] URL: https://fstec.ru/component/attachments/download/675 (дата обращения: 16.10.2020).
2. Государственный реестр сертифицированных средств защиты информации [Электронный ресурс] URL: https://fstec.ru/tekhnicheskaya-zashchita-informatsii/dokumenty-po-sertifikatsii/153-sistema-sertifikatsii/591-gosudarstvennyj-reestr-sertifitsirovannykh-sredstv-zashchity-informatsii-n-ross-ru-0001-01bi00 (дата обращения: 16.10.2020).
3. Сертифицированное программное обеспечение. Средства контроля защищенности от НСД. АИСТ-С Анализатор исходных текстов С и С++ программ [Электронный ресурс]. URL: https://www.cbi-s.ru/catalog/programmnye-sredstva-kontrolya-zashchishchennosti-ot-nsd/aist-s-analizator-iskhodnykh-tekstov-c-i-cpp-programm/ (дата обращения: 16.10.2020).
4. Разработки АО «НПО «Эшелон» [Электронный ресурс]. URL: https://www.cbi-s.ru/catalog/programmnye-sredstva-kontrolya-zashchishchennosti-ot-nsd/aist-s-analizator-iskhodnykh-tekstov-c-i-cpp-programm/ (дата обращения: 16.10.2020).
5. Защита от несанкционированного доступа к информации Часть 1. Программное обеспечение средств защиты информации. Классификация по уровню контроля отсутствия недекларированных возможностейРуководящий документ. Приказ председателя Гостехкомиссии России от 4 июня 1999 г. № 114 [Электронный ресурс]. URL: https://fstec.ru/tekhnicheskaya-zashchita-informatsii/dokumenty/114-spetsialnye-normativnye-dokumenty/382-rukovodyashchij-dokument-prikaz-predsedatelya-gostekhkomissii-rossii-ot-4-iyunya-1999-g-n-114 (дата обращения: 16.10.2020).

## 1.3. Разработка предложений по созданию средства осуществления контроля исходных текстов ПО на отсутствие заимствований (плагиата)

В соответствии со статьей 1261 Гражданского кодекса Российской Федерации (ГК РФ) от 18.12.2006 № 230-ФЗ «Программы для ЭВМ», авторские права на все виды программ для ЭВМ, которые могут быть выражены на любом языке и в любой форме, включая исходный текст и объектный код, охраняются так же, как авторские права на произведения литературы. Программой для ЭВМ является представленная в объективной форме совокупность данных и команд, предназначенных для функционирования ЭВМ и других компьютерных устройств в целях получения определенного результата, включая подготовительные материалы, полученные в ходе разработки программы для ЭВМ, и порождаемые ею аудиовизуальные отображения [1].

Согласно постановлению Пленума Верховного Суда РФ от 26.04.2007 № 14 плагиат – это факт нарушения авторских прав путем присвоения авторства [2].

Данный параграф посвящен разработке предложений по созданию средства контроля исходных текстов программного обеспечения на отсутствие заимствований (плагиата). В рамках этого направления можно выделить следующие задачи:

1. анализ существующих предложений, методов и средств обнаружения заимствований;
2. разработка предложений для повышения эффективности существующих методов обнаружения заимствований;
3. создание демонстрационной версии программной реализации представленных предложений.

Результатом работы представленного программного средства будет являться автоматически генерируемый отчет, позволяющий с приемлемой точностью определить количество заимствованного материала из входных данных – исходных текстов.

Определения и обозначения

В работах [3], [4] предлагается преобразовывать исходные тексты в последовательность токенов и анализировать их совпадения. Несмотря на то, что подобный способ эффективнее прямого анализа исходного текста, данный метод контроля достаточно просто обойти, что продемонстрировано в примерах (рис. 1.7 – 1.9).

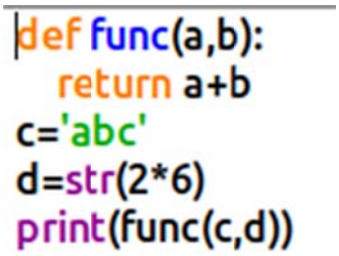



Рисунок 7.7 – исходный текст программы (пример 1)

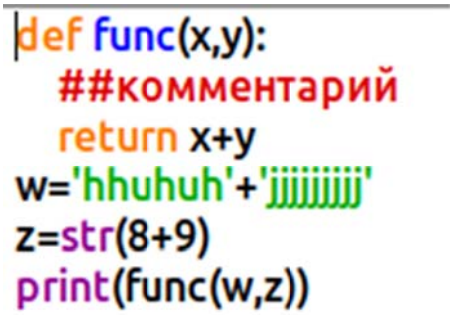



Рисунок 1.8 – исходный текст программы (пример 2)

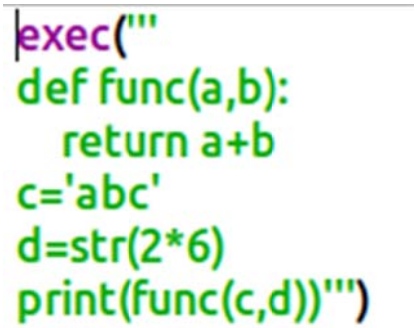



Рисунок 1.9 – исходный текст программы (пример 3)

Рисунки 1.7 – 1.9 демонстрируют примеры исходных текстов для анализа. Следует обратить внимание на то, что несмотря на несущественные отличия в тексте, структуры описываемых программ идентичны.

При реализации предлагаемого метода используются механизмы более глубокого поиска совпадений и их комплексирование с механизмами, описанными ранее. Также представляется демонстрационная версия описываемого средства.

В первую очередь осуществляется контроль совпадений по исходным текстам. Механизм контроля подробно описан в источнике [5]. Реализация анализа исходных текстов из примеров 1, 2, 3 продемонстрирована на рисунках 1.10, 1.11.

Рисунки 1.10, 1.11 наглядно демонстрируют преимущества и недостатки контроля совпадений по тексту: для снижения процента обнаружи-

ваемых совпадений достаточно внести в исходный код изменения, не влияющие на функционал, но изменяющие внешний вид текста (например, переименовать переменные).

```
Обнаружены следующие совпадения:
def func(
):
=str
)
print(func(
))
Процент совпадений:
0.45
```

Рисунок 1.10 – сравнение исходных текстов примеров **1** и 2 по тексту

```
Обнаружены следующие совпадения:
def func(a,b):
    return a+b
c='abc'
d=str(2*6)
print(func(c,d))
Процент совпадений:
0.83
```

Рисунок 1.11 – сравнение исходных текстов примеров **1** и 3 по тексту

Дальнейший анализ подразумевает разбиение исходного текста на поток токенов (сущностей лексики ЯП) и поиск совпадений в их последовательности. Механизмы анализа подробно описаны в источнике [5]. Реализация анализа исходных текстов из примеров 1, 2, 3 продемонстрирована на рисунке 1.12.

Рисунок 6 содержит визуализацию потока токенов для каждого исходного текста примеров 1-3. Заметно практически полное сходство потоков примеров 1 и 2, а также отсутствие совпадений между примерами 1 и 3.

Таким образом, поиск по лексике не выявляет совпадений между примером 1 и примером 3, содержащим в себе полную копию примера 1, что говорит о низкой точности подобного метода анализа в отдельно взятых случаях, и необходимости комплексирования различных методов анализа. Итого, поиск по тексту выявил 83% совпадений между примерами 1 и 3, а поиск по лексике – 0%, из чего можно сделать вывод о необходимости более глубокого поиска совпадений в исходных текстах.

Следующим шагом предлагается построение математической модели на основе потока токенов (синтаксический и семантический анализ).

Результатом является граф, содержащий связи функциональных объектов по управлению и по информации. Анализ построенного графа предоставляет возможности более глубокого поиска совпадений.

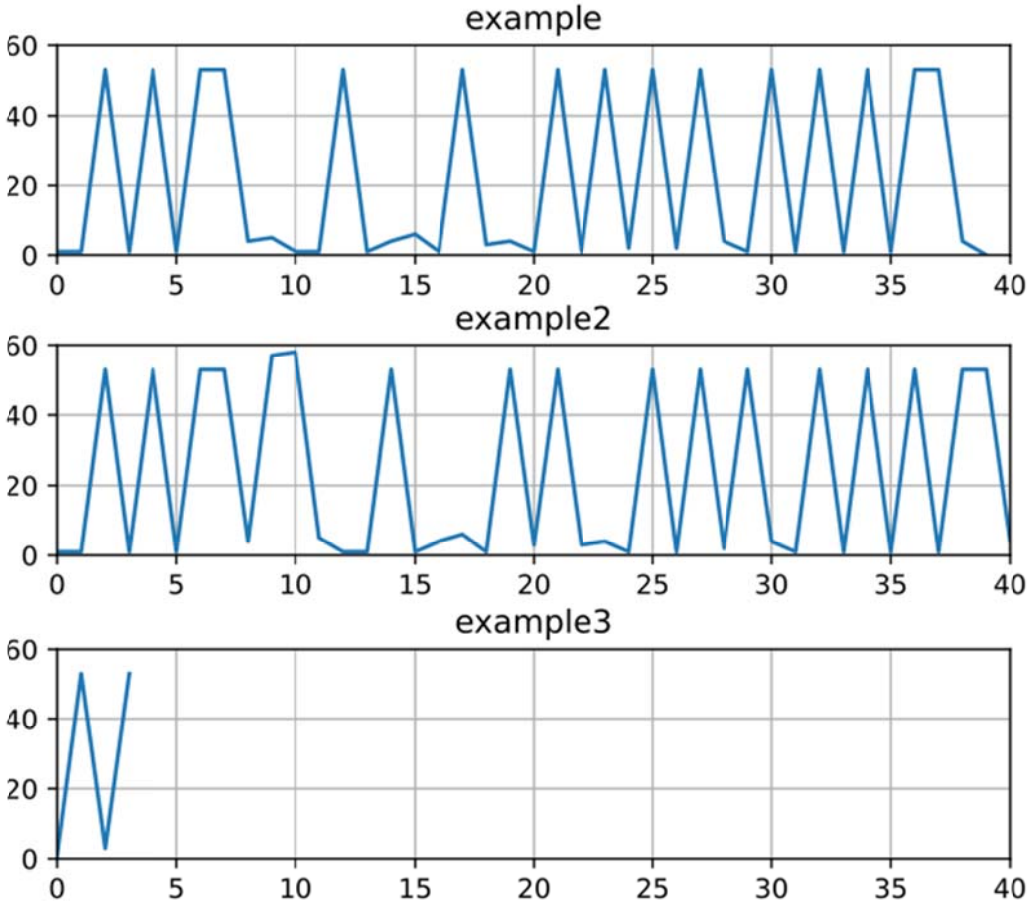



Рисунок 1.12 – сравнение исходных текстов примеров 1-3 по лексике

Пример моделей, сгенерированных на основе исходных текстов, описанных примерами 1 и 3, представлен на рисунках 1.13 и 1.14.

Следует заметить, что граф, изображенный на рисунке 1.14, полностью включает в себя граф, изображенный на рисунке 1.13, что соответствует программной структуре примера 3, полностью включающего в себя пример 1 без каких-либо изменений. Таким образом, пример 1 полностью совпадает с примером 3 по семантике, что является наиболее близким к действительности выводом.

Стоит обратить внимание на возможность снижения точности анализа при увеличении глубины поиска, что было продемонстрировано на рис. 1.13, однако при большем увеличении глубины точность анализа возрастает, а обход контроля заимствований усложняется.

Результатом работы представленного средства является отчет, содержащий информацию о совпадениях, найденных на каждом из этапов контроля.

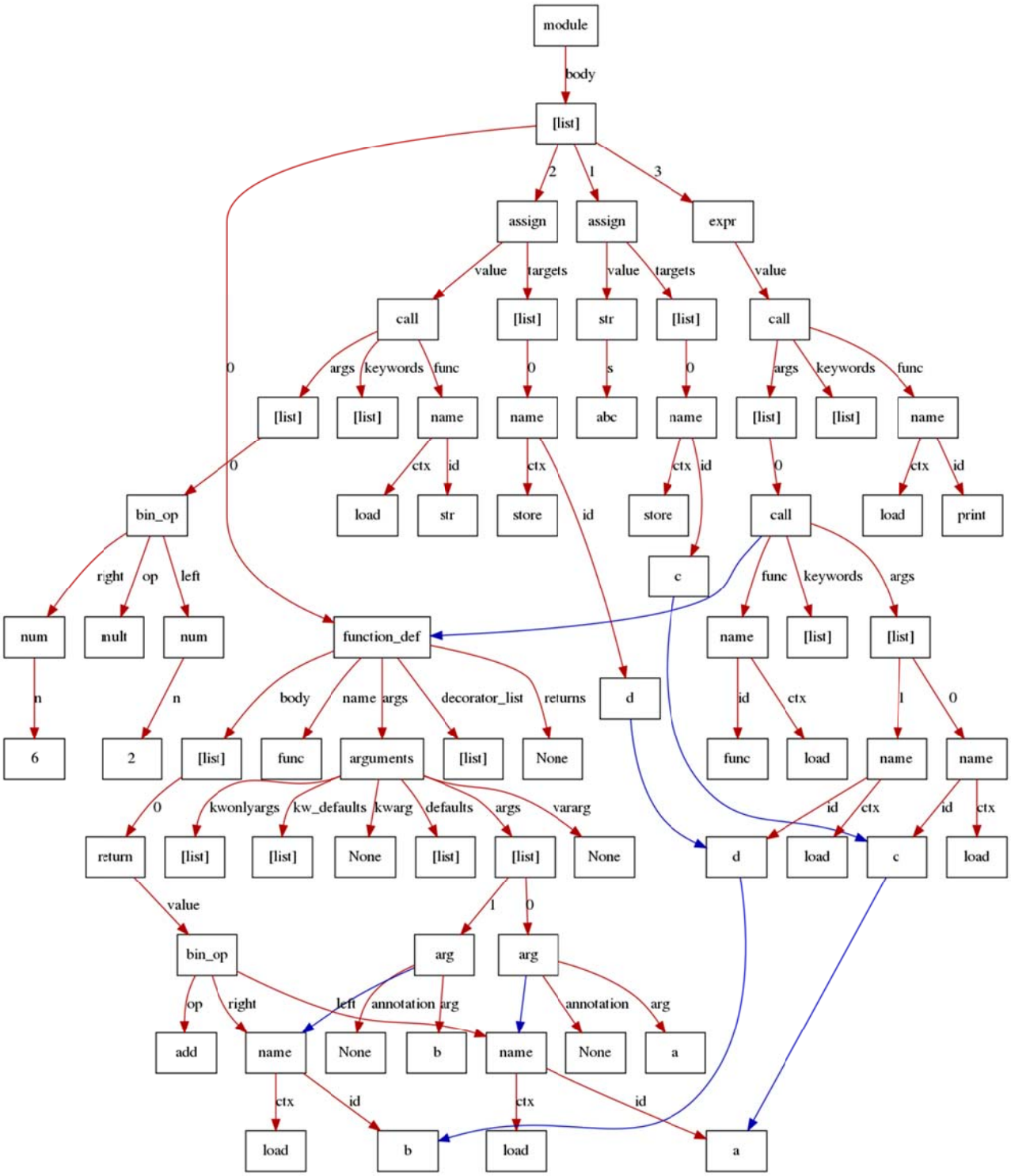



Рисунок 1.13 – Граф, построенный из исходного текста примера 1

Структура отчета, создаваемого демонстрационным вариантом средства контроля:

1. Процент выявленных совпадений по тексту;
   - 83% для примеров 1и 3;
   - 45% для примеров 1 и 2.

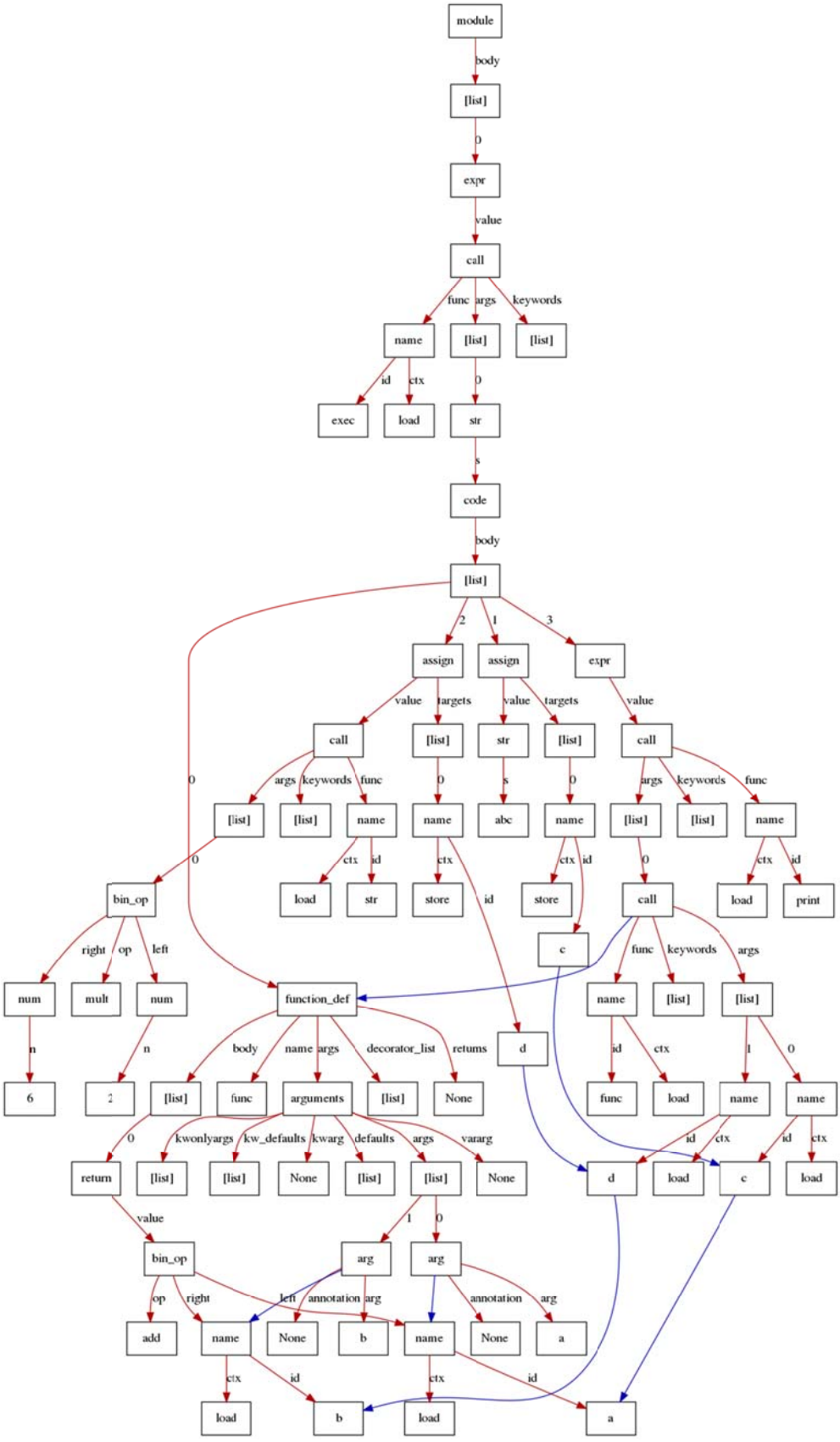



Рисунок 1.14 – Граф, построенный на основе исходного текста примера 3

2. Список выявленных совпадений по тексту;
   - Для примеров 1 и 3, 1 и 2 полученный список указан выше.

3. Процент выявленных совпадений по лексике;
   - 0% для примеров 1 и 3;
   - 88% для примеров 1 и 2.
4. Список выявленных совпадений по лексике;

Представляет собой список сигнатур, составленных из токенов, встречающихся в обоих сравниваемых исходных текстах.

5. Процент выявленных совпадений по семантике;

Процентное соотношение количества изоморфных подграфов, выявленных в обоих сравниваемых графах и общего количества подграфов. В представленной версии средства анализа алгоритм расчета не реализован, вследствие чего данное значение рассчитывается вручную.

Для примеров 1 и 2 приблизительное значение составляет 84%, для примеров 1 и 3 – 97%.

6. Список выявленных совпадений по семантике;

Содержит информацию обо всех выявленных изоморфных подграфах анализируемых графов. В представленной версии средства анализа алгоритм составления данной структуры не реализован, вследствие чего задача выполняется вручную.

Предполагается последующее выполнение специалистом ручного анализа с использованием данных, предоставляемых в отчете.

Описанный вариант программного средства может быть реализован независимо от ЯП, также возможно расширение функционала путем создания базы данных, содержащей сигнатуры большого количества проанализированных открытых исходных текстов. Также возможно повышение точности анализа за счет увеличения глубины поиска путем дальнейших исследований.

Таким образом, проведена разработка предложений по созданию средства контроля исходных текстов ПО на отсутствие заимствований (плагиата), а именно:

- произведен анализ существующих предложений, методов и средств обнаружения заимствований;
- разработаны предложения, обеспечивающие повышение эффективности существующих методов обнаружения заимствований;
- создана демонстрационная версия программной реализации представленных предложений.

Список использованных источников:

1. Гражданский кодекс Российской Федерации (часть четвертая) № 230-ФЗ от 18.12.2006 (ред. от 26.07.2019, с изм. от 24.07.2020) [Электронный ресурс] URL: http://www.consultant.ru/document/cons_doc_LAW_64629/ce1359ed5b9bd99896d7a496c7887e7c223a2cbc/ (дата обращения: 17.10.2020).

2. О практике рассмотрения судами уголовных дел о нарушении авторских, смежных, изобретательских и патентных прав, а также о незаконном использовании товарного знака. Постановление Пленума Верховного Суда РФ от 26.04.2007 № 14 [Электронный ресурс] URL: http://www.consultant.ru/document/cons_doc_LAW_68054/#dst100011 (дата обращения: 17.10.2020).
4. Логинов А.И. Поиск плагиата в исходных текстах программ [Электронный ресурс] URL: https://fpmi.bsu.by/ImgFpmi/Cache/36173.pdf (дата обращения: 17.10.2020).
5. Марганова З.Р. Разработка системы обнаружения плагиата для курса «Функциональное программирование» Бакалаврская работа, Санкт-Петербург, 2015 [Электронный ресурс] URL: https://www.math.spbu.ru/SD_AIS/documents/2015-05-441/2015-05-bs-07.pdf (дата обращения: 17.10.2020).
6. Лифшиц Ю. и др. Обзор автоматических детекторов плагиата в программах [Электронный ресурс] URL: https://logic.pdmi.ras.ru/~yura/detector/survey.pdf (дата обращения: 17.10.2020).

## 1.4. Применение аналитического моделирования к задаче выявления аномалий в сетевом трафике

При поддержке Министерства образования и науки Российской Федерации, соглашение № 05.607.21.0322 (идентификатор RFMEFI60719X0322)

Обеспечение информационной безопасности в телекоммуникационных сетях является актуальнейшей проблемой современного IT-мира [1]. Суть проблемы заключается в сложном взаимообусловленном и взаимопротиворечивом характере сетевого взаимодействия элементов сети между собой и с (агрессивной) внешней средой, полная информация о котором для принятия управленческих решений (динамическая информация) практически отсутствует или представляет собой гигантский массив фактологических данных (статическая информация). Поэтому ее разрешение может лежать, том числе, в плоскости прагматичного анализа текущего состояния сети (фактологии) и прогнозирования ее развития (динамики) во времени, тем самым, предотвращая проведение сетевых атак.

Сетевое взаимодействие представляет собой динамический процесс обмена сетевыми пакетами между совокупностью хостов (далее – Сеть) по определенному набору протоколов. Работа Сети в штатном режиме характеризуется наличием аналитических зависимостей между ее элементами, изменения которых также может считаться закономерным. Так, например, в момент создания сетевого окружения и начала его эксплуатации происходит процесс самоорганизации – по протоколу DHCP хостам назначаются IP-адреса, а маршрутизирующее оборудование обновляет таблицы и прошивки.

Возникновение отклонений от штатного режима работы Сети – т.е. появление аномалий – может свидетельствовать о совершающейся сете-

вой атаке (как внутренней, так и внешней). Выявление атак исключительно по содержимому пакетов сетевого трафика может не всегда оказаться достаточным, поскольку оно не будет учитывать сложные взаимосвязи между хостами, потенциально расположенным на достаточно большом расстоянии (с точки зрения промежуточных пакетов). В то же время злоумышленник использует именно их для проведения распределенных по времени и пространству атак. При этом обнаружение последних с привлечением ручного труда экспертов (даже в случае применения систем поддержки принятия решений) может оказаться недостаточным без применения высокотехнологичных программно-аппаратных решений, облегчающих работу с большими потоками данных.

Далее будет показано, что для решения задачи выявления аномалий в сетевом трафике, в частности при больших объемах данных, целесообразным является использование аппарата аналитического моделирования, который оперирует именно формальными связями между частями сложной информационной системы. Применение подобного моделирования позволит, с одной стороны, использовать упрощенное описание поведения Сети (в противоположность имитационному моделированию), а с другой стороны учитывать существенные для решаемой задачи связи [2-6]. Как результат, состояние и развитие Сети описывается в формализованном виде, к которому возможно применение математических методов (предикатных, вероятностных, массового обслуживания и т.п.).

Основная идея применения аналитического моделирования к задаче выявления аномалий в сетевом трафике заключается в разделении процесса работы на три этапа:

Этап 1. Построение *Обобщенной модели* Сети экспертом, учитывающей общие закономерности (создание шаблона формул).

Этап 2. Построение *Частной модели* Сети по ее штатной работе путем определения аналитических параметров общей модели (уточнение коэффициентов в формулах).

Этап 3. Эксперимент по выявлению аномалий путем вычисления аналитических значений и их сравнение с нормальными значениями путем проверки корректности аналитических формул, полученных на Этапе 1 и уточненных на Этапе 2.

Таким образом, аналитическое моделирование состоит в построении аналитических связей в модели Сети по ее штатной работе и подстановке в модель сетевого трафика исследуемой работы сети – если в модели будет нарушено тождество (т.е. хотя бы один из ее параметров выйдет за допустимые пределы), то это говорит о присутствии аномалии.

Также важным является учет частотно-временных характеристик модели, что позволит сравнивать не только статистические параметры (например, среднее количество используемых портов за месяц), но и ди-

намические (например, график роста сетевой активности на отдельные серверные хосты).

Необходимо отметить, что Сети свойственно изменение во времени, связанное с обновлением оборудования и программного обеспечения, так, например: на части маршрутизирующего оборудования производится обновление прошивок, добавляющее им новые функции; появляются новые подсети, изменяющие граф связей между хостами; некоторые сегменты сети перестают функционировать из-за проводимых ремонтных работ и т.п. Для учета этого на Этапе 3 помимо проверок состояния Сети на предмет наличия аномалий требуется и динамическая корректировка ее параметров (возможно с учетом общего «устаревания» их актуальности).

На Этапе 1 необходимо построить Обобщенную модель Сети, показывающую общие закономерности всех подобных Сетей. Основываясь на существующих научных исследованиях [7-16] данную модель можно представить в виде, отображенном на рис. 1.15.

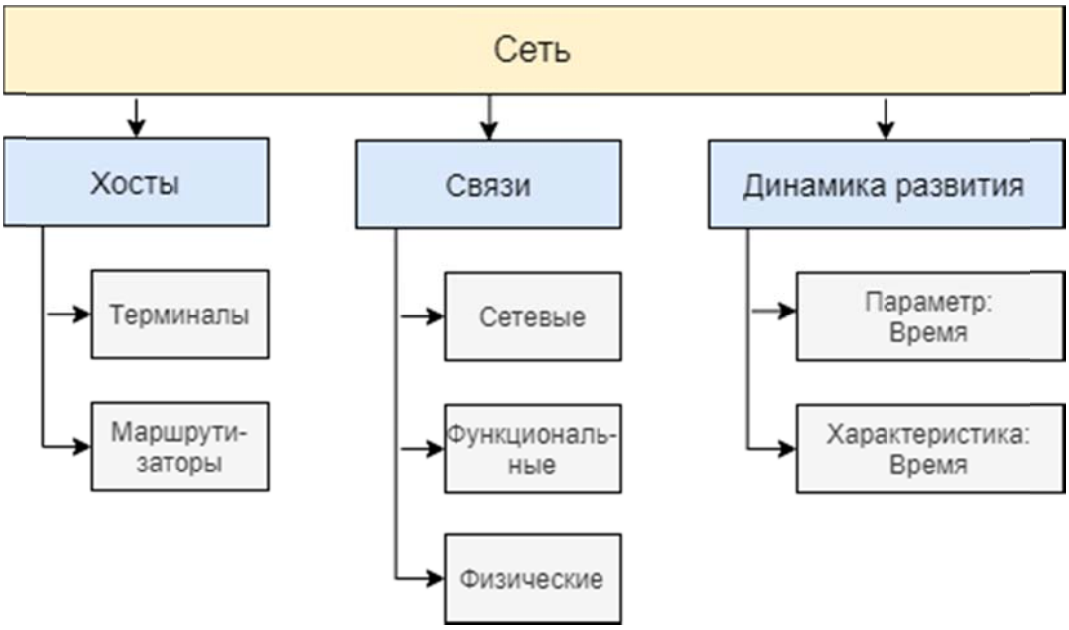



Рисунок 1.15 – Обобщенная модель Сети

Данная модель состоит из следующих взаимосвязанных сущностей:

- Сеть – моделируемая Сеть, состоящая из элементов – хостов и их связей, обладающими неинвариантными (от времени) свойствами;
- Хосты – множество хостов Сети, между которыми происходит взаимодействие, включающие:

a) Терминалы – оконечные хосты, генерирующие и принимающие сетевые пакеты;

b) Маршрутизаторы – промежуточные хосты, т.е. те, которые принимают пакеты, обрабатывают их и передают следующим хостам;

– Связи – совокупность пар хостов, соответствующая отношениям между хостами, включающие типы:

a) Сетевые – связи, определяющие сетевые каналы обмена (например, сетевой кабель);

b) Функциональные – связи, определяющие использование функций одних хостов для выполнения функций других (например, клиент-серверные соотношения);

c) Физические – связи, определяющие физическое, возможно опосредованное, взаимодействие хостов (например, физический запуск одного терминала при выключении другого инженером сети посредством подачи электропитания);

– Динамика развития – совокупность величин, описывающих элементы Сети с позиции их динамического развития (является также общим описанием изменчивости Сети), таких как:

a) Параметр: время – параметр, от которого зависит работа элемента Сети и который изменяется во времени (например, различное функционирование веб-сервера в дневное и ночное время из-за изменения нагрузки на него из внешней сети); таким образом, параметр отражает специфику реального поведения субъектов (пользователей Сети, сознательно вносящих неоднородность в статические закономерности) и включает в себя непосредственное влияние операторов терминальных хостов (например, запуск злоумышленником на хосте программы сканирования);

b) Характеристика: время – характеристика элемента Сети, которая изменяется во времени (например, постепенное ухудшение эксплуатационных характеристик сервера вплоть до полного выхода его из строя); таким образом, данная характеристика отражает специфику реальной работы элементов Сети в общей картине взаимодействия элементов.

В формальном виде Обобщенная модель может быть записана следующим образом:

$$N = \langle E, T \rangle,$$

где $N$ – сеть (Network), $E$ – элементы Сети (Elements), $T$ – динамическое развитие элементов (Time).

А элементы, имеющие вид хостов и их связей, определяются, как

$$E = \langle H, L \rangle,$$

где $H$ – хосты (Hosts), $L$ – связи (Links).

Множество связей может быть задано в виде каждого типа связей, а именно следующим образом:

$$L = L^N \cup L^F \cup L^P,$$

где $L^N$ – сетевые связи (Network Links), $L^F$ – функциональные связи (Functional Links), $L^P$ – физические связи (Physical Links).

Каждая из связей соединяет хосты, а их отличия заключаются лишь в типах соединений; таким образом, они могут быть записаны, как

$$L^l = \langle H, H, LT^l \rangle,$$

где $LT^l$ – тип $l$-ой связи для $l \in \{Network, Functional, Physical\}$.

Динамическое развитие Сети может быть задано в виде совокупности величин, отражающих изменение во времени входных и выходных потоки данных (в т.ч. физических, таких, как сигналы от клавиатур и вывод на монитор) – первые связаны с параметрами элемента, а вторые – с его характеристиками. Развитие может быть описано через *оператор обработки сетевого трафика* $\Delta$ (ООСТ) элементом Сети *E,* принимающий в качестве аргументов:

– входной поток сетевых пакетов $Packets_i^{In}$ на $i$-ый интерфейс;

– элемент Сети $E$;

– зависящие от времени параметры элемента $P(t)$.

Выходными данными оператора $\Delta$ являются сетевые пакеты $Packets_{i_{Out}}^{Out}$ на $i$-ый интерфейс входами-параметрами, который зависят не только от входных пакетов, но и от изменяющихся характеристик элемента *C(t)*:

$$Packets_{i_{Out}}^{Out}(C(t)) = \Delta\left(Packets_{i_{In}}^{In}, E, P(t)\right),$$

где $t$ – некоторая точка времени или их множество (в т.ч. непрерывный диапазон).

Важно отметить, что такой элемент Сети также участвует в обработке пакетов, поскольку в той или иной степени может влиять на их передачу (например, разрыв функциональной связи между базами данных во время их синхронизации приведет и к прекращению обмена сетевыми пакетами).

Т.к. все элементы составляют единую сеть, то выходные пакеты одних операторов являются входными пакетами для других, что в конечном итоге объединяет всю совокупность ООСТ в следующую систему (для простоты представления временная зависимость опущена):

$$\begin{cases} Packets_i = \Delta(\mathrm{Packets_j}, \mathrm{E_k}) \\ Packets_i \equiv Packets_j \text{ если } L_{i,j}^N = 1 \\ Packets_i \equiv Packets_j \text{ если } L_{i,j}^F = 1 \\ Packets_i \equiv Packets_j \text{ если } L_{i,j}^P = 1 \end{cases},$$

где $E_k$ – $k$-ый элемент Сети; $Packets_i$/ $Packets_j$ – i-ый и j-ый пакеты (как выходной в элемент, так и выходной из него); $L_{i,j}^N/L_{i,j}^F/L_{i,j}^P$ – матрицы для сетевых, функциональных и физических связей (в содержание 1, если такая связь есть, и 0, если такой связи нет).

Таким образом, с помощью данной системы аналитическая модель является замкнутой сама на себя, показывая общие закономерности в функционировании сети (т.е. в некотором роде закон сохранения информации). Это дает возможность выявлять аномалии по выходу значений некоторых ее элементов за пределы закономерностей, нарушая тождественность (истинность) аналитических уравнений.

Также, поскольку для задачи выявления аномалий в Сети очевидно не обязательна полная информация о передаваемых пакета вплоть до значения всех их полей, то $Packets^{In}/Packets^{Out}$ могут являться лишь совокупностью необходимых признаков или меток пакетов.

Точная форма ООСТ элементов Сети в большинстве случаев является достаточно сложной и не формулируемой. Тем не менее, для установления алгоритмов работы операторов и их настройки как раз и предназначен следующий этап.

На Этапе 2 производится построение Частной модели путем установления параметров Обобщенной на основании исследования информации о штатном режиме работы Сети – подключении хостов, возникновении новых связей, определении сезонности в работе элементов Сети и пр. Исходя из того, что сетевые потоки в современных сетях представляют большой объем данных, а также из-за сложности определения даже вида ООСТ, наиболее целесообразным может быть применение области искусственного интеллекта, а точнее набора методов машинного обучения. Так, ООСТ может представлять собой в конечном итоге искусственную нейронную сеть, прогнозирующую появления некоторого набора признаков сетевых пакетов ($Packets_{Out}$) при обработке элементом Сети других пакетов ($Packets_{Out}$).

В результате выполнения данного этапа аналитическая модель за счет уточнения всех ее неизвестных параметров (коэффициентов, таблиц, диапазонов и т.п.) будет в виде некоторого тождественного соотношения описывать штатное функционирование в следующем виде:

$$\forall t, t \in [0..t_0]\text{: для } N(t) \begin{cases} E \notin \emptyset \\ C(t) \notin \emptyset \\ P(t) \notin \emptyset \end{cases},$$

означающем, что для любого момента(ов) времени в период штатного функционирования $[0..t_0]$ аналитическая модель Сети $N(t)$ корректна.

На Этапе 3 производится непосредственная оценка текущего состояния Сети на предмет наличия аномалий с помощью проверки следующего условия:

$$\exists t, t > t_0\text{: для } N(t) \begin{cases} E \notin \emptyset \\ C(t) \notin \emptyset, \\ P(t) \notin \emptyset \end{cases}$$

означающего, что существует момент(ы) времени после периода штатного функционирования ($t > t_0$), когда аналитическая модель Сети $N(t)$ не корректна.

Корректность аналитической модели тут означает именно непротиворечивость входящих в нее формул – за счет ограничений на параметр *t* до (т.е. в процессе обучения) и после штатного функционирования (т.е. в процессе выявления аномалий).

Для пояснения введенного понятия *корректности* модели приведем следующий пример. Пусть модель представляет собой простейшую систему линейных уравнений следующего вида, полученную на Этапе 1:

$$\begin{cases} Y = X + t \\ X = Y - t \\ X > 10 \end{cases},$$

где $Y$ – плотность сетевого потока, $X$ – как характеристика хоста с позиции генерации сетевых пакетов, а $t$ – некоторое время.

Первое уравнение соответствует выходящему из хоста сетевому потоку, второе – сетевому трафику, управляющему хостом, а третье – утверждению, что объем исходящего трафика больше 10; тогда уравнения корректны при любых t.

Однако если вследствие действий злоумышленника происходит обновление программного обеспечения узла добавлением на него сканера портов (например, за счет уязвимостей в коде телекоммуникационного оборудования [17]), тогда хост начинает рассылать избыточное количество пакетов (изменяется первое уравнение), не изменяя при этом управляющий хостом сетевой поток (во втором уравнении); третье же уравнение также не изменяется, поскольку точные значения в нем были получены на Этапе 2. Таким образом, исходная система линейных уравнений преобразуется к следующей (предположим, что сканер портов увеличил количество отправляемых пакетов в 2 раза):

$$\begin{cases} Y = 2 \times X + t \\ X = Y - t \\ X > 10 \end{cases}.$$

Тогда, при подстановке значения X из второго уравнения в первое, мы получим:

$$\begin{cases} Y = t \\ X = 0 \\ X > 10 \end{cases},$$

т.е. аналитическое описание сети становится некорректным ($X = 0 \not> 10$).

Приведем гипотетические примеры реальных аномалий и их отражение в аналитической модели.

Во-первых, если в Сети появляется новый элемент – хост или соединение (в реальности, их некоторое множество и за некоторые периоды времени), то параметры аналитической модели могу выйти за допустимые пределы из-за резкого увеличения ООСТ и последующих новых каналах обмена. Т.е. на всем протяжении штатного режима (на любом доступном *t*) такие параметры не достигались. Такая ситуация характерна, когда злоумышленник подключается к Сети напрямую и пытается осуществить атаку.

Во-вторых, если элемент Сети не сможет получить требуемых параметров (например, доступной оперативной памяти), то это выходит за пределы допустимых характеристик по реагированию на запросы, что делает аналитическую модель некорректной – т.к. ни в какой момент работы в штатном режиме сервер не находился в таком состоянии. Такая ситуация характерна для DDoS атак.

В-третьих, ситуация выхода характеристик элемента Сети за допустимые пределы полностью аналогична предыдущему примеру (например, если с хоста идет чрезмерно большой трафик наружу периметра, чего не наблюдалось ранее). Соответственно, если текущее состояние Сети не соответствует штатному, то это означает появление аномалии в ее сетевом трафике.

Исходя из сложности экспертной реализации модели в виде точных математических формул, наиболее целесообразным может быть применение в модели методов машинного обучения. В данном случае, обучающими выборками будет являться штатный режим работы, а результатом работы методов – предположения касательно аномальности сетевого трафика.

Предложенное аналитическое моделирование позволяет анализировать изменяющиеся во времени информационные потоки в сложных системах сетевых хостов на предмет выявления аномалий – отклонений в работе Сети от штатного режима.

Достоинством такого моделирования можно считать отсутствие необходимости создания точной модели реальной физической системы, возлагая детали ее реализации на область машинного моделирования (включая алгоритмы и модели соответствующих методов). Недостатком же является сложность формирования элементов модели, в особенности не имеющих физического прообраза, таких как логические взаимосвязи.

Продолжением исследования может стать практическая реализация системы описанного моделирования, а также проведение натурных экспериментов с целью получения оценок эффективности и границ применимости. В дальнейшем возможно развитие аналитического моделирования на более высокие уровни абстракции информационного обмена, вплоть до архитектур и концепций целых телекоммуникационных сетей [21-25].

## Список использованных источников:


1. Защита информации в компьютерных системах: монография / [М.В.Буйневич и др.]; под ред. Е.В.Стельмашонок, И.Н.Васильевой; М-во образования и науки Рос. Федерации, Санкт-Петербургский гос. экономический ун-т, Кафедра вычислительных систем и программирования – Санкт-Петербург : Изд-во СПбГЭУ, 2017 – 163 с.
2. Максутов В.А., Кокоулин А.Н. Аналитическое моделирование локальных сетей // Сборник научных трудов SWorld. 2014. Т. 5. № 3. С. 86–94.
3. Пушнин А.В., Косенко Е.Ю. Аналитическое решение задачи оптимизации структур распределенных сетей // Известия ТРТУ. 2004. № 8 (43). С. 52.
4. Вакуленко С.А. Аналитическая модель большой сети: динамика и устойчивость // Вестник Санкт-Петербургского государственного университета технологии и дизайна. Серия 1: Естественные и технические науки. 2010. № 1. С. 3642.
5. Смолькина Е.Е., Остапенко А.Г., Баранников Н.И., Батаронов И.Л. Аналитические вероятностные модели реализации атак на DNS-серверы // Информация и безопасность. 2013. Т. 16. № 4. С. 596603.
6. Ермилов Е.В., Калашников А.О. К вопросу о построении математических моделей атак на АСУ критически важных объектов // Информация и безопасность. 2013. Т. 16. № 1. С. 135136.
7. Тамп В.Л., Тамп Н.В. Аналитическая модель формирования и отображения динамических переменных, отражающих состояние исследуемых вычислительных сетей // Наукоемкие технологии. 2015. Т. 16. № 12. С. 8085.
8. Сумин В.И., Лебедев С.А., Дубровин А.С., Обухова Л.А., Юрасов В.Г. Аналитическое моделирование сегментов информационной сети с коммутацией пакетов // Информация и безопасность. 2006. Т. 9. № 2. С. 165173.
9. Подольский Д.В. К вопросу выбора аналитической модели функционирования звена сети следующего поколения // T-Comm: Телекоммуникации и транспорт. 2010. Т. 4. № 7. С. 4345.
10. Иванов М.В., Филимонов П.А. Модель сети интернет на уровне автономных систем в виде безмасштабного графа // Телекоммуникации. 2016. № 11. С. 2226.
11. Андреев А.М., Ключарев П.Г., Джаммул С.М., Бабиченко А.В. Анализ моделей трафика сетей передачи данных // Прикладная физика и математика. 2018. № 4. С. 4251.
12. Михалевич И.Ф. Структурный анализ элементов мультисервисных сетей связи // Телекоммуникации. 2007. № 2. С. 29.
13. Кулешов И.А., Расчесова А.Г., Казакевич Е.В., Глуховченко Р.Н. Сравнительная оценка сети и ее элементов по показателю эффективности // Информация и космос. 2006. № 4. С. 3437.
14. Звягинцев М.В., Маслов Д.С., Соколов Н.А. Выбор структуры сети связи с учетом жизненного цикла ее элементов // Электросвязь. 2010. № 8. С. 3336.
15. Бердова Ю.С. Всемирная сеть интернет: доступы к сети и основные каналы связи // Экономика и предпринимательство. 2015. № 101 (63). С. 10821087.
16. Платунова С.М. Модель корпоративной сети при агрегировании каналов и резервировании линий // Вестник компьютерных и информационных технологий. 2011. № 2 (80). С. 5155.
17. Буйневич М.В., Израилов К.Е., Мостович Д.И., Ярошенко А.Ю. Проблемные вопросы нейтрализации уязвимостей программного кода телекоммуникационных устройств // Проблемы управления рисками в техносфере. 2016. № 3(39). С. 81-89.

18. Буйневич М.В., Израилов К.Е. Утилита для поиска уязвимостей в программном обеспечении телекоммуникационных устройств методом алгоритмизации машинного кода. Часть 1. Функциональная архитектура [Электронный ресурс] // Информационные технологии и телекоммуникации. 2016. Т. 4. № 1. С. 115-130.
19. Израилов К.Е. Утилита для поиска уязвимостей в программном обеспечении телекоммуникационных устройств методом алгоритмизации машинного кода. Часть 2. Информационная архитектура [Электронный ресурс] // Информационные технологии и телекоммуникации. 2016. Т. 4. № 2. С. 86-104.
20. Израилов К.Е., Покусов В.В. Утилита для поиска уязвимостей в программном обеспечении телекоммуникационных устройств методом алгоритмизации машинного кода. Часть 3. Модульно-алгоритмическая архитектура [Электронный ресурс] // Информационные технологии и телекоммуникации. 2016. Т. 4. № 4. С. 104-121.
21. Буйневич М.В., Владыко А.Г., Израилов К.Е., Щербаков О.В. Архитектурные уязвимости моделей телекоммуникационных сетей [Электронный ресурс] // Научно-аналитический журнал «Вестник Санкт-Петербургского университета Государственной противопожарной службы МЧС России». 2015. № 4. С. 86-93.
22. Buinevich M., Fabrikantov P., Stolyarova E., Izrailov K., Vladyko A. Software defined internet of things: cyber antifragility and vulnerability forecast // Application of Information and Communication Technologies (AICT-2017). 2017. pp. 293-297.
23. Buinevich M., Izrailov K., Pokusov V., Sharapov S., Terekhin S.N. Generalized interaction model in the information system // International Journal of Pure and Applied Mathematics. 2018. Vol. 119. Iss. 17d. pp. 1381-1385.
24. Buinevich M., Izrailov K., Stolyarova E., Vladyko A. Combine method of forecasting VANET cybersecurity for application of high priority way // 20th International Conference on Advanced Communication Technology (ICACT-2018). 2018. pp. 266-271.
25. Mescheryakov S., Shchemelinin D., Izrailov K., Pokussov V. Digital cloud environment: present challenges and future forecast // Future Internet. 2020. Vol. 12. Iss. 5. pp. 82.

## 1.5. Применение алгоритмов искусственного иммунитета для обнаружения вторжений

Информационная безопасность в цифровую эпоху играет в системе национальной безопасности ключевую роль, так как от нее зависит сохранность данных и возможность коммуникации и координации между органами власти и обществом. Критическая информационная инфраструктура (КИИ) является связующим звеном между другими секторами национальной инфраструктуры. Поэтому нанесение ущерба КИИ может привести к катастрофическим последствиям. Также переход информационно-коммуникационных технологий (ИКТ) на систему цифровых сигналов упрощает и частично автоматизирует управление процессами, но, в то же время, делает их более уязвимыми перед компьютерными атаками. Подход, в результате которого вероятность угрозы как серьезной атаки на КИИ может быть снижена, является возможность выхода из постоянного

цикла изучения атак на информационную систему после их обнаружения и переход к восприятию атак изнутри, дальнейшему их предотвращению до того, как они произойдут [2].

Повышение уровня эффективности обнаружения сетевых атак за счет системы обнаружения вторжений (СОВ) – Intrusion Detection System (IDS), способной адаптироваться к изменчивому поведению сети и иметь при этом низкую частоту ложных срабатываний на базе искусственного иммунитета является актуальным. Исследования в области компьютерной безопасности свидетельствуют о возможности использования иммунной системы в качестве аналогии для IDS.

Область применения алгоритмов искусственного иммунитета основана на моделировании элементов из естественных иммунных систем, обнаруженных у животных. Цель алгоритмов искусственной иммунной системы (ИИС) – Artificial Immune Systems (AIS) – состоит в том, чтобы воспользоваться преимуществами адаптивных возможностей естественных иммунных систем, а также использовать способность запоминания довольно сложных паттернов. В [2] авторы отмечают три основные теории ИИС: клональный отбор, аффинность, отрицательный отбор.

Целью алгоритма отрицательного отбора является обеспечение толерантности к самоантигенам. Он развивает способность к дифференциации вредных антигенов. После определения нормального паттерна образуются антигены, которые обнаруживают аномалии случайных антигенов. Этот алгоритм является моделью для нормального и аномального процессов создания (созревания) детекторов. Это позволяет AIS выявлять незрелые детекторы.

На рисунке 1.16 представлены детекторы (малые круги), нанесенные на график или размещенные в пространстве объектов, которое представляет собой данные, например, в виде двоичных битов. Это пространство отображает атрибуты объектов векторов. Все внутри синего круга представляет собой нормальную область. Следующий шаг алгоритма заключается в удалении всех детекторов, касающихся круговой синей линии, в итоге получается только аномальная область, и вследствие чего можно классифицировать аномалии.

Алгоритм отрицательного отбора может быть использован для формирования популяции нейросетевых детекторов, которые в свою очередь могут быть протестированы на определенном наборе данных. При условии обнаружения детектором аномалии на данном наборе, происходит его уничтожение и запрос на создание нового. По итогу вышеупомянутых действий, может быть получена популяция детекторов, не реагирующих на нормальные (легитимные) события. Таким образом, на основе алгоритма отрицательного отбора может быть сформирована рабочая популяция детекторов обнаружения аномалий.

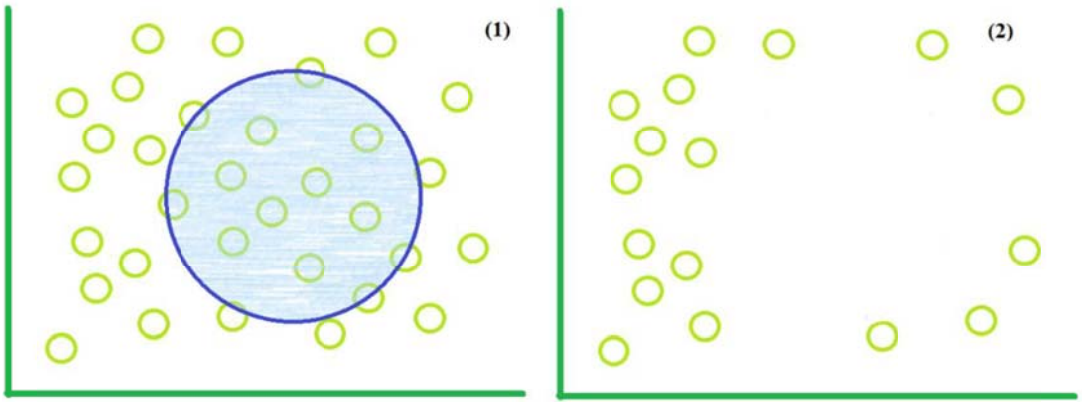



Рисунок 1.16 – Процесс работы алгоритма отрицательного отбора

Принципы работы СОВ могут быть основаны на искусственных иммунных системах. Структурная схема системы обнаружения вторжений показана в виде ее компонентов (модулей) на рисунке 1.17.


Рисунок 1.17 – Структурная схема системы обнаружения вторжений

*Модуль сбора трафика и формирование статистического анализа.* Модуль построен таким образом, чтобы выделять выбранные критерии

данных полученных с узла сети, на котором расположена СОВ. В следствие может быть составлена статистика за выделенный период времени, что позволяет создает вектор из 12 координат [1]:

1) количество запросов на неразрешенные порты UDP;
2) количество пакетов IP, TCP, UDP;
3) количество собранных запросов по протоколу UDP;
4) количество пропущенных вопросов по протоколу UDP;
5) количество необработанных запросов по протоколу UDP, у которых истек срок ответа;
6) количество вопросов к портам TCP;
7) количество запросов на разрешенные порты TCP;
8) количество соединений TCP, FIN SEND за единицу времени;
9) количество соединений TCP, SYN SEND;
10) количество соединений TCP, ESTABLISHED;
11) отношение запросов разрешенных портов TCP к общему количеству всех запросов TCP;
12) соотношение открытых соединений ко всему количеству соединений.

Дальнейший шаг – это анализ модуля обучения.

*Модуль обучения* основан на алгоритме отрицательного отбора [3] в целях создания набора детекторов, которые специально обучаются под выбранную сеть. Этот набор детекторов должен с высокой вероятностью, минимальным количеством ложноположительных реакций показывать какой именно трафик для используемой сети является аномальным, а какой нормальным. Модуль обучения состоит из генерации, обучения и отбора детекторов. Главной концепцией любых разрабатываемых детекторов аномалий считается приравнивание данных, созданного эталона к общей среде обитания и сигнализирование о выходе за определенный барьер значений. Вне зависимости от происхождения системы она должна иметь такой эталон. Обучение нейронов происходит таким образом, который позволяет располагать сами нейроны в рамках распределения выборки.

*Модуль обнаружения вторжений* (*аномалий*) получает отчеты от первого модуля (сбора трафика и формирования статистически анализа), анализирует их, по средствам передачи отчетов на вход каждого детектора из рабочей группы. При условии, если один из детекторов считает статистику отличной от нормальной, аномалия будет обнаружена. Модуль обнаружения содержит следующие составляющие: обнаружение и формирование иммунной памяти (мутация и клонирование детекторов).

Векторы с собранной информационной статистикой о трафике подаются на вход детектора в режиме обнаружения вторжений.

*Модуль оповещения*. В ходе работы системы детекторы либо находят аномалии, либо нет, и при условии их нахождения для наглядности работ-

нику службы безопасности сообщается о нарушении. На рисунке 1.18 представлен процесс обучения системы на выборке построенной на основе нормальных записей.

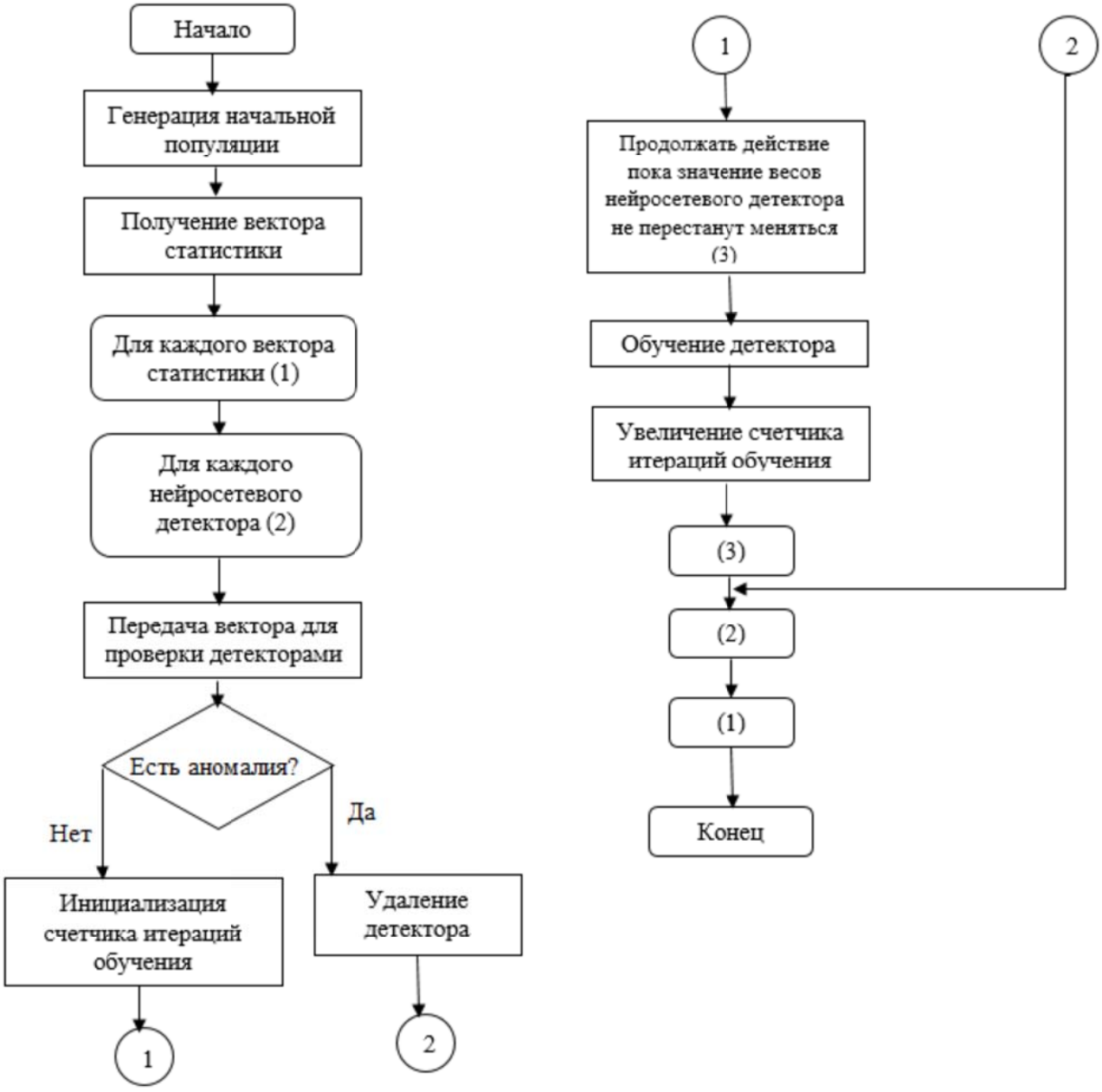



Рисунок 1.18 – Режим обучения детекторов

Система генерирует рабочую группу лимфоцитов, созданную на основе рандомной генерации нейросетевых детекторов. Созданный набор покрывает всю допустимую область значений. Второй шаг включает в себя захват трафика и формирование на основе него ключевых характеристик.

Все нейросетевые детекторы принимают на вход вектор статистики. Именно здесь включается механизм отрицательного отбора, т.е. детекторы, обнаружившие аномальную активность, там, где ее быть не должно,

удаляются, остальные сохраняются. Так создается рабочий набор обученных детекторов и правильно функционирующих [4]. В этот момент система уже имеет созданную для работы популяцию детекторов. На вход нейросетевых детекторов передаются сформированные, на основе собранного с узла сети трафика, векторы статистики. При условии, что детектор сортирует полученную статистику как нормальную его время «жизни» растет, то есть возникает возможность видеть актуальность каждого из детекторов.

Период жизни детектора напрямую зависит от его работоспособности, т.е. найденных им аномалий. Большой период жизни лимфоцита сигнализирует, что за определенный период времени не было обнаружено аномального трафика, и, следовательно, детектор может работать неверно. Вследствие этого происходит его удаление при выходе за определенный барьер. Пороговое значение равно 1000 циклам обработки вектора статистики. Когда детектор обнаруживает аномалию, время его жизни обнуляется и начинается процесс создания 5 его клонов. Клоны подвергаются мутации (изменению весов детектора на небольшую величину). Такой процесс необходим для того, чтобы аномалии сходные с обнаруженной были бы выявлены. Таким образом формируется иммунная память. Видоизмененные аномалии могут быть так же обнаружены детекторами.

Таким образом, эффективность предотвращения угрозы зашифрованных вредоносных программ для КИИ должна основываться на стратегии с учетом методов искусственного интеллекта, машинного обучения и анализа угроз для выявления подозрительных моделей поведения.

Список использованных источников:

1. Васютин С.В., Завьялов С.С. Нейросетевой метод анализа последовательности системных вызовов с целью обнаружения компьютерных атак и классификации режимов работы приложений [электронный ресурс]. URL: http://old.lvk.cs.msu.su/files/mco2005 /vasytin.pdf (дата обращения: 19.10.2020).
2. Гатчин Ю.А., Сухостат В.В. Информационная безопасность критической информационной инфраструктуры: теоретико-методологические аспекты // Инновационные, информационные и коммуникационные технологии: сборник трудов XVII Международной научно-практической конференции. / под.ред. С.У.Увайсова – Москва: Ассоциация выпускников и сотрудников ВВИА им. проф. Жуковского, 2020. – 472 с. С. 213-217.
3. Cannady J. Artificial Neural Networks for Misuse Detection [электронный ресурс] URL: http://csrc.nist.gov/nissc/1998/proceedings/paperF13 (дата обращения: 12.10.2019).
4. Официальный сайт для разработчиков Хабр // Нестандартная классификация 5: Growing Neural Gas [электронный ресурс]. URL: https://habr.com/ru/post/340360/ (дата обращения: 12.05.2020).

# ГЛАВА 2. ОБЕСПЕЧЕНИЕ БЕЗОПАСНОСТИ ЦИФРОВЫХ ТЕХНОЛОГИЙ

## 2.1. Использование платформы Firebase для аутентификации пользователей

Большинству приложений необходимо знать личность пользователя. Проблема аутентификации пользователя стоит достаточно остро. Знание личности пользователя позволяет приложению безопасно сохранять его данные в облаке и обеспечивать одинаковую персонализированную работу на всех устройствах этого пользователя.

В настоящее время существует ряд программ, осуществляющих аутентификацию пользователей. Среди них можно выделить платформу «Firebase Authentication», рассмотрению которой посвящен данный параграф.

«Firebase Authentication» является частью общей платформы Firebase. FastFirebase – это мобильная платформа, которая помогает быстро разрабатывать высококачественные приложения, расширять базу пользователей, и, следовательно, повышать прибыль от продажи и использования приложений.

Firebase Authentication предоставляет серверные службы, простые в использовании SDK и готовые библиотеки пользовательского интерфейса для аутентификации пользователей в приложении. Он поддерживает аутентификацию с использованием паролей, номеров телефонов, популярных федеративных поставщиков удостоверений, таких как Google, Facebook и Twitter, и других.

Firebase Authentication тесно связан с другими службами Firebase и использует отраслевые стандарты, такие как OAuth 2.0 и OpenID Connect, поэтому его можно легко интегрировать с пользовательским сервером.

Аутентификация Firebase позволяет выполнять вход в приложение с помощью системы, с которой пользователи хорошо знакомы. После этого приложение может сохранять данные и персональные настройки пользователя в защищенном облачном хранилище и обеспечивать доступ к ним на всех его устройствах.

Firebase обеспечивает службы серверной части, простые пакеты разработчика и готовые библиотеки интерфейса для аутентификации пользователей различных приложения на любых платформах. Аутентификацию можно выполнять при помощи паролей и интегрированных систем идентификации – Google, Facebook, Twitter и др. Это значительно упрощает процедуру входа в приложение и обеспечивает надежную защиту.

Firebase Authentication имеет следующие основные особенности [1].

– Тесная интеграция с другими функциями Firebase.

- Применение отраслевых стандартов, таких как OAuth 2.0 и OpenID Connect существенно упрощает интеграцию с серверным кодом.
- Использование двух вариантов для разработки: FirebaseUI – полностью интегрируемое универсальное решение для выполнения аутентификации – либо пакет Firebase Authentication SDK, который позволяет вручную интегрировать один или несколько методов входа в приложение.
- Безопасность аутентификации и удобство обеспечиваются за счет возможности выполнять вход через аккаунт Google и другие интегрированные системы идентификации – Facebook, Twitter и т. д. (рис. 2.1). Также для входа можно использовать пароль.
- Удобная рабочая среда с мгновенным доступом к вашему сайту на любых устройствах. Это позволяет удерживать внимание пользователей в течение дня.
- Безопасный доступ к сервисам Google. Возможность сохранять файлы на Google Диск, создавать мероприятия в Google Календаре и делиться новостями со своими контактами прямо из приложения.


Рисунок 2.1 – Сервисы, используемые Firebase

Чтобы подписать пользователя в приложение, следует сначала получить учетные данные для аутентификации от пользователя. Эти учетные данные могут быть адресом электронной почты и паролем пользователя или токеном OAuth от федеративного поставщика удостоверений. Затем эти учетные данные передаются в SDK Firebase Authentication. Серверные службы Firebase Authentication проверяют эти учетные данные и возвращают ответ клиенту.

После успешного входа можно получить доступ к основной информации профиля пользователя и контролировать его доступ к данным, хранящимся в других продуктах Firebase. Также можно использовать предоставленный токен аутентификации для проверки личности пользователей в собственных серверных службах.

Следует отметить, что по умолчанию аутентифицированные пользователи могут читать и записывать данные в базу данных Firebase Realtime и облачное хранилище. Однако существует возможность контролировать доступ этих пользователей, изменив свою базу данных Firebase Realtime и правила безопасности облачного хранилища.

Firebase сертифицирована в соответствии с основными стандартами конфиденциальности и безопасности [2]. Все сервисы Firebase (кроме App Distribution и Crashlytics) успешно прошли процесс оценки ISO 27001 и SOC 1, SOC 2 и SOC 3, а некоторые также прошли процесс сертификации ISO 27017 и ISO 27018 (рис. 2.2).

| Наименование услуги | ISO 27001 | ISO 27017 | ISO 27018 | SOC 1 | SOC 2 | SOC 3 |
|---|---|---|---|---|---|---|
| База данных Firebase в реальном времени | ✓ | | | ✓ | ✓ | ✓ |
| Динамические ссылки Firebase | ✓ | | | ✓ | ✓ | ✓ |
| Лаборатория тестирования Firebase | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Мониторинг производительности Firebase | ✓ | | | ✓ | ✓ | ✓ |
| Облачное хранилище для Firebase | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Облачные функции для Firebase | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Обмен сообщениями Firebase в приложении | ✓ | | | ✓ | ✓ | ✓ |
| Обмен сообщениями Firebase Cloud | ✓ | | | ✓ | ✓ | ✓ |
| Платформа Firebase | ✓ | | | ✓ | ✓ | ✓ |
| Проверка подлинности Firebase | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Прогнозы Firebase | ✓ | | | ✓ | ✓ | ✓ |
| Распространение приложений Firebase | | | | | ✓ | |
| Удаленная конфигурация Firebase | ✓ | | | ✓ | ✓ | ✓ |
| Хостинг Firebase | ✓ | | | ✓ | ✓ | ✓ |
| Cloud Firestore | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Firebase A / B-тестирование | ✓ | | | ✓ | ✓ | ✓ |
| Firebase Crashlytics | | | | | ✓ | |
| Firebase ML | ✓ | | | ✓ | ✓ | ✓ |
| Google Analytics для Firebase | ✓ | | | ✓ | ✓ | ✓ |

Рисунок 2.2 – Сертификаты Firebase

Firebase автоматически шифрует все данные перед их записью. Данные автоматически дешифруются при чтении авторизованным пользователем.

Благодаря шифрованию на стороне сервера [3], Google управляет криптографическими ключами от имени пользователя, используя усиленные системы управления ключами. Данные и метаданные каждого объекта Firebase зашифрованы в соответствии с 256-битным AES-шифром, и каждый ключ шифрования сам зашифрован с помощью регулярно меняющегося набора главных ключей.

Шифрование на стороне сервера можно использовать в сочетании с шифрованием на стороне клиента. При шифровании на стороне клиента он управляет своими собственными ключами шифрования и шифрует данные перед их записью в Firebase. В этом случае данные клиента шифруются дважды: один раз ключами клиента и один раз ключами Google.

Чтобы защитить данные при их передаче через Интернет во время операций чтения и записи, Firebase использует протокол TLS.

Рамки Privacy Shield предоставляют механизм для соблюдения требований защиты данных при передаче личных данных из ЕЭЗ, Великобритании или Швейцарии в США и др.

В то время как Соглашение об обмене конфиденциальной информацией между Швейцарией и США в настоящее время остается в силе, в свете недавнего постановления Суда Европейского Союза о передаче данных, признающего недействительным Соглашение об обмене конфиденциальной информацией между ЕС и США, Firebase перешла на Стандартные договорные положения для передачи соответствующих данных. Этот документ, согласно постановлению, может оставаться действующим юридическим механизмом для передачи данных в соответствии с GDPR.

Список использованных источников:

1. Аутентификация Firebase [электронный ресурс]. URL: https://firebase.google.com/docs/auth/?hl=ru (дата обращения 17.09.2020).
2. Конфиденциальность и безопасность в Firebase [электронный ресурс]. URL: https://firebase.google.com/support/privacy/?hl=ru-UA (дата обращения 18.09.2020).
3. Server-side encryption [Электронный ресурс]. URL: https://cloud.google.com/firestore/docs/server-side-encryption (дата обращения 20.09.2020).

## 2.2. Особенности обеспечения безопасности NoSQL баз данных

NoSQL – это подход к реализации базы данных с моделью данных, отличающийся от классической реляционной модели. NoSQL-базы данных ориентированы на хранение и обработку больших объемов данных с разной структурой. Выделяют следующие типы баз данных NoSQL [4]:

1. Ключ-значение – вариант хранилища данных, использующий ключ для доступа к значению в рамках большой хэш-таблицы. Системы управления такими базами данных применяются для хранения изображений, создания специализированных файловых систем, в качестве кэшей для объектов, в Big Data системах. Примерами использования баз данных данного типа являются игровые, рекламные приложения и приложения IoT.

2. Документно-ориентированные – единицей хранения является документ, обладающий определенной структурой и представленный произвольным набором полей, который может быть представлен в JSON, XML или BSON формате. Документ может содержать вложенные документы и массивы. В базе данных поддерживаются индексы и поиск по полям документа. Такие СУБД используются для организации каталогов, пользовательские профилей и систем управления контентом, где каждый документ уникален и изменяется со временем.

3. Колоночное хранилище – хранит информацию в виде разреженной матрицы. Для доступа к данным используются три ключа: ключ строки (по нему отсортированы строки в базе), ключ столбца, временная метка. Наличие временных меток позволяет использовать СУБД для организации счетчиков, регистрации и обработки событий, связанных со временем, например, в системах биржевой аналитики, IoT/IIoT-приложениях и др.

4. Графовые хранилища представляют сетевую структуру, которая использует узлы и ребра для отображения и хранения данных. Как с узлами, так и с ребрами, отражающими связи, можно ассоциировать свойства (пары ключ-значение), в которых хранятся данные. Графовые СУБД поддерживают ACID-требования и специализированные языки запросов (например, Gremlin, Cypher и т.д.). Графовые СУБД используются в задачах, ориентированных на связи: социальные сети, маршруты общественного транспорта, дорожные карты, сетевые топологии.

NoSQL СУБД обладает следующими преимуществами.

1. Хранение в базах данных больших объемов неструктурированных данных. В СУБД NoSQL нет ограничений на использование только скалярных типов для хранимых данных, допускается использование таких пользовательских типов данных, как структуры и массивы.
2. NoSQL СУБД базы поддерживают горизонтальное масштабирование.

Как известно существует два вида масштабирования:

Вертикальное масштабирование – увеличение производительности приложения при добавлении ресурсов в рамках имеющегося сервера (например, увеличение ОП и т.д.). В этом случае масштабируемость ограничена. В рамках реляционных СУБД в основном поддерживается именно вертикальное масштабирование.

Горизонтальное масштабирование – увеличение производительности приложения за счет распределения нагрузки между имеющимся сервером

и добавляемыми в вычислительную систему новыми, необязательно супермощными серверами. Горизонтальное масштабирование серверов является особенно актуальным для распределенных систем обработки данных.

3. Возможность обеспечения высокой доступности за счет репликации данных и шардинга. Шардинг – это разделение информации по разным серверам. Каждый сервер отвечает только за определенный набор данных и обрабатывает запросы, относящиеся только к этому набору данных. Это позволяет увеличить скорость обработки данных.

Указанные преимущества NoSQL по отношению к реляционным SQL базам данным привели к тому, что в настоящее время NoSQL базы данных составляют значительную часть сложившейся системы инструментов для хранения данных, повсеместно применяемых как в небольших, так и крупных веб-проектах.

Вопросы безопасности NoSQL баз данных являются не менее актуальными чем безопасность реляционных баз данных. Особенности организации данных в NoSQL базах, работа с базами данных через веб-приложение, использование облачных вычислений и хранилищ требуют использования дополнительных механизмов по обеспечению безопасности баз данных [5,6].

Рассмотрим средства обеспечения безопасности NoSQL базы на примере MongoDB документно-ориентированной СУБД, в настоящее время широко применяемой в системах для организации каталогов, пользовательские профилей и систем управления контентом [1].

Подсистема безопасности MongoDB включает механизмы, обеспечивающие аутентификацию, авторизацию, шифрование, аудит, защиту сети и конфигурации сервера.

MongoDB поддерживает следующие механизмы аутентификации:

– *SCRAM аутентификация с ответом на вызов* (механизм аутентификации, используемый для MongoDB по умолчанию). Он основан на стандарте IETF RFC 5802, который определяет методы реализации механизмов запрос-ответ для аутентификации пользователей с помощью паролей.
– *аутентификацию сертификата x.509* для аутентификации клиента и внутренней аутентификации членов наборов реплик и сегментированных кластеров. Клиенты используют сертификаты x.509 для аутентификации на серверах вместо имен пользователей и паролей. Каждый уникальный пользователь MongoDB должен иметь уникальный сертификат. Центр сертификации должен выдавать сертификаты как для клиента, так и для сервера. Для аутентификации сертификата x.509 требуется безопасное соединение TLS / SSL.

- *аутентификация прокси-сервера* через службу Lightweight Directory Access Protocol (LDAP). MongoDB поддерживает запросы к серверу LDAP о группах LDAP, членом которых является аутентифицированный пользователь. MongoDB сопоставляет отличительные имена (DN) каждой возвращаемой группы с ролями admin в базе данных и авторизует пользователя на основе сопоставленных ролей и связанных с ними привилегий.
- *аутентификация Kerberos*. Kerberos — это стандартный протокол аутентификации для больших клиент/сервер систем. Для каждого пользователя, который будет аутентифицироваться с помощью Kerberos необходимо создать пользователя в MongoDB.

*Авторизация*. MongoDB использует управление доступом только на основе ролей (RBAC).

Пользователю предоставляется одна или несколько ролей, членством в которых он получает доступ к ресурсам и операциям базы данных. Другого способа получения привилегий для пользователя в MongoDB не существует.

Роль может включать одну или несколько существующих ролей в свое определение, и в этом случае она наследует все привилегии включенных ролей. Кроме встроенных ролей допускается создание и пользовательских ролей

Привилегия состоит из указанного ресурса и действий, разрешенных для этого ресурса. Ресурс представляет собой базу данных, набор коллекций, или кластер. Действие определяет операцию, разрешенную на ресурсе.

Механизмы шифрование в MongoDB применяются для:

- шифрования данных в хранилище;
- шифрования сетевого трафика с использованием TLS/SSL (безопасность транспортного уровня / уровень защищенных сокетов);
- шифрование на уровне приложения.

Процесс шифрования данных включает:

- генерацию мастер-ключа;
- генерацию ключей для каждой базы данных;
- шифрование ключей базы данных мастер-ключом;
- шифрование данных ключами базы данных.

Шифрование данных происходит прозрачно на уровне хранения (файлов), в незашифрованном виде данные существуют только в памяти и во время передачи.

Шифрование на уровне приложения обеспечивает шифрование для отдельных полей или документов на уровне приложения. Для шифрования полей и документов на уровне приложения на стороне клиента должны быть доступны ключи для шифрования и дешифрования данных.

В MongoDB реализованы средства аудита, позволяющие администраторам отслеживать события аудита, связанные с работой в системе нескольких пользователей и приложений. События аудита могут записываться в консоль, системный журнал, файл JSON или файл BSON. Поддерживается также и механизм фильтрации событий. Для аудита может быть настроен фильтр с целью ограничения регистрируемых событий.

Одним из основных свойств SQL базы данных является необходимость содержания правильных, непротиворечивых данных [2]. В реляционных СУБД база данных хранит данные, определенные в соответствии с некоторой моделью и правилами (ограничениями целостности) гарантирующими их целостность. Приведем сравнение терминологии, принятой для определения структур в MS SQL SERVER и MongoDB:

| | | |
|---|---|---|
| Таблица | — | Коллекция |
| Строка | — | Документ |
| Столбец | — | Поле |
| Первичный ключ | — | _Id |

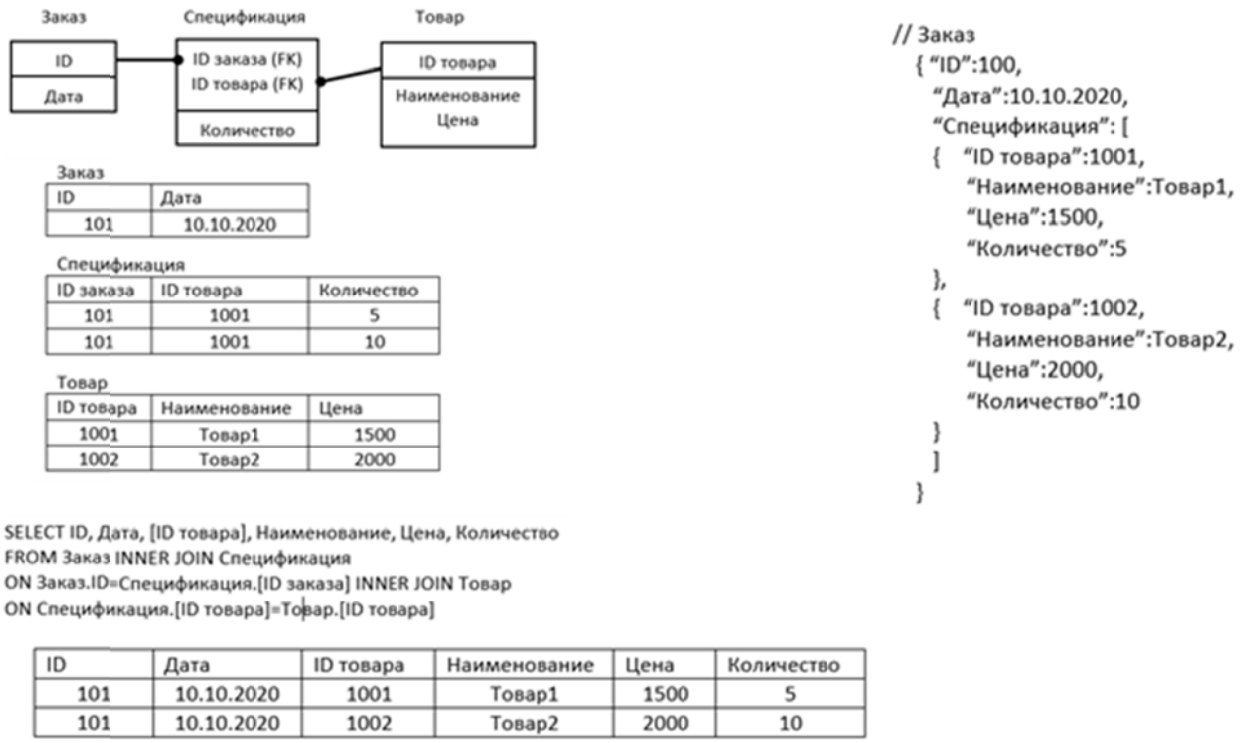
В базе данных SQL требуется сначала определить схему таблицы, а затем уже добавлять данные, и каждая вновь добавляемая строка должна содержать данные, соответствующие схеме таблицы.

В коллекции MongoDB по умолчанию не требуется, чтобы ее документы имели одну структуру. То есть документы в одной коллекции не обязательно должны иметь одинаковый набор полей, а тип данных для поля может различаться в разных документах из одной коллекции. Структура документа может быть представлена агрегатом данных, что позволяет сохранить данные в том виде, в котором они существуют в исходном документе. Элементом структуры документа может быть массив значений или структура другого документа. Поля встроенных документов могут быть связаны с полями основного документа отношением один-к-одному или один-ко-многим. Такое агрегирование данных в документе приводит к денормализации данных, однако позволяет значительно быстрее осуществлять доступ к ним.

На рисунке 2.3 приведен пример нормализованной схемы SQL базы данных и соответствующий ему документ в NoSQL БД.

Для получения данного документа в базе данных SQL требуется выполнение запроса на соединение трех таблиц.

Приведенный в примере документ соответствует встроенной модели данных. Встроенная модель данных обеспечивает возможность запрашивать и извлекать связанные данные за одну операцию с базой данных. Основной принцип проектирования структуры данных в соответствии с этой моделью – она должна быть максимально оптимизированной под наиболее частые запросы из приложения.




Заказ

| ID | Дата |
|---|---|
| 101 | 10.10.2020 |

Спецификация

| ID заказа | ID товара | Количество |
|---|---|---|
| 101 | 1001 | 5 |
| 101 | 1001 | 10 |

Товар

| ID товара | Наименование | Цена |
|---|---|---|
| 1001 | Товар1 | 1500 |
| 1002 | Товар2 | 2000 |

```
SELECT ID, Дата, [ID товара], Наименование, Цена, Количество
FROM Заказ INNER JOIN Спецификация
ON Заказ.ID=Спецификация.[ID заказа] INNER JOIN Товар
ON Спецификация.[ID товара]=Товар.[ID товара]
```

| ID | Дата | ID товара | Наименование | Цена | Количество |
|---|---|---|---|---|---|
| 101 | 10.10.2020 | 1001 | Товар1 | 1500 | 5 |
| 101 | 10.10.2020 | 1002 | Товар2 | 2000 | 10 |

Рисунок 2.3 – Документ NoSQL базы данных,
соответствующий результату запроса
к нормализованным реляционным таблицам

В MongoDB нет механизма связывания документов по внешнему ключу, так как это осуществляется в SQL базе данных, однако разрешается устанавливать связи между документами с помощью ссылок. Существует два метода связывание документов с использованием ссылок.

– Добавление в документ поля, значением которого является id другого документа. Для доступа к связанным данным в этом случае потребуется два запроса. Первый к документу, а второй к данным связанного документа.
– Использование DBRefs – ссылки из одного документа в другой по id с указанием коллекции, которой принадлежит документ и базы данных, если коллекция принадлежит другой базе данных.

Рассмотренные методы позволяют установить связи между документами и получить доступ к данным связанных документов. Однако механизм реализации таких связей не представляется достаточно простым в практической реализации.

MongoDB позволяет определить связи между документами для доступа к данным, но при этом нет механизма поддержки целостности связанных данных.

В SQL базах данных при определении схемы базы данных можно с помощью декларативных ограничений целостности установить правила

проверки данных на уровне столбцы, таблицы или связанных данных двух таблиц. Контроль за соблюдением установленных правил осуществляет сервер.

При работе с NoSQL базами данных клиентское приложение в случае необходимости должно осуществлять контроль данных и реализовывать процедуры поддержки ссылочной целостности между документами.

Как в любой информационной системе, функционирующей в архитектуре клиент-сервер в системах, использующих NoSQL базы данных, можно выделить три уровня, подверженные атакам злоумышленников:

- СУБД (например, рассмотренная MongoDB);
- API программный интерфейс для работы с NoSQL базой данных;
- клиентское Web-приложение.

Как уже было сказано выше клиентскому приложению передаются функции контроля входных данных. В случае нахождения в приложении неконтролируемых данных у злоумышленников появляется возможность передачи инъекции в JSON строке.

MongoDB является СУБД с открытым исходным кодом, поэтому всегда существует возможность получения и анализа исполняемого кода сервера и при успешном поиске уязвимости злоумышленник получает возможность атаки на все приложения, работающие с MongoDB.

MongoDB одна из наиболее популярных документно-ориентированных СУБД. В настоящее время большинство NoSQL баз данных размещается в открытых облачных хранилищах. Как показали исследования, выполненные в апреле 2019 года компанией DeviceLock, из 1900 серверов, которые используют платформы MongoDB, Elasticsearch и Yandex ClickHouse, более половины из них (52%) предоставляли возможность неавторизованного доступа [3]. 10% из числа открытых баз данных содержали персональные данные россиян или коммерческую информацию компаний, а 4% из них уже были взломаны, и хакеры требовали выкуп за похищенные базы данных.

Можно сделать вывод о недостаточно безопасной схеме аутентификации на сервере, а также о неправильной конфигурации сервера и баз данных. Одной из причин наличия этих уязвимостей может являться некомпетентность администраторов, которые эти облачные базы подключают и настраивают. Допущенные ими ошибки приводят к проблемам с безопасностью.

Список использованных источников:

1. Документация по СУБД MongoDB [электронный ресурс] URL: https://docs.mongodb.com/ (дата обращения: 15.10.2020).
2. Материалы сайта AWS. Что такое SQL [электронный ресурс] URL: https://aws.amazon.com/ru/nosql/ (дата обращения: 15.10.2020).

3. Материалы сайта Мир информационных технологий. Анна Савельева Почти миллион клиентов банков под угрозой [электронный ресурс] URL: https://www.it-world.ru/it-news/security/146137.html (дата обращения: 15.10.2020).
4. Материалы сайта Школа больших данных. Анна Вичугова. NoSQL [электронный ресурс] URL: https://www.bigdataschool.ru/wiki/nosql (дата обращения: 15.10.2020).
5. Празян К.А., Лачихина А.Б. JSON-уязвимости формата NoSQL [электронный ресурс] URL: https://vre.instel. /jour/article/viewFile/784/774 (дата обращения: 15.10.2020).
6. Фримучков А.Н. NoSQL-инъекции на примере нереляционной СУБД MongoDB [электронный ресурс] URL: https://cyberleninka.ru/article/n/nosql-inektsii-na-primere-nerelyatsionnoy-subd-mongodb/viewer (дата обращения: 15.10.2020).

## 2.3. Уязвимости npm-пакетов при разработке приложений на платформе Node.js

Node.js является кроссплатформенной средой выполнения JavaScript-кода. Первые версии проекта были выпущены в 2010 году, и с тех пор Node.js обрел крайне высокую популярность в области веб-программирования. За все время развития в разработке платформы приняло участие более двух с половиной тысяч человек. Тысячи компаний включая Microsoft, NASA и Uber используют Node.js в своих продуктах [4].

Популярность проекта обусловлена несколькими факторами:

- использование javascript в качестве основного языка разработки как клиентской, так и серверной части приложения;
- неблокирующая архитектура, обеспечивающая высокую скорость работы, сравнимую с показателями компилируемых языков [1];
- простота использования;
- более миллиона библиотек, которые доступны для использования любому Node.js-разработчику [5].

Большое количество библиотек и простота работы с ними внесли огромную роль в популярность платформы, образовав мощную экосистему.

Работа с библиотеками организована через менеджер пакетов npm (аббр. node package manager), который предоставляет функционал создания, публикации, загрузки и установки пакетов. Все опубликованные пакеты хранятся в открытом репозитории npm, и представляют собой архивы JavaScript и JSON файлов. Содержимое пакета описывается файлом манифеста, включающим в себя информацию о названии, версии и зависимостях пакета. Помимо этого, файл манифеста декларирует ссылки на исходный код.

Важной особенностью библиотек npm является широкое применение транзитивных зависимостей. Так, например, если пакет A использует пакет B, то при установке пакета A в пакет C возникает зависимость между паке-

тами C и B. По статистике, устанавливая один npm-пакет, разработчик вынужден довериться в среднем семидесяти девяти сторонним пакетам и тридцати девяти лицам, их сопровождающим. Наиболее популярные библиотеки могут прямо или косвенно влиять на широкий ряд других пакетов, часто более чем на сотни тысяч, поэтому достаточно очень малого количества скомпрометированных аккаунтов разработчиков таких библиотек, чтобы внедрить вредоносный код в большое количество пакетов [6].

Эксплуатируя уязвимости npm-пакетов, злоумышленник способен получить широкий ряд возможностей – от хищения данных кредитных кард, паролей, конфиденциальной информации, до эксплуатации вычислительных ресурсов жертвы для организации майнинговых сетей.

Вышесказанное делает уязвимости пакетов npm благоприятным полем для атак.

**Классификация уязвимостей**

Описанные особенности архитектуры npm открывают ряд различных уязвимостей. Для части из них уже имеются прецеденты эксплуатации, и на основе успешных атак созданы рекомендации по безопасности. Другие не имеют строгих принципов защиты и во много базируются на человеческом факторе. Отдельные категории уязвимостей не могут быть полностью решены с помощью технических средств, поэтому обеспечение собственной безопасности при использовании пакетов npm лежит на пользователе.

Для удобства оценки уязвимостей и защиты от них можно ввести классификацию по положению, занимаемому в архитектуре экосистемы npm. При такой классификации хорошо прослеживается роль каждого участника экосистемы и степень серьезности уязвимости. Можно выделить следующие уязвимости:

- уязвимости скриптов пост-установки;
- уязвимости на этапе сборки архива для публикации в репозитории;
- уязвимости в бизнес-логике библиотек;
- уязвимости контроля доступа к исходным кодам библиотек.

*Уязвимости скриптов пост-установки*

По умолчанию любой NPM-пакет имеет возможность выполнить произвольный код в момент, когда завершается его установка. Обычно такой код используется для первоначальной настройки пакета и декларируется в файле package.json. Пример конфигурации пакета, использующего пост-установочный скрипт, выглядит следующим образом:

```
{
  "name": "malicious-package",
  "scripts": {
    "postinstall": "malicious-script.sh"
  }
}
```

Исполняемый файл, указанный в секции «postinstall» будет выполнен от имени текущего пользователя системы. Таким образом злоумышленник может получить доступ к объектам файловой системы, а при достаточном уровне доступа и к другим данным [2].

*Уязвимости на этапе сборки архива для публикации в репозитории*

Одним из преимуществ платформы Node.js является использование библиотек с открытым исходным кодом. NPM-пакеты используют git в качестве системы контроля версий и декларируют ссылки на хостинги git-репозиториев в файле package.json. Таким образом пользователю библиотеки необязательно скачивать пакет, чтобы просмотреть исходные коды. Однако не существует прямой зависимости между кодом, находящимся в архиве библиотеки и кодом, опубликованным в публичном git-репозитории. Разработчик библиотеки может сконфигурировать пакет таким образом, чтобы часть файлов исходного кода не попала в репозиторий, но при этом осталась бы частью пакета. Для достижения этого результата достаточно добавить список исключаемых из публичного репозитория файлов в список исключений системы контроля версий, который содержится в файле .gitignore.

При выполнении команды публикации пакета NPM создает архив всех файлов, находящихся в рабочей директории, и отправляет его на сервера NPM. После этого разработчик имеет возможность опубликовать новый исходный код в публичном репозитории. При этом исключенные файлы не попадут в репозиторий, но будут присутствовать в пакете. Таким образом исходный код, доступный в git-репозитории, не будет содержать признаков угрозы в то время, как реальный код пакета может содержать что угодно.

Единственным следом злонамеренного кода будет ссылка на него в файле .gitignore, однако часто этот файл содержит большое количество исключений, среди которых могут быть артефакты среды разработки, файловой системы, утилит и прочих объектов, необходимых для разработки. Поэтому анализ исключений системы контроля версий не всегда может быть эффективен.

Анализ скачанного пакета так же может быть мало эффективен в связи с тем, что вредоносный код может быть обфусцирован и затем минифицирован таким образом, что прочитать и понять его назначение крайне сложно.

*Уязвимости в бизнес-логике библиотек*

По своей сути уязвимости в бизнес-логике библиотек ничем не отличаются от уязвимостей в бизнес-логике приложений, которые пишутся пользователями библиотек. Единственное отличие состоит в том, что уязвимость библиотеки не всегда может эксплуатироваться сама по себе, для ее эксплуатации требуется приложение, использующее эту библиотеку.

Таким образом уязвимость библиотеки становится уязвимостью пользовательского приложения.

Примером такой уязвимости может служить ранний подход библиотеки Redux к инициализации первоначального состояния клиентской части веб-приложения при отрисовке на стороне сервера.

Код HTML-страницы содержал инструкцию инициализации состояния, преобразующую JavaScript-объект в строку:

```
<script>
const state = ${JSON.stringify({ user })};
window.__PRELOADED_STATE__ = state;
<script>
```

Если бы злоумышленнику удалось зарегистрироваться с данными, содержащими строку `</script>`, то он получил бы возможность выполнить произвольный JavaScript-код:

```
const user = { name: "John</script><script>//
any malicious code</script>"  };
```

*Уязвимости контроля доступа к исходным кодам библиотек*

Уязвимости данного типа во многом основаны на человеческом факторе. Как уже было сказано, благодаря длинным цепочкам транзитивных зависимостей, один популярный NPM пакет может влиять на сотни тысяч связанных с ним. Получение доступа к аккаунту разработчика библиотеки может, согласно исследованиям, привести к контролю 54% всех пакетов [3].

Имея доступ к публикации изменений в пакете, злоумышленник может модифицировать код таким образом, чтобы получить слабозащищенные пароли и ключи аккаунтов лиц, использующих вредоносный пакет. Рост количества зараженных пакетов в таком случае будет близок к экспоненциальному [6].

**Методы защиты от эксплуатации уязвимостей**

Для защиты от эксплуатации описанных категорий уязвимостей разработчик должен обладать широким спектром различных данных. В настоящее время для каждого пакета NPM отображает количество загрузок, зависимостей и пакетов, зависимых от данного. Однако прямого способа получить информацию о транзитивных зависимостях и лицах, поддерживающих пакет, не существует. Однако NPM постоянно отслеживает все публикуемые пакеты и в автоматическом режиме ведет учет найденных уязвимостей.

Весь публикуемый код проходит этап статического анализа на предмет возможных уязвимостей. Пакеты, содержащие намеренно внедренные уязвимости, удаляются. После установки любого пакета в свой проект пользователь получает отчет о состоянии используемых им зависимостей, сформированный инструментами NPM. Данный отчет содержит характе-

ристики найденных уязвимостей, степень серьезности и дату первого обнаружения.

Полагаясь на средства автоматизированных средств защиты, предоставляемые NPM, пользователи должны уметь сами оценивать устанавливаемые пакеты и анализировать свои проекты на предмет существующих уязвимостей для обеспечения приемлемого уровня безопасности.

Для защиты от эксплуатации уязвимостей скриптов пост-установки пользователю следует устанавливать пакеты от имени учетной записи с минимальными возможными полномочиями. Анализ исходных кодов скриптов также может быть полезен при поиске вредоносного кода. Существует также возможность полного запрета выполнения скриптов пост-установки.

Чтобы защититься от вредоносного кода, внедренного на этапе публикации пакета, необходимо проявлять бдительность при анализе git-репозитория, исключений системы контроля версий и исходных кодов пакета.

Для защиты от атак, базирующихся на уязвимостях в исходных кодах библиотек, не существует четких рекомендаций. Пользователь должен либо принимать риски, либо не использовать данные библиотеки. В общем случае необходимо пользоваться самой актуальной версией пакета, содержащей патчи для всех выявленных уязвимостей. Таким образом риск будет минимален.

Защита от атак, эксплуатирующих уязвимости контроля доступа к пакетам, должна основываться в первую очередь на базовых правилах безопасности паролей и учетных записей. Следует избегать хранения учетных данных в файле .npmrc, а при необходимости использования таких данных в автоматическом режиме, например, в конвейере непрерывной интеграции, использовать специальные защищенные хранилища.

Рассмотренная классификация уязвимостей npm-пакетов показывает, насколько многогранными могут быть модели атак на экосистему. Большое количество пакетов и тесное их влияние друг на друга, простота использования библиотек и часто низкая квалификация их пользователей открывают широкие возможности для злоумышленников. Несмотря на то, что разработчики npm постоянно внедряют новые механизмы защиты и разрабатывают советы по безопасности, пользователи npm должны осознавать риски и знать, какие уязвимости могут быть привнесены в их приложения с каждой новой зависимостью. Рассмотренные методы защиты от эксплуатации уязвимостей могут повысить уровень безопасности приложений, разрабатываемых на платформе Node.js.

Список использованных источников:

1. Casciaro, M. Node.Js Design Patterns / M. Casciaro – Packt Publishing, 2016. – 526 p.
2. NPM Documentation: сайт. – URL: https://docs.npmjs.com/misc/scripts (дата обращения: 22.10.2020).

3. Skovoroda, N.A. Gathering weak npm credentials [Электронный ресурс] // Github. – URL: https://github.com/ChALkeR/notes/blob/master/Gathering-weak-npm-credentials.md (дата обращения: 22.10.2020).
4. SPEC INDIA. Why Node Js is Very Popular Among Fortune 500 Companies? / SPEC INDIA [Электронный ресурс] // Medium. – URL: https://medium.com/quick-code/node-js-and-fortune-500-companies-fewer-efforts-more-rewards-282db19160c0 (дата обращения: 21.10.2020);
5. Tal L. NPM passes the 1 millionth package milestone! What can we learn? [Электронный ресурс] // Snyk. – URL: https://snyk.io/blog/npm-passes-the-1-millionth-package-milestone-what-can-we-learn/ (дата обращения: 21.10.2020);
6. Zimmermann, M. Small World with High Risks: A Study of Security Threats in the npm Ecosystem / M. Zimmermann, Cristian-Alexandru Staicu, C. Tenny, M. Pradel // 28th USENIX Security Symposium. – 2019 – P. 3 – 17.

## 2.4. Применение сетевых протоколов для криптографической защиты DNS

С начала 2019 года в новостных лентах стали все чаще упоминаться DoH (DNS поверх HTTPS) и DoT (DNS поверх TLS). Тогда же крупные IT компании, такие как Microsoft, Mozilla, Chrome, Cloudflare и AdGuard, заявили об интеграции данных протоколов в свои сервисы. Внедрение данных протоколов в Android, iOS, Windows 10 и web-браузеры должно снизить количество различных DNS атак, включая DNS фишинг и спуффинг, DNS туннелирование. На данный момент, согласно отчетам Global DNS Threat Report компании IDC [1,2], DNS атаки являются одними из самых популярных кибератак.

При использовании DNS без шифрования наиболее вероятными атаками являются трекинг и спуфинг.

Поскольку запросы DNS отправляются в открытом виде, возможен их перехват и отслеживание (трекинг) посещаемых страниц. Трекинг может осуществлять любой промежуточный узел в сети, через который идет транспортировка запросов, в том числе ненадежный DNS-сервер или маршрутизатор, который был назначен сетью в автоматическом режиме.

Спуфинг означает перехват и подмену ответа DNS-сервера. Как правило, целью такой подмены является перенаправление на поддельный сайт. С другой стороны, с помощью подмены может быть заблокирован доступ к реальному, действующему сайту. Cisco Talos, группа по анализу угроз Cisco, обнаружила несколько атак, основанных на перехвате DNS и манипуляциях с ним в качестве основного вектора заражения [3]. Начиная с 2018 года, вектор атак направлен на компрометацию самой DNS инфраструктуры (то есть компрометацию DNS серверов разного уровня), что вызывает серьезную озабоченность. По данным Palo Alto Networks Unit 42

Threat Research, примерно 85% вредоносных программ используют DNS для установления канала управления и контроля, обеспечивая проникновение вредоносного кода в корпоративные сети [4].

Легкость манипулирования объясняется отсутствием у DNS протокола каких-либо механизмов обеспечения безопасности. Таким образом, становится возможным успешное выполнение атаки «человек посередине» (MITM). Подобные атаки могут приводить к выводу из строя приложений и операционных систем, компрометации конфиденциальной информации. В то время, когда был разработан DNS, от него требовалась прозрачность и легкое масштабирование. Позднее IETF (Internet Engineering Task Force – Инженерный совет интернета, открытое международное сообщество проектировщиков сетей, операторов, провайдеров и исследователей, занимающихся развитием архитектуры интернета и вопросами обеспечения бесперебойного функционирования сети) признал проблему всеобщего мониторинга трафика. Главным решением данной проблемы стало шифрование.

Согласно статистике [5], представленной в виде графика (рис. 2.4), что процент веб-страниц, загруженных Firefox с использованием HTTPS, ежедневно растет и на сегодняшний день в мире составляет более 83%. Приведенная статистика основана на данных сервиса Lets's Encrypt, запущенного компанией Internet Security Research Group, которая фокусируется на вопросах интернет безопасности. Сервис Lets's Encrypt предоставляет автоматическую выдачу бесплатных сертификатов для TLS-шифрования. Одним из спонсоров данного сервиса является компания Google.

В настоящее время подключение по протоколу HTTPS стало дефакто стандартом в веб. Наличие или отсутствие поддержки сайтом HTTPS влияет на его место в поисковой выдаче. Более того, некоторые браузеры имеют режим HTTPS-only, который запрещает посещать сайт без HTTPS.

HTTPS – протокол прикладного уровня базирующийся на использовании защищенного транспортного протокола SSL/TLS. Современные реализации используют TLS, начиная с версии 1.2. В августе 2018 года принята новая версия – TLS 1.3 (RFC 8446), призванная усилить безопасность соединений. В этой версии прекращена обратная совместимость с SSL, запрещены устаревшие и небезопасные опции, введена обязательность цифровой подписи и использование аутентифицированного шифрования.

Известны многочисленные атаки на ранние версии SSL/TLS, однако лишь некоторые из них могут быть реализованы вплоть до TLS 1.2 включительно [6]. За счет поддержки устаревших криптонаборов в целях обратной совместимости для этих версий возможно понижение уровня защиты и реализация MITM атак. Весной 2020 года компании Microsoft,

Mozilla и Google планировали прекратить поддержку устаревших версий протоколов шифрования TLS 1.0 и 1.1 [7], однако реализация этого решения была отложена из-за пандемии.

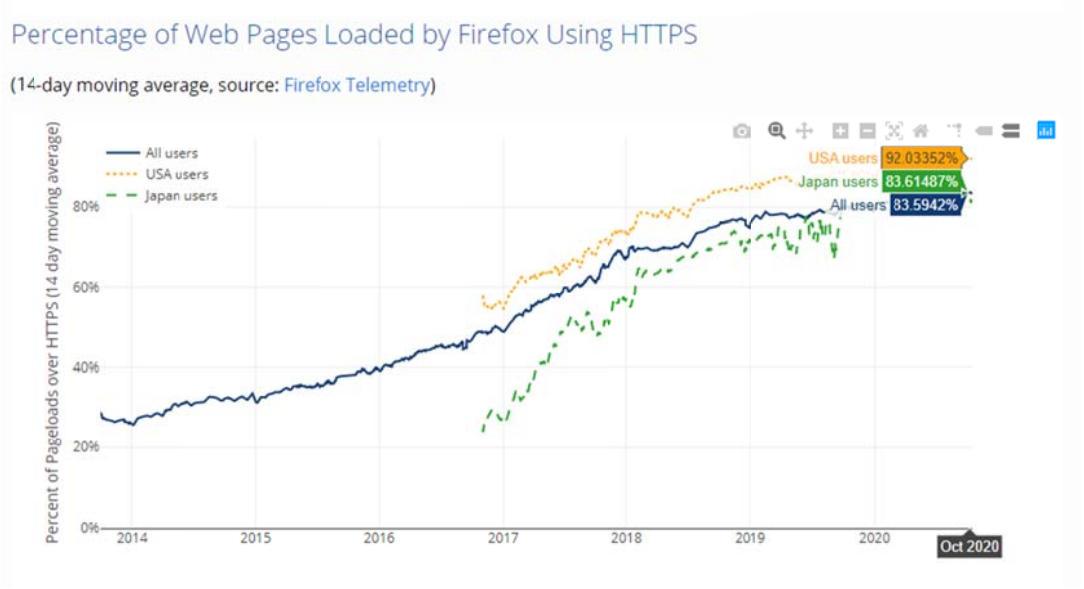



Рисунок 2.4 – Доля веб-страниц, загруженных Firefox с использованием HTTPS

Согласно текущей версии стандарта PCI DSS, должна использоваться версия TLS 1.3 для обеспечения защиты данных владельцев карт различных международных платежных систем. Кроме того, он является основой различных прикладных протоколов помимо HTTPS. Стоит отметить, позитивное отношение к TLS 1.3 российских регуляторов в сфере защиты информации. Так, Росстандартом утверждены рекомендации по стандартизации Р 1323565.1.030-2018 «Информационная технология. Криптографическая защита информации. Использование российских криптографических алгоритмов в протоколе безопасности транспортного уровня (TLS 1.3)», которые введены в действие 1 июня 2020 года.

В качестве решения проблем с безопасностью DNS, и прежде всего защиты от атаки «человек по середине» (MITM), было предложено использовать DNS подключение поверх HTTPS. Соответствующий протокол, получивший название DoH (DNS over HTTPS, RFC 8484) активно продвигается Mozilla.

Трудности по обеспечению конфиденциальности DNS заключаются в том, что при «рукопожатии» для получения IP-адреса сервера какого-либо сайта, будет происходить многократная обработка данного запроса различными серверами-операторами. При этом каждый из них будет знать полное содержание данного запроса.

DoH отправляет введенное доменное имя на DNS-сервер, совместимый с DoH, с помощью зашифрованного HTTPS-соединения вместо про-

стого текста. Это защищает информацию о том, какие веб-сайты запрошены, от просмотра сторонними лицами [8].

Другим уязвимым местом является SNI (Server Name Indication) – расширение компьютерного протокола TLS, которое позволяет клиентам сообщать имя хоста, с которым он желает соединиться во время процесса «рукопожатия». Для того, чтобы нельзя было не санкционированно узнать данную информацию, был разработан механизм ESNI (Encrypted SNI) для протокола TLS 1.3, который шифрует имя запрашиваемого сайта с помощью открытого ключа сайта, получаемого из системы имен DNS.

DoH требует поддержки не только на стороне клиента, на и на стороне сервера, а безопасность информации на сервере обеспечивается обращением только к доверенным рекурсивным резолверам (Trusted Recursive Resolver, TRR). Это не дает реализовать угрозы подмены резолвера, отслеживания по пути запроса маршрутизаторами и отслеживания DNS-запросов DNS-серверами.

Примером TRR является сервис рекурсивного резолвинга от компании Cloudflare. Эта компания обязалась удалять спустя сутки всю персонализированную информацию и не продавать такую информацию третьим лицам. Также регулярно проводятся внешние проверки выполнения данных правил. Cloudflare выбран в браузере Mozilla в качестве DNS резолвера по умолчанию при включении DoH. В настоящее время DoH используется не по умолчанию, его необходимо активировать вручную в настройках. Существуют и другие TRR, например, Google Public DNS, Adguard и др.

DoH использование представляется разумным выбором для защиты DNS-трафика между веб-приложением и внешним сервисом DNS-резолвера. Другой частью DNS является рекурсивный поиск, предусматривающий взаимодействие резолвера с серверами DNS, осуществляющими поддержку различных доменных зон. Для защиты рекурсивного поиска лучше подойдет вариант протокола, защищающего DNS непосредственно при помощи TLS без дополнительной внутренней обертки в виде HTTP.

Протокол шифрования DoT (DNS over TLS, RFC 7858) использует TLS «напрямую», передавая DNS трафик непосредственно внутри защищенных TLS-записей. Этот протокол можно использовать и в контексте DNS взаимодействия между веб-приложением и выбранным им рекурсивным резолвером.

Оба протокола (DoH, DoT) обеспечивают шифрование и контроль целостности данных, передаваемых между клиентом и сервером, а также аутентификацию сервера.

Следует отметить, что DNS имеет свой механизм криптографической защиты адресной информации – DNSSEC (расширения безопасности системы доменных имен, RFC 4033, RFC 4034, RFC 4035). В начале 2019

года организация ICANN, регулирующая вопросы, связанные с IP-адресами и доменными именами, призвала к использованию DNSSEC для всех доменных имен [9]. Этот протокол позволяет удостоверить подлинность ответов DNS сервиса при помощи электронной подписи. При этом DNSSEC не обеспечивает конфиденциальность данных, поскольку получаемая от DNS информация не шифруется, а также не подразумевает аутентификации узлов, участвующих в обмене данными. Поэтому DNSSEC и использование защищенных протоколов TLS/HTTPS для защиты DNS трафика являются взаимно дополняющими, но не противоречащими друг другу мерами. Но именно протоколы DoH и DoT рассматриваются как средство повышению уровня приватности интернет-пользователей.

Комбинированное сочетание DoH, eSNI и TLS 1.3 не только обеспечивает повышения уровня конфиденциальности и целостности, но и соответствует общемировым тенденциям информационной безопасности, что отражается, в частности, закреплением указанных протоколов в стандартах IETF.

С другой стороны, при использовании данных протоколов возникают определенные риски, связанные с относительной новизной их использования, а также невозможности эффективного функционирования некоторых механизмов информационной безопасности.

Например, некоторые механизмы используют блокировку DNS, например, при родительском контроле, при запрете доступа к сайтам с запрещенным контентом, а также для мошеннических сайтов. Происходит простая фильтрация, проверяется есть ли запрашиваемое доменное имя в списках известных вредоносных команд и командных центров ботнетов. Такая блокировка не будет работать в случае использования DoH.

Следует отметить, что технологии криптографической защиты активно используются вредоносным кодом для противодействия обнаружению. Так, в 2019 году обнаружено первое семейство вредоносных программ (Godlua), использующих этот протокол для маскировки своих сообщений с целью обхода средств межсетевого экранирования. Затем аналогичный прием использовал вредоносный код PsiXBot [4].

Также при использовании DoH придется решить проблему с управлением контентом, например, с родительским контролем или корпоративной DLP-системой. Маскировка имени запрашиваемого узла делает невозможным обход ограничений на посещение определенных интернет-ресурсов, поскольку расшифровать DNS запрос сможет только доверенный резолвер TRR.

Кроме того, возникает еще и проблема доверия самому TRR. Конечно, использование DoH обеспечивает защиту от атак типа «человек посередине», но, с другой стороны, при повсеместном использовании данного

протокола придется полагаться в том числе на уровень защищенности TRR, а их количество весьма ограничено, кроме того, они подконтрольны нескольким иностранным компаниям.

В сентябре 2020 года Министерством цифрового развития, связи и массовых коммуникаций Российской Федерации вынесен на общественное обсуждение законопроект, запрещающий использование на территории России eSNI, DoH, DoT и TLS 1.3 [10] путем внесения соответствующих поправок в федеральный закон № 149-ФЗ «Об информации, информационных технологиях и защите информации». Конкретные названия протоколов в тексте законопроекта не фигурируют, используется формулировка «протоколы шифрования, позволяющие скрыть имя (идентификатор) Интернет-страницы или сайта в сети «Интернет», однако перечень протоколов приведен в пояснительной записке к проекту поправок [11]. В министерстве полагают, что данные меры помогут в борьбе с распространением информации, запрещенной в РФ.

Действительно, использование данных алгоритмов шифрования приведет к невозможности или снижению эффективности блокировки сайтов с содержимым, запрещенным на территории РФ. В самом деле, отличить системный DNS-трафик DoH от «обычного» HTTPS, по которому пользователи просматривают веб-сайты, весьма проблематично. При использовании более старых версий протоколов, в том числе TLS 1.1 и 1.2, а также SNI, просмотр поля SNI на ранних этапах HTTPS-соединений позволяет определить, к какому домену пытается подключиться пользователь. В случае же TLS 1.3 поле SNI можно скрыть посредством ESNI. С другой стороны, технологии маскировки DNS трафика никак не влияют на прохождение самого IP трафика, а значит, не препятствуют «блокировке по IP» [12].

Законопроект предлагает блокировать интернет-ресурсы за нарушение данного запрета уполномоченным федеральным органом исполнительной власти.

Следует отметить, что по аналогичному пути блокировки ранее пошел КНДР. Специалисты из iYouPort, университета Мэриленда и Great Firewall Report, представили совместные отчеты, которые гласят, что в конце июля 2020 года китайские власти обновили «Великий китайский файрвол» таким образом, чтобы блокировать зашифрованные HTTPS-соединения, которые используют современные протоколы и технологии, защищающие от перехвата. В частности, производится блокировка всего HTTPS-трафика, использующего TLS 1.3 и ESNI, а IP-адреса, участвующие в таких соединениях, подвергаются временной блокировке длительностью от двух до трех минут [13].

На настоящий момент в сети интернет существуют различные высказывания экспертов по поводу запрета маскирующих протоколов, а ука-

занный законопроект вызвал широкий резонанс в российском IT-сообществе. Что касается запрета TLS 1.3 как такового, то это явно противоречит современным требованиям сетевой безопасности, однако в случае DoH все не так однозначно.

По словам партнера и директора компании «Интеллектуальный резерв» Павла Мясоедова, DoH- и DoT-протоколы в перспективе станут самыми популярными, их запрет грозит сложностями для российских компаний [14].

Партнер и руководитель практики управления киберрисками Deloitte Денис Липов заявляет, что протоколы разрабатываются в первую очередь для повышения защищенности интернет-сервисов, так что их ограничение, действительно, будет на руку злоумышленникам [14].

В будущем популярные иностранные браузеры и сайты будут использовать данные протоколы шифрования, как следствие, можно столкнуться с проблемой невозможности использования легитимных сайтов, не содержащих информацию, запрещенную на территории РФ.

В [12] отмечается, что проблема невозможности контроля DNS запросов существует только при использовании в качестве резолверов некоторых внешних DNS-серверов (например, Cloudflare, Google Public DNS и др.). Такая проблема может быть решена внедрением независимых DoH и DoT резолверов внутри Рунета и использованием резолверов, контролируемых интернет-провайдером. Хотя такой подход безусловно более трудоемок и затратен, чем простой запрет отдельных защищенных протоколов, он более конструктивен, поскольку позволит сочетать современные требования безопасности с необходимым уровнем контроля.

Список использованных источников:

1. Understanding the Critical Role of DNS in Network Security Strategy. IDC 2019 Global DNS Threat Report [электронный ресурс]. URL: https://www.efficientip.com/resources/idc-dns-threat-report-2019/ (дата обращения: 19.10.2020).
2. The Critical Role of DNS in Network Security StrategyIDC 2020 Global DNS Threat Report [электронный ресурс]. URL: https://www.efficientip.com/resources/idc-dns-threat-report-2020/ (дата обращения: 19.10.2020).
3. Ben Nahorney. DNS under attack – 25.07.2019 [электронный ресурс]. URL: https://blogs.cisco.com/security/dns-under-attack (дата обращения: 19.10.2020).
4. Минимизация рисков использования DNS-over-TLS (DoT) и DNS-over-HTTPS (DoH) – 11.10.2020 [электронный ресурс]. URL: https://www.securitylab.ru/blog/personal/Morning/349609.php (дата обращения: 19.10.2020).
5. Let's Encrypt Stats [электронный ресурс]. URL: https://letsencrypt.org/stats/#percent-pageloads (дата обращения: 19.10.2020).
6. Венедюхин А. Ключи, шифры, сообщения: как работает TLS – 01.09.2020 [электронный ресурс]. URL: https://tls.dxdt.ru/tls.html (дата обращения: 19.10.2020).
7. Firefox внедряет режим «только HTTPS» – 25.03.2020 [электронный ресурс]. URL: https://habr.com/ru/company/globalsign/blog/494024/ (дата обращения: 19.10.2020).

8. DNS через HTTPS в Firefox [электронный ресурс]. URL: https://support.mozilla.org/ru/kb/dns-cherez-https-v-firefox (дата обращения: 19.10.2020).
9. ICANN Calls for Full DNSSEC Deployment, Promotes Community Collaboration to Protect the Internet – 22 February 2019 [электронный ресурс]. URL: https://www.icann.org/news/announcement-2019-02-22-en (дата обращения: 19.10.2020).
10. Минцифры предложило запретить в РФ маскирующие протоколы шифрования – 21.09.2020 [электронный ресурс]. URL: https://www.interfax.ru/russia/727929 (дата обращения: 19.10.2020).
11. Пояснительная записка к проекту Федерального закона «О внесении изменений в статьи 2 и 10 Федерального закона «Об информации, информационных технологиях и о защите информации» [электронный ресурс]. URL: https://regulation.gov.ru/Files/GetFile?fileid=40090bac-b8da-42d4-8f8f-35b12957a9cb (дата обращения: 19.10.2020).
12. Венедюхин А. О последствиях криптозащиты DNS-запросов – 22.11.2019 [электронный ресурс]. URL: https://d-russia.ru/o-posledstviyah-kriptozashhity-dns-zaprosov.html (дата обращения: 19.10.2020).
13. Нефедова М. Китай блокирует весь HTTPS-трафик, использующий TLS 1.3 и ESNI – 10.08.2020 [электронный ресурс]. URL: https://xakep.ru/2020/08/10/great-firewall/ (дата обращения: 19.10.2020).
14. DNS поверх HTTPS (DNS-over-HTTPS, DoH) – 06.10.2020 [электронный ресурс]. URL: https://www.tadviser.ru/index.php/%D0%A1%D1%82%D0%B0%D1%82%D1%8C%D1%8F:DNS_%D0%BF%D0%BE%D0%B2%D0%B5%D1%80%D1%85_HTTPS_(DNS-over-HTTPS_DoH) (дата обращения: 19.10.2020).

## 2.5. Безопасность протоколов цифровых валют

Проблема создания электронного аналога наличных денег впервые была рассмотрена Дэвидом Чаумом, который в 1982 году предложил механизм слепой подписи (blind signature) как основу системы электронной наличности. При этом основным требованием к такой системе является неотслеживаемость платежей, то есть обеспечение анонимности покупателей – невозможности связать пользователя с его покупками. С другой стороны, система должна исключать возможность мошенничества, в частности, повторную трату электронной купюры (чека, монеты). Это значит, что если владелец уже предъявил электронную купюру к оплате, он не сможет использовать ее для оплаты повторно. Свойства идеальной системы электронных денег сформулированы в [1].

Задача обеспечения анонимности электронных купюр решается с помощью технологии слепой подписи, которая использует маскирующие (затемняющие) множители, что позволяет удостоверить (подписать) информацию, не зная ее содержания. Система электронных платежей Д. Чаума является централизованной, то есть требует участия банка во всех транзакциях. Эмиссия электронных купюр, равно как и решение пробле-

мы повторной оплаты также производятся централизовано банком, играющим роль доверенной стороны. Кроме того, эта система гарантирует анонимность покупателя, но не продавца.

В 1990-х Дэвидом Чаумом была предпринята попытка создания частной системы электронных платежей, которая получила название DigiCash, все транзакции в его проекте проверялись централизованно. Проект потерпел неудачу, а компания в 1998 году объявила о банкротстве. Вместе с тем широкое распространение получили цифровые валюты (Биткойн и другие, например, Ethereum), основанные на децентрализованном подтверждении транзакций и технологии блокчейна.

*Биткойн*

Биткойн (BTC, Bit Coin, цифровая монета) – протокол цифровой валюты (криптовалюты), которая, в отличие от фиатных денег, не эмитируется и не контролируется на государственном уровне. По сути, сеть Биткоин представляет одноранговую пиринговую (P2P, peer-to-peer) платежную систему, в которой каждый узел равноправен и самодостаточен, отсутствуют какие-либо управляющие или процессинговые центры. Проблема двойных трат (повторного использования в платежной системе одних и тех же цифровых монет) решена в сети Биткойн за счет концепции децентрализованного консенсуса и блокчейн технологии.

*Анонимность Биткойна*

Наряду с невозможностью контроля со стороны государственных органов или каких-либо других регуляторов, Биткойн ценится также и за обеспечение анонимности транзакций.

Транзакции Биткойна используют обычные (не слепые) электронные подписи ECDSA с фиксированными параметрами эллиптической кривой (secp256k1), при этом вся информация о транзакциях между адресами системы доступна в открытом виде, обеспечивая их полную прозрачность. Анонимность пользователей обеспечивается невозможностью связать Биткойн-адрес, используемый в транзакциях, с конкретным физическим или юридическим лицом за счет полного отсутствия в системе персональных данных владельцев Биткойн-адресов.

Право владения биткойнами устанавливается через криптографические ключи, Биткойн-адреса и электронные подписи. Владелец закрытого ключа контролирует средства, связанные с определенным Биткойн-адресом. При этом потеря личного ключа (ЛК) фактически означает безвозвратную утрату этих средств, поскольку формирование новой ключевой пары для существующего адреса невозможно.

Биткойн-адрес получается на основе открытого ключа (ОК) владельца кошелька с помощью однонаправленных хэш-функций (рис. 2.5) [2]. Получателями средств могут быть сценарии, их Биткойн-адреса определяются по иной схеме.

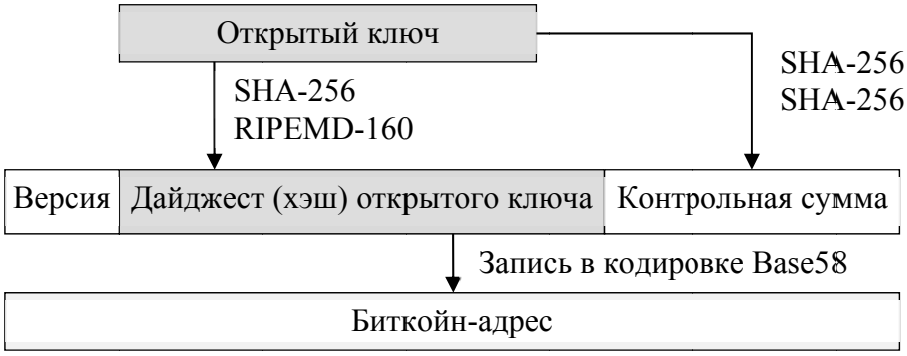



Рисунок 2.5 – Алгоритм формирования Биткойн-адреса

Свойство однонаправленности хэш-функций в описанном алгоритме наряду со сложностью задачи дискретного логарифмирования в группе эллиптической кривой обеспечивают практическую невозможность обратимости цепочки закрытый ключ → открытый ключ → Биткойн-адрес.

Для подтверждения владения биткойнами должно быть продемонстрировано знание соответствующего личного ключа, что достигается посредством подписи транзакции. При этом открытый ключ не связан с владельцем кошелька и публикуется анонимно, то есть связан только с адресом в транзакции (рис. 2.6). Открытые ключи включаются в состав большинства транзакций, что вызвано необходимостью проверки подписи владельца при совершении платежа.

Способ получения Биткойн-адреса на основе открытого ключа исключает необходимость использования цифровых сертификатов, поскольку всегда можно однозначно связать открытый ключ с соответствующим ему адресом, и ни с каким другим.

С точки зрения обеспечения анонимности Биткойн-адреса являются псевдонимами пользователей системы. Если удается связать Биткойн-адрес с конкретным человеком, то эта персонализация будет справедлива для всех транзакций с использованием этого адреса.

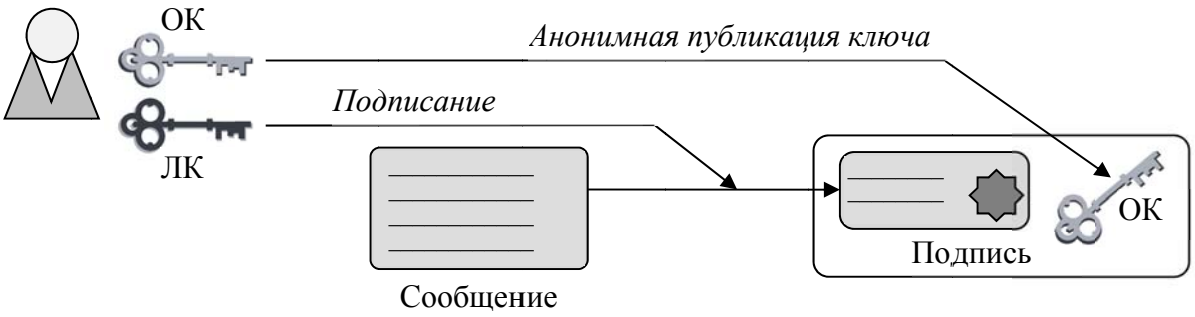



Рисунок 2.6 – Электронная подпись в Биткойн

Можно сказать, что анонимность транзакций поддерживается только в рамках сети Биткойн. Использование биткойнов на криптобиржах или в магазинах, принимающих криптовалюту, в большинстве случаев потребует стандартной идентификации владельца криптосредств. Кроме того, такая связь может быть установлена с помощью косвенной информации о пользователе, например, через историю браузера [3] или используемые IP-адреса [4].

В основе атак на анонимность лежит статистическое сопоставление между связанными наборами данных, что определяет сложность задачи обеспечения анонимности активности пользователей (узлов) сети. Таким образом, имеется возможность отследить связь Биткойн-адресов друг с другом, а также перемещение значительных объемов ценностей в сети [5]. Кроме того, имеется возможность вычислить баланс счета, сопоставленного с одним открытым ключом, а также агрегировать балансы, принадлежащие открытым ключам, контролируемым конкретным пользователем.

Деанонимизацию адреса можно провести путем его связывания с некоторой внешней идентифицирующей информацией, такой как почтовые и электронные адреса, номера платежных карт и банковские реквизиты, IP-адреса и др.

Это значит, что крупные централизованные поставщики Биткойн-сервисов, такие как криптобиржи и службы кошельков, способны отслеживать и персонифицировать значительную часть Биткойн-транзакций.

Для повышения конфиденциальности можно создавать отдельные адреса для каждой транзакции. Это осложняет сопоставление адресов с конкретным владельцем.

*Блокчейн*

Информация об эмиссии биткойнов и передаче имеющихся в сети биткойнов между ее участниками, то есть транзакциях в сети Биткойн, хранится в открытом (нешифрованном) виде в распределенном децентрализованном реестре – блокчейне (block chain, цепочка блоков). Каждый новый блок рассчитывается на основании предыдущего, а фактически – всей сформированной на текущий момент цепочки. Достигается это посредством хэширования информации блока (рис. 2.7).

Блок включает в себя список проверенных транзакций и заголовок, содержащий набор служебных полей, включая ссылку на хэш предыдущего блока. Для каждой включаемой в блок транзакции вычисляется ее хэш-код, кроме того, вычисляется хэш для всего вновь созданного блока, с учетом как информации о транзакциях, так и полей заголовка. Таким образом, в каждом новом блоке отражены все предыдущие операции блокчейна, включая самый первый блок. Благодаря этому практически невозможно подменить или удалить какой-нибудь блок из середины цепочки, поскольку это потребовало бы пересчета хэш-значений всех последующих

блоков. Такой пересчет оказывается слишком трудоемким, то есть требует недостижимых вычислительных и временных ресурсов.


Рисунок 2.7 – Упрощенная схема блокчейна

Целостность реестра обеспечивается также его реплицированием между всеми участниками сети. Как только кто-то вносит изменения в блокчейн, его копия сразу же проверяется и рассылается всем остальным пользователям системы. Каждый участник, имеющий доступ к распределенному реестру, хранит у себя его полную и актуальную копию. Поэтому факт подмены тут же будет обнаружен другими участниками сети. В настоящее время размер реестра составляет более 250 Гб, при этом размер реестра постоянно увеличивается.

Такая архитектура делает блокчейн чрезвычайно отказоустойчивой системой, однако не позволяет вносить никаких изменений в отношении ранее проведенных операций. То есть невозможна отмена подтвержденной операции, даже в случае ошибки или мошенничества (например, когда платеж был отправлен на ошибочный или несуществующий адрес, либо для подтверждения транзакции был использован скомпрометированный личный ключ). Таким образом, блокчейн технология не предусматривает наличия механизмов отмены подтвержденной операции, а проведенные транзакции необратимы.

Однако возможно использование групповых подписей, что позволяет привлечь для выполнения транзакции арбитра и обеспечить возврат биткойнов в случае невыполнении контрагентами оговоренных условий.

*Структура транзакции*

Транзакция – это подписанное владельцем сообщение о переходе прав на монеты биткойн. Сами монеты являются виртуальными и физически по сети не передаются.

Для передачи биткойнов их текущий владелец создает новую транзакцию, которая, помимо указаний о количестве передаваемых монет, содержит подписанный инициатором хэш предыдущей транзакции, по которой биткойны были получены. Предыдущая транзакция становится «входом» текущей транзакции. В качестве «выхода» указывается публичный ключ или Биткойн-адрес нового владельца (получателя) монет (рис. 2.8).

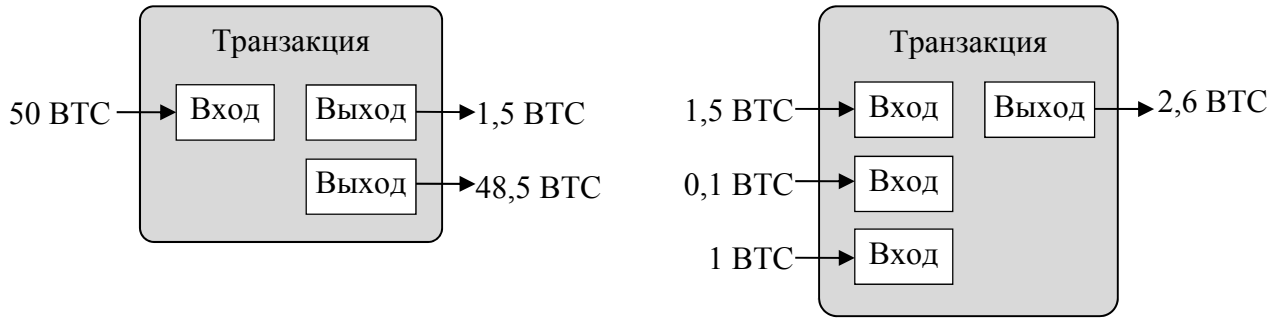



Рисунок 2.8 – Пример транзакций с множественными выходами и входами

Особенностью протокола Биткойн является необходимость использования всей суммы «входа», то есть невозможно перевести лишь часть биткойнов из суммы входной транзакции. Однако входную сумму можно распределить на несколько «выходов», один из которых может указывать на этот же адрес, то есть часть биткойнов будет передана самому себе как «сдача».

*Подтверждение транзакции и децентрализованный консенсус*

Сформированные транзакции не актуализируется сразу же после выполнения операции. Вместо этого они транслируются в сеть и какое-то время считаются не завершенными. Только после подтверждения транзакции и включения в распределенный реестр нового блока с данной транзакцией можно говорить о ее завершении (на самом деле рекомендуется дождаться формирования последовательности еще из пяти-шести блоков).

Поскольку блокчейн-система не опирается на модель централизованного доверенного органа, распределенные узлы должны согласовывать подтверждение (валидацию) транзакции. Суть задачи сводится к проблеме «византийских генералов», то есть принятии коллегиального решения при отсутствии доверия между сторонами.

Участник, сформировавший блок, проверяет поступившие транзакции на основе правил консенсуса сети Биткойн (транзакции должны иметь стандартный формат, содержать корректные подписи, должны быть исключены двойные траты и т.д.), а затем включает транзакции в блок, и отсылает его другим участникам сети. После проверки и подтверждения большинством остальных участников блок записывается в блокчейн.

При этом если нарушитель сможет контролировать боле половины (51%) узлов сети, выполняющих подтверждение транзакций, он сможет создать поддельную цепочку блоков, добиваясь подтверждения только своих блоков, и отвергая чужие. Для того, чтобы такая атака стала невозможной, в Биткойн используется концепция консенсуса на основе доказательства выполненной работы (PoW, Proof-of-work). Это значит, что формирование блока является затратным с вычислительной точки зрения.

Технически концепция PoW в сети Биткойн сводится к удовлетворению определенных ограничений на вид формируемых блоков. Хэш-значение заголовка блока транзакций (рис. 2.9), с учетом подбираемого параметра (nonce) должно быть меньше или равным целевому значению сложности (target). Это условие фактически задает число ведущих нулей в записи хэш-значения.

| | |
|---|---|
| Prev_block | Хэш предыдущего блока |
| Merkle_root | Хэш транзакций, включенных в блок |
| Timestamp | Временная метка (дата и время создания этого блока) |
| Bits | target – значение, регулирующее сложность получения новых блоков |
| Nonce | Подбираемый параметр |
| Txn_count | Количество транзакций в блоке |
| Список транзакций | |

Рисунок 2.9 – Упрощенная схема блока сети Биткойн

Чем меньше целевое значение target, тем меньше вероятность выполнения условия со случайным значением nonce ($P = (n–m)!/n!$, где $n$ – разрядность хэш-значения, $m$ – число ведущих нулевых разрядов). Поскольку хэш-функция ведет себя как псевдослучайная функция, получить подходящее значение nonce можно только перебором, что требует соответствующих вычислительных затрат.

Таким образом, варьируя значение сложности target, можно сделать задачу генерации правильного блока более или менее трудно решаемой. Значение target устанавливается в сети Биткойн автоматически таким образом, чтобы генерация нового блока занимала, в среднем, около десяти минут. Пересчет значения target производится каждые 2016 блоков (примерно раз в две недели). Если блоки формируются быстрее, то после пересмотра значения target достичь цели становится труднее, и наоборот. Поэтому изменение суммарной вычислительной мощности сети лишь очень незначительно влияет на количество создаваемых блоков.

Для того, чтобы пользователи сети проверяли транзакции, тратя свои вычислительные ресурсы, нужен какой-то стимул. Таким стимулом является вознаграждение в биткойнах за создание нового блока, а также комиссионные сборы, которые могут устанавливаться добровольно инициатором транзакции. Первая транзакция в блоке всегда формируется автоматически и передает вознаграждение за создание нового блока, последующие транзакции выбираются из очереди еще не проверенных и не записанных в блокчейн транзакций создателем блока на свое усмотрение, при этом он может отдавать предпочтение транзакциям с комиссией. Таким образом, транзакции с комиссией будут подтверждаться (а значит и выполняться) быстрее.

Деятельность по созданию новых блоков ради возможности получить вознаграждение в форме эмитированных биткойнов и комиссионных сборов получила название «майнинг» (англ. mining – добыча полезных ископаемых).

За каждый проверенный блок сделок успешный майнер получает вознаграждение, размер которого первоначально был установлен на уровне 50 биткойнов. Но после каждых 210 тысяч проверенных блоков (примерно, раз в четыре года) награда уменьшается вдвое. Уменьшение размера вознаграждения произошло уже трижды – в 2012, 2016 и 2020 году.

Таким образом, размер вознаграждения составляет убывающую геометрическую прогрессию {50; 25; 12,5; 6,25; …}, а общий объем эмиссии биткойнов ограничен (как сумма членов убывающей геометрической прогрессии) и составляет порядка 21 млн BTC. Предполагается, что эмиссия остановится к 2140 году, поскольку награда за блок не сможет превышать $10^8$ BTC, однако и до этого основным источником вознаграждения за формирование новых блоков станут комиссионные сборы.

Так, снижение награды за добычу биткойна весной 2020 года было отчасти компенсировано ростом комиссий – до снижения вознаграждения единичная транзакция в среднем обходилась пользователю в 50 центов, а к середине августа 2020 года комиссии выросли более чем в 10 раз, до $5,5.

Следует отметить, что вероятность успешного формирования блоков (и получения вознаграждения) тем выше, чем больше вычислительная мощность майнера, а точнее его доля в общей мощности сети, поэтому для майнинга используется специальное оборудование (специализированные ASIC процессоры), а общая вычислительная мощность сети растет, однако этот процесс регулируется размером вознаграждения и стоимостью криптовалюты (так, сразу после снижения вдвое вознаграждения в 2020 году совокупная вычислительная мощность сети Биткойн упала на четверть, а затем с ростом курса криптовалюты восстановилась почти в полном объеме) [6].

После создания блока он рассылается другим участникам сети, которые после проверки включают его в свои локальные копии блокчейн-реестра. Такую проверку производят не только другие майнеры, но и все узлы, хранящие полные актуальные копии реестра. Работа таких узлов во многом обеспечивает стабильность сети, поскольку неверные блоки (например, включающие двойные траты) сразу будут обнаружены и отклонены.

При формировании блоков могут возникнуть ситуации, когда несколько майнеров сформировали корректные блоки одновременно. В этом случае несколько новых блоков будут считать предыдущим один и тот же блок, то есть произойдет ветвление цепочки блоков. Решением данной ситуации является продолжение майнерами той ветви, которая включает более длинную цепочку блоков (рис. 2.10). Блоки, входящие в более короткие цепочки (orphan blocks), отбрасываются, а входящие в них транзакции переходят в очередь неподтвержденных.

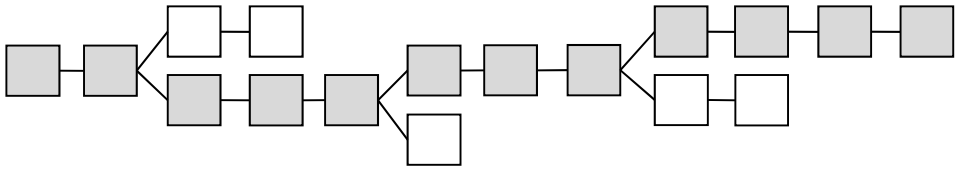


Рисунок 2.10 – Отсечение побочных ветвей цепочки блоков

Тогда, чтобы навязать свою цепочку блоков, нарушителю придется сделать ее длиннее остальных веток, обогнав других майнеров. То есть нарушитель должен обладать вычислительной мощностью, превосходящей половину мощности всей сети. Таким образом, степень защищенности сети определяется ее совокупной вычислительной мощностью.

С другой стороны, стремление повысить шансы на успех за счет увеличения вычислительной мощности склоняет майнеров к объединению в майнинговые пулы, у большинства из которых имеется ведущая компания. Таким образом, наблюдается тенденция к централизации процесса валидации новых транзакций, что противоречит основному принципу безопасности блокчейн-сети.

Следует отметить, что, если будет отбрасываться слишком много блоков, часть вычислительных мощностей честных участников сети будет тратиться впустую. Тогда нарушителю достаточно контролировать уже не половину, а меньшую долю всех вычислительных мощностей. Для того, чтобы снизить число отбрасываемых блоков и обеспечить своевременную синхронизацию копий реестра, пропускная способность сети искусственно снижается (например, вводится ограничение на скорость генерации новых блоков – примерно 1 блок в 10 секунд в Биткойн). С ростом числа

транзакций при фиксированном размере блока это может оказать существенное влияние на скорость обработки транзакций.

Вынужденное снижение пропускной способности наряду с необходимостью существенных затрат физических ресурсов (электроэнергии) для поддержания PoW консенсуса называют в качестве основных недостатков Биткойн.

Для того, чтобы обеспечить, с одной стороны, высокую пропускную способность сети, а другой – необходимый уровень безопасности, используют более сложные способы построения цепочек (в виде древовидных структур, направленных ациклических графов) и критерии отбрасывания блоков. Примерами являются протоколы Ghost (использовался в системе Ethereum, 2013), Spectre, Phantom [7].

PoW является не единственной концепцией коллективного консенсуса в блокчейн-системах. К недостаткам блокчейн-систем, построенных на PoW консенсусе, принято относить:

- значительную трату физических ресурсов (электроэнергии), не приносящая никакой пользы для реального мира;
- дороговизну майнинга (оборудования, электроэнергии), что снижает его демократичность;
- недостаточную децентрализацию, что обусловлено тенденцией к концентрации вычислительных мощностей.

Следует отметить, что с целью сохранения принципа децентрализации в отличных от Биткойна протоколах, основанных на PoW консенсусе (криптовалюты Litecoin, Worldcoin и др.), вместо обычной хэш-функции при генерации блока используется функция scrypt. Это делается для того, чтобы участники, обладающие специализированным оборудованием (на ASIC процессоров), не получали преимуществ при майнинге. Функция scrypt основана на PBKDF2 (то есть многократном последовательном вычислении хэш-функции с разными параметрами), однако реализована таким образом, чтобы требовать большого количества оперативной памяти [8]. Реализации scrypt, не требующие увеличения объема доступной памяти, слишком медленны.

Тем не менее разработчики GPU и ASIC находят решения и для scrypt майнинга. Для майнинга scrypt криптовалют, так же, как и для Биткойна, распространенным явлением является объединение в пулы. Причем пулы могут как ориентироваться только на одну монету, так и быть мультивалютными, что позволяет майнеру переключаться между разными проектами.

В настоящее время активно развивается модель консенсуса PoS (Proof of Stake, доказательство доли владения). В соответствии с ней вероятность получения права на формирование блока зависит от доли участника, то есть определяется объемом вложенной криптовалюты. Выбор уз-

ла в качестве валидатора включает фактор случайности, однако учитывает богатство узла и возраст монеты (как долго монеты заблокированы или находятся в доле).

Безопасность PoS консенсуса строится на предположении, что участникам, которые владеют большим количеством монет, невыгодно атаковать систему, поскольку в этом случае произойдет обесценивание криптовалюты, и они потеряют свои накопления. Дополнительно может быть предусмотрены блокировка нечестного участника и аннулирование его замороженной доли. Вместе с тем, здесь нет затрат реальных физических ресурсов (вложений в оборудование для майнинга, затрат на электроэнергию), поэтому обоснование PoS многим кажется менее убедительным по сравнению с PoW.

Есть еще одно существенное возражение против принципов PoS консенсуса – это «несправедливость» экономики. Поскольку вероятность получения права на генерацию новых блоков и, соответственно, выплаты за них, пропорциональны накоплениям, шанс на увеличение богатства выше у тех участников, которые и так владеют большими вкладами. Миноритарные же владельцы монет такого шанса практически не имеют.

Таким образом, основными проблемами PoS консенсуса являются:

- «ничего на кону», когда нечестные участники не рискуют реальными ценностями;
- «богатство для богатых», когда накопления концентрируются в руках мажоритарных владельцев монет, что не способствует децентрализации сети.

Несмотря на наличие практических реализаций PoS консенсуса (протокол Ouroboros в криптовалюте Cardano, гибридный протокол PoW/PoS консенсуса Casper в Ethereum), PoW по-прежнему считается самым легким и в то же время самым стабильным алгоритмом децентрализованного консенсуса.

*Групповые подписи*

Групповые подписи (мультиподписи) позволяют требовать заверения несколькими личными ключами для выполнения транзакции. Простейшим примером является использование депозитной ячейки, деньги из которой могут быть переданы получателю только после подтверждения сделки обеими сторонами. Другим примером является необходимость принятия коллегиального решения о сделке (например, в случае крупного перевода), требующего утверждения несколькими руководителями компании.

В случае использования традиционной ECDSA-схемы проверка подписи каждой из сторон производится по-отдельности, с использованием соответствующего открытого ключа. При этом становится очевидной следующая информация о транзакции (рис. 2.11):

– факт существования транзакции с групповой подписью;
– число адресов, участвующих в групповой подписи;
– какие именно адреса участвовали в подписании транзакции.

Кроме того, несколько отдельных подписей увеличивают размер записи транзакции, что сказывается на размере комиссии.

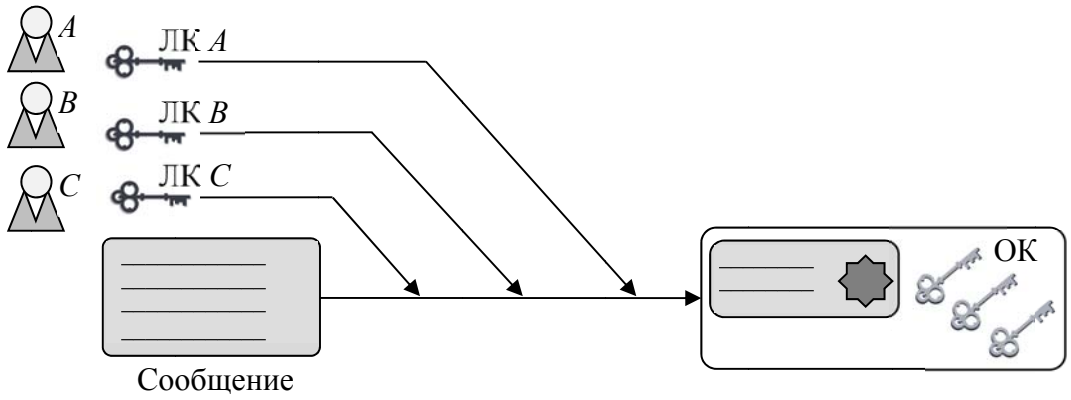



Рисунок 2.11 – Групповые подписи ECDSA в Биткойн-транзакции

В отличие от Биткойна, многие альтернативные криптовалюты используют подпись Шнорра, которая была защищена патентами до 2008 года. Подпись Шнорра позволяет естественным путем агрегировать открытые ключи для проверки подписи, позволяя проводить проверку групповой подписи одним общим (агрегированным) открытым ключом, скрыв таким образом, информацию как о количестве, так и об адресах подписавших, что значительно повышает анонимность транзакций.

Агрегирование ключей позволяет сделать транзакции с групповой подписью фактически неотличимыми от обычных транзакций (рис. 2.12). Использование одной подписи вместо нескольких позволяет сократить размер записи транзакции и повысить эффективность ее проверки.

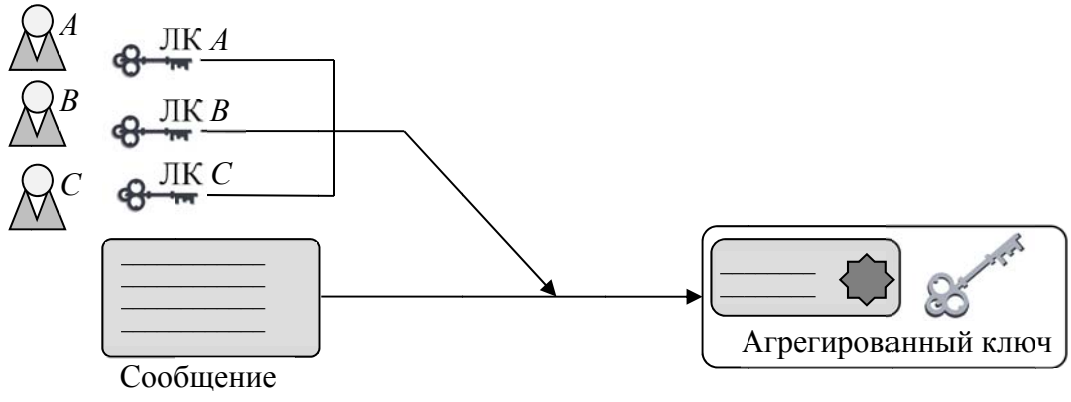



Рисунок 2.12 – Подписи Шнорра в Биткойн-транзакции

*Подпись Шнорра*

Подпись Шнорра является модификацией схем Эль-Гамаля и Фиата-Шамира с сокращенной длиной подписи и базируется на сложности задачи дискретного логарифмирования.

Выбирается большое простое число $p$, такое, чтобы число $p - 1$ имело достаточно большой простой делитель $q$, а также число $g$, такое, что $g^q \bmod p = 1$. В схеме электронной подписи используется хэш-функция $H$ со значением выхода, не превосходящим $q$.

Вычисление подписи к сообщению $M$, производится следующим образом.

1. Генерируется случайное число $k$, $1 < k < q$.

2. Вычисляется значение $r = g^k \bmod p$.

3. К сообщению $M$ дописывается значение $r$, а затем вычисляется хэш-функция $h = H(M \| r)$.

4. Вычисляется значение $s = (k - xh) \bmod q$, где $x$ – личный ключ подписывающего.

Значение подписи составляет пара значений $(h,s)$.

Проверка подписи под сообщением $M$ проводится следующим образом.

1. Вычисляется значение $r' = g^s y^h \bmod p$, где $y$ – открытый ключ подписывающего.

2. К сообщению $M$ дописывается значение $r'$, а затем вычисляется хэш-функция $h' = H(M \| r')$.

3. Выполняется сравнение значений $h$ и $h'$. При совпадении значений подпись считается подлинной.

В модификации этой схемы, где $s = (k + xh) \bmod q$, во время проверки рассчитывается значение $r' = g^s y^{-h} \bmod p$.

В случае участия нескольких сторон, подписание сообщения $M$ по схеме Шнорра может производиться последовательно, при этом:

1. Каждая из сторон генерирует свое случайное число $k_i$, $1 < k_i < q$.

2. Вычисляется значение $r = g^{ki} \bmod p$.

3. Вычисляется значение хэш-функции $h = H(M \| r)$.

4. Вычисляется значение $s = (k_i + x_i h) \bmod q$, где $x_i$ – личные ключи подписывающих.

Пара значений $(h,s)$ является групповой подписью, для проверки которой может использоваться агрегированный открытый ключ $y = y_i = g^{xi} \bmod p$.

Проверочное равенство имеет вид:

$g^s = ry^h$.

То есть проверяющая сторона должна рассчитать рассчитывается значение $r' = g^s y^h \bmod p$, $h' = H(M \| r')$, а затем проверить совпадение $h' = h$.

В самом деле, $g^s = g^{ki+xih} = g^{ki}(g^{xi})^h = r(y_i)^h = ry^h$.

Этот упрощенный алгоритм, допускает, однако атаку, в которой нечестный участник устанавливает открытый ключ вида, что позволяет ему подписывать сообщения от имени всей группы.

Модификация этого алгоритма, предложенная в работе [9] и названная авторами MuSig, принята в начале 2020 года в качестве официальных предложений по улучшению Биткойна.

В подписи MuSig первые два шага совпадают с рассмотренным выше алгоритмом, а затем каждым участником вычисляются частные значения хэш-функции $h_i$ и подписи $s_i$.

$a_i = H_{agg}(L,y_i)$ – значение, зависящее от открытого ключа подписавшего,

где $L$ – список открытых ключей участников группы, представленный в виде строки данных в любой кодировке: $L = \{y_1,y_2,\ldots,y_n\}$.

$h_i = a_i \ h$,

$h = H_{sig}(Y, r, M)$,

где $Y$ – агрегированный открытый ключ .

$s_i = k_i + x_i h_i$

Значение $s$ получается тогда, как $s = (k_i + x_i h_i)$,

а проверочное равенство – $g^s = rY^h$

$g^s = g^{(ki + xihi)} = g^{ki}(g^{xi})^{aih} = r(y_i^{ai})^h = rY^h$.

Эта схема является доказуемо безопасной в предположении, что все участники независимо выбирают случайные значения $k_i$ до получения значений $r_i = g^{ki}$ от других подписывающих. Кроме того, необходимо обеспечить смену случайного значения $k_i$ при повторной попытке подписи. В противном случае участники подписи могут осуществить атаку, нацеленную на восстановление личного ключа [9].

Нарушителю для подделки подписи необходимо знать открытые ключи всех участников подписавшей группы, в то время как для проверки подписи потребуется только агрегированный открытый ключ.

*Применение блокчейн технологии*

Биткойн стал первой реализацией блокчейн-системы, предложив успешную модель децентрализации, однако на сегодняшний день существует множество других реализаций, использующих иные алгоритмы построения цепочек блоков и достижения коллективного консенсуса, а область применения распределенных реестров не ограничивается операциями с криптовалютой.

Построенные на блокчейне системы хранения данных могут использовать различные схемы организации доступа. Если обычным участникам информация доступна только для чтения, а право на администрирование, управление данными, аудит доступно только привилегированным участникам, говорят о приватном (закрытом) блокчейне. Генерация новых блоков здесь выполняется централизованно очерченным кругом доверенных

узлов. Такой тип блокчейн-систем чаще используется для внутрикорпоративных проектов. Централизация приватных блокчейнов снижает степень их защищенности от нечестных участников.

Характерной особенностью публичных (открытых) блокчейн-систем является коллективный контроль над системой, когда решения принимаются коллегиально всеми участниками. Поскольку какой-либо центральный орган, принимающий административные решения, в публичном блокчейне отсутствует, блокировка участника или откат совершенной транзакции, пусть даже и мошеннической, невозможны, если такое решение не поддержано большинством участников сети (примером является разделение сети Ethereum на две ветки после крупного инцидента в 2016 году).

В Российской Федерации принят федеральный закон от 31.07.2020 №259-ФЗ «О цифровых финансовых активах, цифровой валюте и о внесении изменений в отдельные законодательные акты Российской Федерации», вступающий в действие с 1 января 2021 года. Закон содержит понятия цифровых финансовых активов (ЦФА) и цифровой валюты, создает основу для выпуска зарегистрированными организациями российских цифровых активов на частных блокчейнах. Операторами обмена ЦФА смогут стать российские банки и биржи, если они будут включены в специальный реестр, регулируемый Банком России. Кроме того, цифровые активы и цифровые валюты получили в России законный статус в качестве имущества.

В то же время к публичным блокчейнам законодатели относятся более прохладно. Согласно закону, децентрализованные криптовалюты не могут считаться платежным средством на территории России. Это означает, что российским резидентам запрещено принимать цифровые валюты в качестве оплаты за товары и услуги, так же, как и распространять информацию о возможности расплатиться цифровой валютой. Ожидается, что остальные аспекты регулирования децентрализованных криптовалют будут определены еще одним законом, который пока находится в стадии законопроекта, – «О цифровой валюте».

Основное преимущество технологии блокчейн – обеспечение прямого, непосредственного взаимодействия участников без необходимости привлечения посредников (центров доверия) для подтверждения сделки в условиях отсутствия доверия между сторонами и в недоверенной среде. Неизменность транзакции в блокчейне гарантирует алгоритмическое подтверждение обязательств, при этом транзакции могут не только сводиться к передаче цифровых монет, а реализовывать более сложные алгоритмы. Основной функциональной нагрузкой блокчейн-систем стали смарт-контракты – реализованные в виде программного кода наборы формализованных правил, выполнение которых влечет за собой исполнение неко-

торых обязательств в реальном мире или цифровых системах. При этом поскольку блокчейн создает среду, обладающую свойством неизменности данных, ход исполнения смарт-контракта также не подлежит изменению.

По оценкам аудиторской компании PwC массовое внедрение технологии блокчейна может принести мировой экономике до $1,76 трлн к 2030 году и создать порядка 40 млн новых рабочих мест [10].

Вместе с тем, следует понимать, что блокчейн не является универсальной технологией, прежде всего ввиду необратимости транзакций, то есть заявленные сделки не подлежат отмене или пересмотру условий. Кроме того, технология блокчейн имеет ряд технических ограничений: малая пропускная способность сети, невозможность хранения большого объема данных, постоянный рост самого реестра и др.

В конце 2019 года Д. Чаум заявил о принципиальной невозможности совместного обеспечения существующими блокчейн системами таких качеств как децентрализация, высокая скорость обработки транзакций, анонимность и безопасность в долгосрочной перспективе, что связано с появлением угрозы практической реализации квантовых вычислений [11]. В качестве решения Д. Чаумом предложен протокол цифровой валюты Praxxis, основанный на новом квантово устойчивом алгоритме коллективного консенсуса.

Существует мнение, что смарт-контракты могут быть использованы для создание финансовых пирамид, выявление и прекращение деятельности которых осуществить значительно сложнее, чем в традиционных финансовых системах. Перенос готовых решений на новые предметные области без учета присущей им бизнес-логики заведомо обречен на неудачу. Теория и практика блокчейна требуют своего дальнейшего развития, в том числе и с точек зрения информационной и экономической безопасности.

Список использованных источников:

1. Шнайер Б. Прикладная криптография: Протоколы, алгоритмы, исходные тексты на языке Си. – М.: Триумф, 2012. – 815 с.
2. Технология блокчейн и Биткоин – 23.01.2020 [электронный ресурс]. URL: https://intuit.ru/studies/professional_skill_improvements/22093/info (дата обращения: 17.10.2020).
3. Нефедова Мария. Mozilla: истории браузера достаточно для надежной идентификации пользователя – 03.09.2020 [электронный ресурс]. URL: https://xakep.ru/2020/09/03/browsing-history/ (дата обращения: 17.10.2020).
4. D. Kaminsky. Black Ops of TCP/IP Presentation. Black Hat, Chaos Communication Camp, 2011 [электронный ресурс]. URL: https://www.blackhat.com/presentations/bh-asia-02/Kaminsky/bh-asia-02-kaminsky.pdf (дата обращения: 17.10.2020).
5. Fergal Reid, Martin Harrigan. An Analysis of Anonymity in the Bitcoin System – 7 May 2012 [электронный ресурс]. URL: https://arxiv.org/abs/1107.4524 (дата обращения: 17.10.2020).

6. Фомин Дмитрий. 3 месяца спустя. Как изменилась прибыльность майнинга после халвинга // РБК – 19 августа 2020 [электронный ресурс]. URL: https://www.rbc.ru/crypto/news/5f3d496f9a79477d5fce7220 (дата обращения: 17.10.2020).
7. Distributed Lab. Обзор актуальных протоколов достижения консенсуса в децентрализованной среде – 3 августа 2018 [электронный ресурс]. URL: https://habr.com/ru/company/distributedlab/blog/419185/ (дата обращения: 17.10.2020).
8. RFC 7914. The scrypt Password-Based Key Derivation Function – August 2016 [электронный ресурс]. URL: https://tools.ietf.org/html/rfc7914.html (дата обращения: 17.10.2020).
9. Gregory Maxwell, Andrew Poelstra, Yannick Seurin, Pieter Wuille. Simple Schnorr Multi-Signatures with Applications to Bitcoin – May 20, 2018 [электронный ресурс]. URL: https://eprint.iacr.org/2018/068.pdf (дата обращения: 17.10.2020).
10. Фомин Дмитрий. PwC оценила эффект от повсеместного внедрения блокчейна в $1,7 трлн // РБК – 13 октября 2020 [электронный ресурс]. URL: https://www.rbc.ru/crypto/news/5f85674e9a79476774e454ae?from=newsfeed (дата обращения: 17.10.2020).
11. Praxxis. Nechnical Paper – December 2019 [электронный ресурс]. URL: https://xx.network/praxxis-technical-paper-v1.pdf (дата обращения: 17.10.2020).

## 2.6. Проблемы защиты персональных данных в Интернете вещей

На сегодняшний день сети малогабаритных устройств, которые все чаще применяются как «умные вещи» в Интернете вещей, приобретают повсеместное развитие. Несмотря на то, что мы практически каждый день сталкиваемся с такими устройствами в реальной жизни, термин Интернет вещей для многих остается новым. В нашей стране с 2012 года ведутся исследования в этом направлении, в частности, в секторе стандартизации МСЭ-Т Исследовательской комиссией ИК13 («Новое поколение сетей») [1].

В 2020 г. Росстандарт утвердил серию предварительных национальных стандартов в области Интернета вещей, сенсорных сетей и промышленного интернета вещей. Документы были разработаны техническим комитетом «Киберфизические системы» при поддержке Минпромторга России.

Как правило, Интернет вещей – это совокупность подключенных к Интернету посредством беспроводных технологий малогабаритных устройств, способных собирать, а иногда и анализировать данные, циркулирующие в сети. Если говорить, например, о технологии «умного дома», то в первую очередь обрабатываемой информацией в такой сети являются персональные данные.

Выход на рынок большого числа новых устройств разных производителей, а, следовательно, значительное уменьшение цен на устройства, при существенном удобстве пользования и автоматизации многих бытовых процессов приводит к стремительному расширению этого сектора

рынка телекоммуникаций. Разрабатываемые и принимаемые стандарты в области Интернета вещей, к сожалению, не охватывают вопросы, связанные с предъявлением требований по информационной безопасности, поэтому не только у рядовых пользователей, но и у производителей и интеграторов нет представления об общей модели защищенного взаимодействия в Интернете вещей.

Если рассматривать Интернет вещей на примере концепции «умного дома», то на первый план выходят угрозы конфиденциальности обрабатываемой информации. Угрозы целостности не являются столь актуальными, поскольку их реализация может привести лишь к искажению передаваемых команд между устройствами, чего нельзя сказать при рассмотрении таких угроз в рамках промышленного Интернета вещей. Угрозы доступности так же не могут привести к критичному событию в рамках «умного дома», потому что вывод из строя отдельных элементов может лишь приостановить на время выполнение каких-либо бытовых функций, которые не являются жизненно важными.

Защита персональных данных в Российской Федерации регламентируется Федеральным законом «О персональных данных» от 27.07.2006 №152-ФЗ, а также Постановлениями правительства и руководящими документами ФСТЭК. Настоящие нормативные документы определяют порядок обращения с персональными данными, ответственность за нарушение в этой сфере, определяют требования по обработке персональных данных в государственных информационных системах. Основным недостатком существующей нормативной базы является отсутствие систематизированного определения методов и мер по защите персональных данных при их обработке, хранении и передачи, в том числе, при личном пользовании. Руководствуясь существующими нормативно-правовыми документами невозможно определить необходимый и достаточный перечень требований для их реализации в устройствах «умного дома». Более того, нет единого подхода к определению защищаемой информации, то есть, нет определения того, что из передаваемой между датчиками и сенсорами в Интернете вещей, является персональными данными или другой конфиденциальной информацией.

Определим перечень информации, циркулирующей в сети «умного дома» (таблица 2.1).

Таблица 2.1 – Перечень циркулирующей в сети «умного дома» информации

| Вид информации | Степень конфиденциальности |
|---|---|
| Команды управления | открытая информации, при условии, что в командах не содержится другая конфиденциальная информация |

Окончание табл. 2.1

| Вид информации | Степень конфиденциальности |
|---|---|
| Данные о пользователе – местонахождение, история поисковых запросов, личная переписка и др. | информация, которая по определению действующих нормативно-правовых актов не может быть отнесена к одному из видов конфиденциальной информации, однако ее утечка может нанести ущерб ее обладателю |
| Данные платежных карт | конфиденциальная информация, которая по определению не относится к персональным данным, однако может нанести материальный ущерб ее обладателю |
| Голос | биометрические персональные данные |

Таким образом, из таблицы 2.1 видно, что устройства Интернета вещей обрабатывают различные категории информации – от той, конфиденциальность которой не установлена на законодательном уровне, до наивысшей категории персональных данных. При этом ни один существующий стандарт не определяет необходимых мер защиты даже для той информации, степень конфиденциальной которой определена.

Основной проблемой в защите персональных данных является даже не несовершенство законодательства, а отсутствие знаний у обычных пользователей, на которых в первую очередь и возлагается обязанность по защите персональных данных.

В 2016 году случился известный прецедент судебного разбирательства с Linkedln за нарушение ФЗ №152, в рамках которого сервис обвинили в использовании и передачи данных о местоположении и поведении пользователей на сайте без их согласия. При этом в ФЗ №152 дается следующее определение персональных данных: «персональные данные – любая информация, относящаяся прямо или косвенно определенному или определяемому физическому лицу (субъекту персональных данных)» [2]. Таким образом, исходя из судебной практики, можно сделать вывод о том, персональными данными является история поведения пользователя на сайте – так называемые cookies. Сегодня, заходя на любой сайт, где пользователи оставляют свои данные или они собираются автоматически, можно увидеть уведомление об использовании cookies, а также форму, в которой пользователь должен проставить «галочку» и согласиться с обработкой его персональных данных. Законом не установлена такая форма предоставления согласия, однако Роскомнадзор дал разъяснение, что проставление «галочки» в веб-форме может быть приравнено к получению согласия, однако не для специальных и биометрических категорий персональных данных [3]. Кроме того, на сайте, сервисе или приложении должна быть размещена политика обработки персональных данных.

Таким образом, при развертывании сети «умного дома» важно понимать, где и как будет храниться и обрабатываться передаваемая в сети информация.

На сегодняшний день существует достаточно большое количество производителей и интеграторов систем «умного дома» как отечественных, так и зарубежных: Яндекс, Amazon, Google, Xiaomi, Apple, Samsung, Fibaro, Z-Wave, ZigBee и др. При выборе платформы масштаб и известность производителя не должны быть решающими, так как, например, одна из крупнейших облачных платформ Amazon в 2019 году была участником крупной утечки персональных данных пользователей Facebook.

Пока не разработаны стандарты, определяющие технические подходы к защите данных в сетях Интернета вещей, субъекты персональных данных должны самостоятельно комплексно подходить к организации таких сетей внутри «умного дома». Вопросы защиты информации в более крупных сетях уровня «умных городов» требуют непосредственного вмешательства в проработку этих вопросов со стороны регуляторов.

В 2018 году на смену европейскому регламенту по защите персональных данных пришел регулирующий акт Евросоюза «Общие положения по защите данных» (General Data Protection Regulation, GDPR). Для многих пользователей – субъектов персональных данных, не раскрытым остается вопрос, как действует этот регламент в нашей стране и можно ли защитить свои данные в рамках GDPR.

Требования стандарта GDPR обязаны соблюдать те компании, которые находятся на территории Евросоюза, или же обрабатывают данные хотя бы одного пользователя, находящегося на территории ЕС, даже если он не является гражданином ЕС. С принятием GDPR многие российские компании – операторы персональных данных, были вынуждены пересмотреть подходы к организации обработки персональных данных во исполнение требований нового стандарта. Для пользователей – субъектов персональных данных, рассчитывать на дополнительную защиту в рамках GDPR можно лишь в случае, если они передают свои данные компании или сервису, находящемуся на территории ЕС. При этом нельзя однозначно ответить на вопрос, безопасней ли разворачивать сеть «умного дома» в облачной платформе зарубежных производителей, чем в отечественных. Несмотря на различия европейского стандарта и российского законодательства, GDPR так же не определяет конкретных технических мер, способных определить адекватную защиту персональных данных. При этом в GDPR понятие персональных данных является более широким, штрафы за несоблюдение требований значительно больше, а в случае утечки данных оператор обязан уведомить пользователей в течение 72 часов.

Таким образом, на сегодняшний день проблемы защиты персональных данных остаются открытыми, в частности в таком развивающемся сегменте телекоммуникационного рынка, как Интернет вещей. Ответственность за реализацию адекватной защиты персональных данных лежит на самом обладателе – субъекте персональных данных. Технологии

стремительно развиваются, позволяя пользователям упрощать многие процессы, поэтому отказываться от их использования не представляется возможным. Для реализации первичных мер защиты конфиденциальной информации должны быть решены следующие задачи.

1. Проработка на законодательном уровне организационно-технических мер, направленных на определение профилей защиты устройств, применяемых для обработки конфиденциальной информации в Интернете вещей.
2. Повышение уровня осведомленности пользователей – субъектов персональных данных, об основных сведениях в области защиты персональных данных.
3. Организация сетей «умного дома» на платформах, где проработаны вопросы защиты данных, принята адекватная политика конфиденциальности.
4. Применение первоочередных правил защиты конфиденциальной информации – использование сложных паролей, своевременное обновление программного обеспечения, использование средств шифрования и антивирусной защиты.

Список использованных источников:

1. Рекомендация МСЭ-Т Y.2060. Обзор интернета вещей [электронный ресурс]. URL: https://www.itu.int/rec/T-REC-Y.2060-201206-I (дата обращения: 20.10.2020).
2. Федеральный закон от 27.07.2006 №152-ФЗ «О персональных данных».
3. Роскомнадзор напоминает о том, что персональные данные граждан России должны обрабатываться только с их согласия, 09.11.2017 [электронный ресурс]. URL: www.consultant.ru/law/hotdocs/51392.html/ (дата обращения: 20.10.2020).

## 2.7. Программные инструменты стеганографии для ОС Windows

В настоящее время информация является ресурсом, позволяющим получить преимущество, как в бизнесе, так и в политике.

Ценность информации определяется уровнем ее конфиденциальности. Владельцы конфиденциальной информации заинтересованы в сохранности и недоступности ее третьим лицам. Криптография является отличным способом обеспечить конфиденциальность информации, однако, зашифрованное сообщение несомненно привлечет внимание криптоаналитиков и станет целью криптографических атак, в тоже время стеганография обладает весомым преимуществом – маскировкой скрываемого сообщения в «безобидных» файлах. Никто не будет даже пытаться анализировать и искать в графическом или в мультимедийном файле что-то спрятанное, потому что потоки информации, которые проходят каждую

секунду по сети без преувеличения можно назвать колоссальными. Скрываемое сообщение при помощи методов стеганографии сможет «раствориться» в другой информации, в так называемом «контейнере». Контейнер или же стегоконтейнер – это вместилище для секретного сообщения. Он может принимать различные формы: от обычного текста до фотографий, видео, аудио и другого контента. В настоящее время существует большое количество готовых программных средств для создания стегоконтейнеров и внедрения в них информации. Для сравнения выбраны наиболее популярные программы, предназначенные для встраивания информации в готовые стеганографические контейнеры, анализа заполненности контейнера и извлечения скрытой информации из готовых стеганографических контейнеров – файлов в графических и мультимедийных форматах.

О популярности рассматриваемых программ можно судить по примерному количеству загрузок [4]:

- SSuite Picsel Security – примерное количество загрузок 1700;
- QuickStego – примерное количество загрузок 13000;
- DeepSound – примерное количество загрузок 5100;
- DeEgger Embedder – примерное количество загрузок 7308;
- SilentEye – примерное количество загрузок 1800.

SSuite Picsel Security – это бесплатный портативный инструмент для встраивания текста в графические файлы, не сохраняя при этом целостность файла. Особенностью программы является отображаемый промежуточный результат процесса сокрытия, где наглядно отображаются пиксели, в которые будет записана скрываемая информация. Преимуществами программы является простота использования и высокая скорость маскировки. Главным недостатком программы является метод встраивания – информация сконцентрирована в верхней части графического файла, из-за чего ее легко обнаружить с помощью специализированных инструментов. Заполненный контейнер можно сохранить только в формате bmp и png, поэтому при сокрытии информации в цифровых фотографиях, это приведет к созданию файлов большого размера, что также может привлечь внимание. Также инструмент не поддерживает шифрование. Кроме того, при анализе заполненного стегоконтейнера было обнаружено, что программа добавляет в структуру файла текст с нерабочей гиперссылкой, что выдает манипуляции с контейнером, а также для прочтения скрытого текста можно только при наличии оригинального файла [1].

QuickStego. Эта условно-бесплатная программа, которая может скрывать текстовые и графические файлы. Выходной формат файла только bmp, что является недостатком, так как в эпоху сети Интернет это довольно непопулярный формат графических файлов. Еще одним недостатком является отстуствие поддержки шифрования в бесплатной версия

программы. Преимущество данной программы заключается в сохранении целостности файла. Для сохранения целостности файла используются различные методы такие как: перерасчет хэша, изменение в метаданных и редактирование младших бит. Инструмент работает достаточно быстро и позволяет скрывать текст в графических файлах таким образом, чтобы только другие пользователи QuickStego могли извлекать и читать скрытые секретные сообщения. Секретное сообщение равномерно распределено по графическому файлу, что затрудняет его обнаружение. После того как текст скрыт в графическом файле, он не изменяет свой формат.

OpenStego – бесплатный инструмент для стеганографии, реализованный на Java, на основе файлов формата png, с поддержкой шифрования данных и сохраняющий целостность файлов. К достоинствам можно отнести поддержку популярных форматов таких как png, bmp, jpg, gif, а также то, что само секретное сообщение встраивается незаметно и сохраняется целостность файла. В настоящее время он поддерживает встраивание текстовых и графических файлов в другие графические файлы 24bpp [2].

Наряду со встраиванием информации в файлы графического формата используется встраивание в мультимедийные файлы, поэтому многие современные инструменты для стеганографии поддерживают кодирование сообщения в аудиофайлы.

DeepSound – бесплатная программа, предназначенная для сокрытия и извлечения секретных сообщений из аудиофайла. Инструмент поддерживает множество форматов аудиофайлов в качестве пустого контейнера, а в процессе встраивания информации преобразует контейнер в один из форматов со сжатием без потерь: flac, wav, ape. К достоинствам программы можно отнести возможность настройки качества выходного файла и сокрытие файлов большого размера – при низком качестве выходного файла возможно скрыть файл размером около 50% размера стегоконтейнера. Также инструмент сохраняет целостность аудиофайлов и поддерживает систему шифрования, основанную на AES-256, которая считается более совершенной системой шифрования по сравнению с другими. К недостаткам можно отнести медленную скорость встраивания, а также то, что при загрузке заполненного контейнера в программу, пользователю сразу предлагается ввести пароль, что выдает наличие скрытой информации в файле [1].

DeEgger Embedder – программа для стеганографии, поддерживающая большое количество форматов, что является ее главным преимуществом. Она принимает файл любого типа и объединяет его другим файлом любого формата (.avi, .jpg, .mp3, .mp4, .pdf, .png). Также программа сохраняет целостность файлов. Недостатком является алгоритм работы – программа дописывает скрытое сообщение в конец файла, а также не поддерживает шифрования [1].

SilentEye – это бесплатное кроссплатформенное приложение, упрощающее использование стеганографии, в данном случае скрытие текстовых и графических файлов в других графических или звуковых файлах (jpg, bmp, wav). Основными преимуществами является гибкая настройка алгоритмов встраивания информации, поддержка шифрования и возможность интеграции новых алгоритмов стеганографии и шифрования с помощью системы плагинов, а также сохранение целостности файлов. Недостатки инструмента заключаются в медленной скорости работы и низкой степени сокрытия информации в контейнерах малого размера [3].

В таблице 2.1 представлена сравнительная характеристика описанных выше стеганографических инструментов.

Таблица 2.1 – Сравнительные характеристики стеганографического ПО

| Характеристика | SSuitePicsel | OpenStego | SilentEye | DeepSound | DeEgger Embedder | QuickStego |
|---|---|---|---|---|---|---|
| Способ распространения | бесплатно | бесплатно | бесплатно | бесплатно | бесплатно | условно бесплатно |
| Входные форматы файлов | текст, графика | текст, графика | текст, графика | flac, ape и wav | текст, графика | текст, графика |
| Выходные форматы файлов | png, bmp | png | jpg, bmp. wav | flac, wav | jpg, bmp. wav | bmp |
| Целостность файла | нет | есть | есть | есть | есть | есть |
| Поддержка шифрования | нет | есть | есть | есть | нет | только в платной версии |
| Дополнительные опции | Отображение результата встраивания информации | – | Гибкая настройка алгоритма | Возможность выбрать качество выходного файла | – | – |
| Скорость работы | быстро | быстро | медленно | медленно | быстро | быстро |
| Заметность скрытого сообщения | заметно | не заметно | зависит от настроек | не заметно | заметно | не заметно |

Стеганография – эффективный способ защищенного общения. Пользователи сети могут сначала зашифровать конфиденциальный файл, а затем спрятать его внутри графического или мультимедийного файла, прежде чем отправлять адресату. Это снизит шансы перехвата, так как информацию при использовании данного метода очень сложно обнаружить.

Список использованных источников:

1. Стеганографические программы для сокрытия и защиты информации [электронный ресурс] // Блог Артема Чуйкова. URL: https://artem-guy.blogspot.com/2016/10/blog-post.html (дата обращения: 26.10.2020).
2. Прячем файлы в картинках: семь стеганографических утилит для Windows [электронный ресурс] //Хакер. URL: https://xakep.ru/2017/01/23/windows-stenographic-tools/ (дата обращения: 26.10.2020).
3. Топ 10 инструментов стеганографии для Windows 10 [электронный ресурс] // Information Security Squad. URL: https://itsecforu.ru/2018/10/29/топ-10-инструментов-стеганографии-для-windows/ (дата обращения: 26.10.2020).
4. Веб-архив программного обеспечения “Softpedia” [электронный ресурс]. URL: www.softpedia.com (дата обращения: 27.10.2020).

## 2.8. Генерация общего секретного ключа на основе искусственных нейронных сетей

Последние 10 лет неуклонно растет интерес к искусственным нейронным сетям (ИНС) – математическим моделям, пытающимся повторить работу нейронных связей биологического мозга. ИНС находят применение в решении широкого спектра зада: это разделение объектов по классам, нахождение сложных функциональных зависимостей, анализ изображений и многое другое.

Так что же такое нейронная сеть, и как она может использоваться в сфере защиты информации? Искусственная нейронная сеть – математическая модель, старающаяся повторить процессы передачи сигналов между нейронами в биологических организмах. Первыми такими моделями были нейронные сети У.Маккалока и У.Питтса. После разработки первых методов обучения спектр задач, решаемых нейронными сетями стал заметно шире: они стали использоваться в задачах прогнозирования сложных математических зависимостей (многослойные перцептроны), для распознавания образов на изображениях (сверточные нейронные сети) и др.

Одной из областей применения ИНС в информационной безопасности (ИБ) является область нейрокриптографии. Нейрокриптография – раздел криптографии, изучающий применение стохастических алгоритмов, в частности, нейронных сетей, для шифрования и криптоанализа. В криптоанализе используется способность нейронных сетей исследовать пространство решений. Также имеется возможность создавать новые типы атак на существующие алгоритмы шифрования, основанные на том, что любая функция может быть представлена нейронной сетью. Взломав алгоритм, можно найти решение, по крайней мере, теоретически. Так как в задаче нахождения сложных функциональных математических зависимо-

стей лучше всего справляется модель многослойного перцептрона, и поэтому далее мы будем рассматривать именно эту архитектуру ИНС.

Так что же такое искусственная нейронная сеть? ИНС представляет собой систему, состоящую из небольших процессоров – искусственных нейронов, выполняющих простые линейные комбинации входных сигналов. Математическая модель искусственного нейрона приведена на рисунке 2.13.

На вход каждого нейрона подается значение связанного с ним нейрона из предыдущего слоя ($x_1$, $x_2$, …, $x_n$), умноженного на весовой коэффициент связи, называемый синаптическим весом ($w1j$, $w2j$, …, $wnj$). Далее эти значения ($x_1 \cdot w_{1j}$, $x_2 \cdot w_{2j}$, …, $x_n \cdot w_{nj}$) суммируются и передаются функции активации. Функцией активации называется некоторая линейная или нелинейная функция, определяющая значение нейрона. Функцией активации может являться тождественная функция, Единичная ступенька, тангенс, функция распределения Гаусса и другие.

Все нейроны в ИНС можно условно поделить на три группы: нейроны входного слоя, нейроны внутренних слоев и нейроны выходного слоя. Нейроны входного слоя являются первыми в нейронной сети, и их значение задаются условиями решаемой задачи, и не получают на вход сигналов других нейронов (если не используются более сложные архитектуры ИНС).

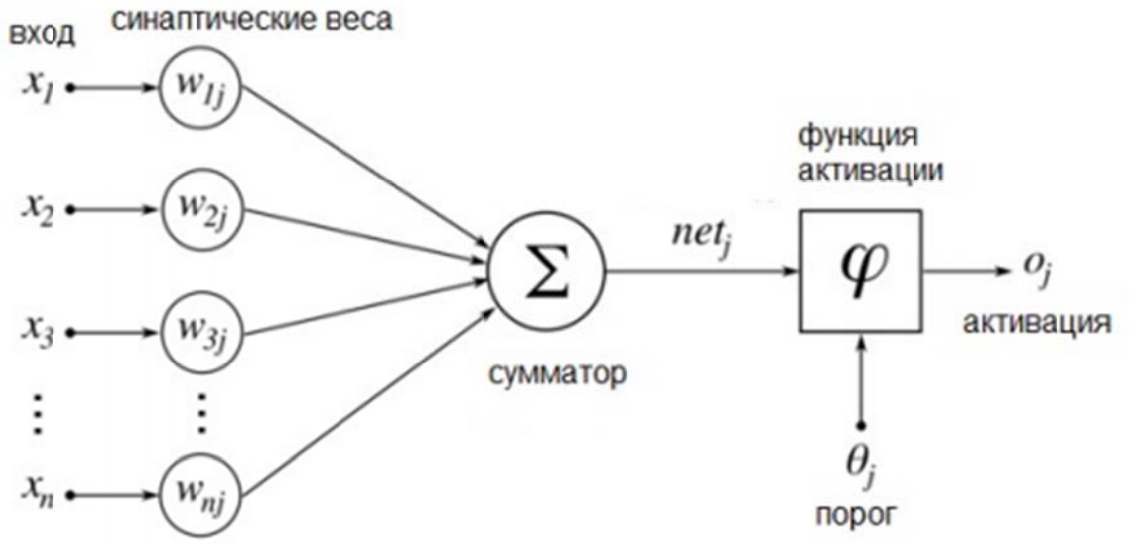



Рисунок 2.13 – Модель искусственного нейрона

Модель простой нейронной сети с одни внутренним слоем приведена на рисунке 2.14. Нейроны внутренних слоев суммируют значения связанных с ними нейронов предыдущего слоя, умноженных на весовые коэффициенты, и с помощью функции активации получают собственное значение, которое передают в следующий слой. Нейроны выходного слоя,

сгенерировав собственное значение, не передают его дальше, а принимаются за ответ задачи и являются прогнозом нейронной сети.

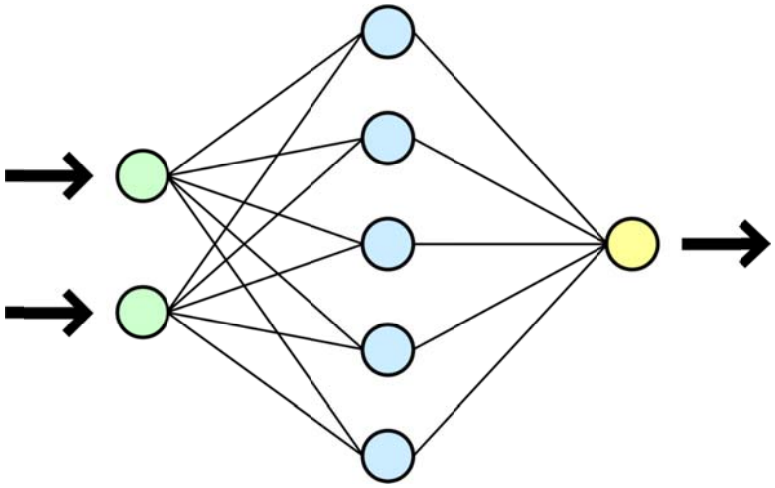


Рисунок 2.14 – Модель нейронной сети с одним внутренним слоем

Так как многослойные нейронные сети могут работать как сложная нелинейная функция, ИНС могут быть использованы там, где возможна реализация непрерывной генерации ключей [1,2]. Идея состоит в возможности синхронизации двух ИНС со случайным начальным набором весовых коэффициентов, и в результате общего переобучения друг под друга общий для обеих систем вектор значений весов, который может быть использован в качестве секретного ключа [3,4].

Поскольку по общественным каналам передаются только входные и выходные значения двух ИНС, то возможная атакующая сторона не в состоянии получить текущие значения их весовых коэффициентов. Такая модель приведена на рисунке 2.15. Опишем пример реализации такой модели.

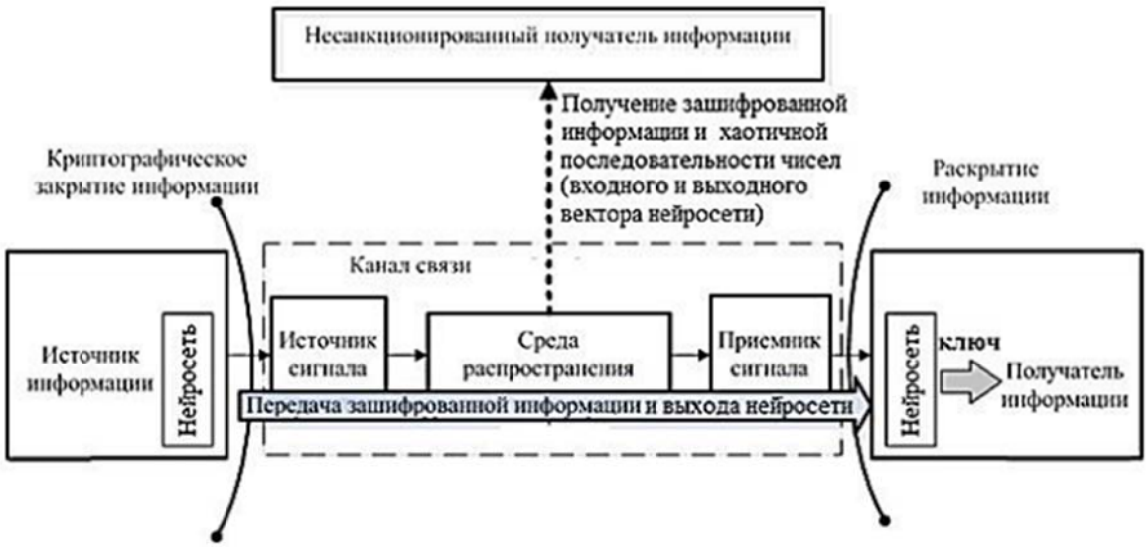



Рисунок 2.15 – Передача ключа с помощью нейронных сетей

Первым шагом в решении этой задачи будет моделирование двух ИНС общей архитектуры. Изменение архитектуры нейронной сети у одной из сторон приведет к большим различиям в конечных, полученных после обучения весовых коэффициентов. На этом же этапе происходит заполнение всех весовых коэффициентов случайными значениями. Так же происходит и с первыми входными векторами данных для каждой нейронной сети.

Далее происходит генерация первых выходных значений, которые и передаются в общем доступе по сети. Эти данные практически бесполезны для злоумышленника, ведь обучение нейронных сетей абонентов происходит друг под друга. Эти выходные данные *i* итерации передаются как входные значения для нейронной сети для генерации новых выходных значений *i*+1 итерации. Во время получения вектора выходных значений ИНС абонента 2, ИНС абонента 1 изменяет собственные веса по определенному правилу. Таким правилом могут быть:

Для обновления весовых коэффициентов могут использоваться следующие правила:

1. Правило Хебба:
$$w_i^+ = w_i + \sigma_i x_i \Theta(\sigma_i \tau) \Theta(\tau^A \tau^B)$$
2. Анти-правило Хебба:
$$w_i^+ = w_i - \sigma_i x_i \Theta(\sigma_i \tau) \Theta(\tau^A \tau^B)$$
3. Случайное блуждание:
$$w_i^+ = w_i + x_i \Theta(\sigma_i \tau) \Theta(\tau^A \tau^B)$$

Этот цикл повторяется до тех пор, пока обе ИНС не будут синхронизированы, т.е. весовые коэффициенты их связей не будут равны. Когда синхронизация двух ИНС будет достигнута, набор весовых коэффициентов может быть использована как секретный ключ в любых симметричных способах шифрования.

Являясь более безопасной версией получения общего секретного ключа, чем алгоритм Диффи-Хелмана, алгоритм с использованием ИНС все же имеет определенные недостатки. Как можно заметить, такой алгоритм получения общего секретного ключа является безопасным только при условии передачи в среде, защищенной от изменения данных нарушителем.

В случае, если проверка подлинности сообщения невозможна, нарушитель может синхронизировать свою нейронную сеть и перехватывать сообщения между абонентами. Для защиты от подобного рода атак в этом алгоритме могут использоваться электронные цифровые подписи для подтверждения целостности полученного сообщения.

Список использованных источников:

1. Воеводин В.В. Математические методы и модели в параллельных процессах. – М.: Наука, 1986.

2. Кобайло А.С. Основы теории синтеза вычислительных структур реального времени. – Минск: БГУИР, 2001.
3. A. Klimov, A. Mityagine. Analysis of Neural Cryptography, 2003 [электронный ресурс]. URL: https://link.springer.com/content/pdf/10.1007%2F3-540-36178-2_18.pdf (дата обращения: 15.10.2020).
4. Shital Daulat Jagtap1, Mr. P. Bala Ramud. Cryptography based on artificial neural network, August 2015 [электронный ресурс]. URL: https://www.ijaiem.org/Volume4-Issue8/IJAIEM-2015-08-03-1.pdf (дата обращения: 15.10.2020).

## 2.9. Использование технологии объемных изображений в системах контроля управления доступом

Системы контроля и управления доступом (СКУД) представляют собой неотъемлемую часть безопасности любого предприятия. В современных реалиях сложно представить свободный доступ на территорию масштабных предприятий и компаний. Каждое подобное сооружение оборудовано системами контроля и управления доступом.

Данная система дает возможность ограничить доступ посторонним, у которых нет прав доступа и проконтролировать перемещение работников предприятия. Но не все компоненты удовлетворяют требованиям по обеспечению защиты от проникновения. Достаточно высока вероятность проникновения злоумышленника на территорию объектов при организации СКУД с помощью атрибутных идентификаторов и поста охраны. Поиску новых решений по аутентификации способствуют и недостатки существующих биометрических считывателей [1].

СКУД заключает в себе такие основополагающие функции как разграничение и контроль прав доступа. Далее мы рассмотрим несколько типовых режимов работы данных систем.

Первым типом работы можно выделить стандартный режим доступа. Когда каждому пользователю дается право прохода через систему доступа с помощью его личного идентификатора, это может быть как устройство в виде карточки или ключа, либо биометрический идентификатор, как отпечаток пальца или непосредственно лицо пользователя. Идентификатор сохраняется в памяти контроллера системы, в котором каждому пользователю присваивается необходимый для его работы уровень доступа [2]. Когда данные есть в системе, человек подносит один из идентификаторов к считывателю контроллера, после чего СКУД принимает решение о допуске или отказе доступа. Каждая попытка входа, удачная или наоборот, записывается в базу данных системы и передается на главный компьютер системы доступа.

Вторым, но не по значимости, выделяется режим запрета на повторный проход через защищенную зону. В этом режиме работы пользователю

запрещается проходить повторно на любую зону, если он предварительно из нее не вышел. Данный режим запрета на повторный проход работает только для дверей, у которых имеется контроль направления прохода. Существует три типа данного режима работы:

- мягкий, когда в базу данных системы будет занесен факт повторного прохода, но система пропустит пользователя;
- временной, когда система, зависимо от проведенного в защищенной зоне времени, будет принимать решение о повторном проходе или выходе из нее;
- строгий, когда система полностью запрещает повторный проход.

Далее можно выделить несколько режимов, которые менее распространены, чем первые два. Такими являются режим двойной идентификации, когда для прохода через контроллеры считывателя понадобится распознавание двух идентификаторов (карты или ключа и биометрического опознавания), закрытый режим прохода, когда доступ через систему к защищенному объекту полностью запрещен и открытый режим прохода, когда через систему доступа проход осуществляется свободно, без предъявления идентификаторов.

В целях повышения эффективности системы для выявления нарушителя рассмотрим возможность внедрения компонента многофакторной аутентификации, который основывается на технологии объемных изображений.

Предпосылками для данного вида аутентификации послужило появление камеры 3D ToF от компании Analog Devices, которая совмещает в себе новые технологии распознавания и из-за новизны аппаратуры еще не использовалась для систем контроля и управления доступом.

Максимально исключить человеческий фактор, а точнее рассеянное внимание, усталость охранника, уменьшая парадокс наблюдателя. Подсистема обеспечивает минимизацию вероятности ложного отказа в доступе зарегистрированному в системе пользователю и вероятности ложного доступа, когда система ошибочно дает право доступа незарегистрированному в системе пользователю.

При реализации статистического метода аутентификации лица математически сложная задача в сравнении с идентификации пользователей по отпечаткам пальцев. Метод предполагает наличие дорогостоящей цифровой, видео и фотоаппаратуры для захвата изображения лица. Достоинствами таких методов является маленький вес файлов идентификационных шаблонов. Лицо человека разделяется на участки, которые не изменяются в зависимости от человека. При вычислении эталонного образца идентификатора любого человека требуется от 1 до 4 десятков участков характерных лицу.

Устройство для захвата изображения устанавливается на расстояние в несколько сантиметров от объекта. После получения изображения, система проводит анализ различных параметром лица, примером является расстояние между глаз и промежуток до носа или размер и форма ушей. Технологии не стоят на месте, подавляющее число алгоритмов может проанализировать лицо пользователя даже если у него есть борода, очки, шапка и другие элементы, которые мешают идентифицировать человека. Сканирование инфракрасными лучами позволяет добиться подобной точности.

В данный момент есть несколько основных методов аутентификации лица, которые различаются по применению и сложностью реализации.

Первым методом является «eigenfaces», в котором для идентификации используется изображение собственного лица. Ранее зарегистрированный шаблон идентификатора сравнивается с лицом человека, определяя коэффициент различия, для которого следует установить подлинность.

Вторым методом выделяют метод анализа «отличительных черт». В этой технологии используются характерные особенности разных областей лица, при котором учитывается не только различия, но и относительное местоположение этих областей. Из-за индивидуальных соединений характеристик лица определяются особенности идентификаторов каждого из пользователей.

Следующим методом является анализ при помощи «нейронных сетей». В этом методе сравниваются совпадения зарегистрированного эталонного образца с проверяемым идентификатором. Используемые алгоритмы устанавливают максимально возможное число соотношений индивидуальных характеристик лица пользователя и параметров эталонного образца.

В течении сопоставления параметров определяются различия между лицом пользователя и эталонным образцом, зарегистрированным в системе, после этого подключается механизм, который при помощи весовых коэффициентов определяет уровень соотношения анализируемого идентификатора человека эталонному образцу из базы данных.

Заключающим методом выделяется автоматический анализ обработки изображения лица. Эта технология является самой простой. Для аутентификации используется расстояние и ее соотношение между самыми легко определяемыми частями лица. Ими являются глаза, губы, уголки рта и нос.

Для решения проблемных ситуаций существующих систем контроля и управления доступом будут использоваться технологии объемных изображений. Последнее время технологии распознавания лиц испытывают глобальные изменения. В данный момент они обширно используются в различных сферах жизни. Они не нуждаются в затратных ресурсах и от-

личаются своей точностью для применения их в системах безопасности. Распознавание лиц применяется для автоматического определения человека при прохождении пропускного пункта. Функция распознавания состоит из процессов захвата изображения лица работника и сопоставление предоставленных данных с уже имеющейся базой изображений пользователей в системе. Из-за того, что для идентификации используются объемные изображения вероятность ошибок первого рода, а также второго рода сводятся к минимуму. Злоумышленник не сможет пройти по фотографии уже зарегистрированного пользователя.

Для увеличения безопасности и сокращения воздействия человеческого фактора предлагается использовать многофакторную аутентификацию. Для решения этой задачи системы на существующем объекте пользователь может возможность выбрать разные режимы идентификации или верификации. Следовательно, обнаруживается возможность подобрать подходящее соотношение значения уровня безопасности и быстродействия. От выбранного режима работы будет зависеть какие сочетания считывателей будут задействованы, будут это биометрические сенсоры, встроенные считыватели карт или сенсорная клавиатура для ввода кода.

*Двумерное распознавание*

Основой технологии двумерного распознавания лиц являются плоские двухмерные изображения. Методы распознавания лиц применяют антропометрические характеристики лица, их образцы и изображения с лицами, которые представляют из себя совокупность физических или же математических признаков. Распознавание двумерных изображений самая используемая и востребованная технология в настоящее время. Благодаря тому, что большая часть баз данных с идентифицированными лицами, которые существуют в мире являются двумерными, а также основное оборудование, которое уже установлено, тоже двумерное. Из этого делаем вывод, что наибольшая часть спроса приходится на двумерные системы распознавания лиц. Плюсом 2D распознавания лиц считается факт существования готовых баз данных образцов и инфраструктуры. Этот сегмент занимает максимум спроса, что станет инициированием разработчиков развивать технологии. Недостатком является то, что коэффициенты ложного пропуска и ложного отказа более высокие чем у трехмерного распознавания лиц.

*Трехмерное распознавание*

Трехмерное распознавание лиц выполняется в большинстве случаев по реконструированным 3D образам. Данная технология распознавания лиц имеет достаточно высококачественные свойства, но не является идеальной [3].

Существует несколько разнообразных 3D сканеров:

1.лазерные сканеры, которые оценивают дальность от сканера до поверхности объекта;

2.сканеры с подсветкой поверхности объекта с последующей математической обработкой изломов полос;

3.сканеры, которые обрабатывают фотограмметрическим способом синхронные стереопары получившихся изображений лиц пользователей.

Face ID, от компании Apple является одним из самых исследованных и знаменитых в кругах потребителей, и экспертов в теме трехмерных сканеров. Фактически это единственное устройство, выпущенное в общий доступ на масс-маркет с трехмерной технологией распознавания лица. Преимуществами 3D технологий являются наибольшая точность результата и уменьшение ошибок по сравнению с описанной ранее двухмерной технологией распознавания лиц. Недостатками данной технологии являются возможность подделки идентификатора с помощью 3D принтера, но пока подобная возможность доступна только для профессионалов, достаточно большая стоимость по сравнению с двухмерными устройствами распознавания.

*Распознавание по текстуре кожи*

Еще одним фактором развития технологий распознавания лица является высокое разрешение изображения. Благодаря этому стал вероятен довольно детальный анализ текстуры кожи. После данного анализа определяется часть кожи лица и преобразуется в изображение, после этого разбивается на маленькие блоки, которые преобразуются в математически измеряемые пространства, на которых записаны линии, текстура кожи и даже поры.

*Распознавание с помощью термографии*

Многообещающим направлением для разработки и распознавания лиц является внедрение термографии, тепловизионных камер и теплового профиля, но полностью скомплектованных и готовых систем для внедрения рентабельных идей пока нет. Это направление является очень перспективным так как почти полностью минимизирует ошибки первого и второго рода [4].

Достоинствами данной технологии является распознавание лиц при недостаточном освещении и абсолютной темноте, даже любой грим, макияж, стрижка, борода, шапка и очки не являются препятствием для тепловизионных камер, в том числе и распознавание близнецов.

*Технология 3D ToF*

Реализация аппаратной стороны подсистемы будет происходить на основе трехмерной времяпролетной камеры. Это вид несканирующего лидара, у которого обнаружение и идентификация объектов с определением расстояния до них происходит с помощью света. В данной камере применяются сильные оптические импульсы, которые продолжаются несколько наносекунд для захвата данных о глубине в рамках интересующей области. Важным составляющим является высокое разрешение для обнаруже-

ния объектов. В разрабатываемой системе будет использоваться камера 3D ToF от компании Analog Devices, которая оснащена датчиком с разрешением 640x480 пикселей, а это гарантирует разрешение в 4 раза выше, чем у многих представленных на рынке подобных датчиков [5].

В добавок ко всему выше сказанному, одним из важным плюсом является тот факт, что вспышки от камеры, используемой в системе, не влияют на здоровье пользователей, в то время как аналогичные системы яркими вспышками света и импульсами значительно вредят здоровью.

Для реализации поставленной задачи можно добавить выявление теплового профиля пользователя. Хотя системы, основанные на использовании термографии лица, не получили распространения, применение этих идентификаторов поможет исключить прохождение на точку доступа злоумышленника даже в профессиональном гриме.

Существует ряд достоинств биометрических систем, которые базируются на принципе распознавания лица по термограмме. Подобные системы нечувствительны к повороту головы, мимике лица, старению, гриму и макияжу, плохому освещению или его отсутствию и даже различает близнецов. В добавок изображение может фиксироваться с расстояния, что идеально подходит для наружного использования. Но существует ряд недостатков, например, потребуются сложные математические вычисления, завышенная стоимость оборудования и лучшие результаты распознавания осуществлялись только в лабораторных условиях [6].

Предлагается использовать подсистему дополнительной аутентификации, в которой контроллером является полноростовой моторизованный роторный турникет. В шлюзе турникета располагается камера 3D ToF от Analog Devices, которая идентифицирует пользователя, подает сигнал серверу и записывает прибытие работника. Перед турникетом находится считыватель бесконтактных карт для того чтобы повторно идентифицировать личность по его карте на расстоянии. Идентификатором для считывателя бесконтактных карт является непосредственно карта, которая записывается в базу данных работников со всей информацией о сотруднике и выдается каждому работнику при заключении договора о принятии на работу, идентификатором для камеры является эталонный образец лица пользователя, который также записывается при принятии на работу или при внедрении подсистемы автоматически. Процедура записи эталонного образца лица должна обновляться каждый год или при каком-либо изменении внешности сотрудника, такие как пластические операции или иные изменения строения характеристик идентификатора, о котором работник сообщает в компанию заранее.

При идентификации используется сервер принятия решения, в котором происходит сопоставление идентификатора пользователя с их эталонным образцом.

Так как в процессе идентификации основным элементом является камера и бесконтактная карта, стандартная аутентификация может происходить на расстоянии, что не замедляет поток людей, благодаря этому пропускная способность системы увеличивается.

Данная система является дополнительным элементом аутентификации для обеспечения максимальной защиты и расширения функций для уже существующей системы контроля и управления доступом на предприятии или другом объекте.

Опишем вариант точки доступа с подсистемой дополнительной аутентификации на базе технологии объемных изображений для обеспечения контроля в точке доступа.

Точка доступа состоит из роторного полноростового турникета, представляющий из себя шлюз, для обеспечения прохода людей без задержки. Турникет используются как прерыватель потока движения сотрудников и для создания ограниченного помещения (шлюза). Длина шлюза совпадает с длинной турникета и составляет около 1,60 метра, при ширине около 0,65 метра.

При помощи контроллера (например, BOLID «С2000-2» в составе АРМ «Орион про») осуществляется управление доступом. Перед полноростовым турникетом со стороны входа устанавливается устройство идентификации для бесконтактных карт, а внутри шлюза стоит камера 3D ToF. Возле турникета с условием хорошей видимости шлюза, находится пост охраны, для контроля прохода. Полноростовой турникет просматривается насквозь и внутри него находится камера, что обеспечивает полный обзор человека, находящегося в шлюзе, и возможность сравнения его с фотографией на мониторе автоматизированного рабочего места.

Оснащение поста охраны программно-аппаратными средствами происходит согласно техническому регламенту по организации АРМ «Орион про» фирмы BOLID: устанавливается пульт управления с двумя кнопками «ВЫХОД», чтобы охранник мог впустить в шлюз человека без предъявления идентификатора, две кнопки «ПОДТВЕРЖДЕНИЕ», чтобы выпустить человека из шлюза, и кнопка «ЗАПРЕТ», для отказа в доступе, монитор для отображения данных о сотруднике и панель с индикаторами, которые оповещают о результатах сканирования системой объемного изображения с помощью камеры 3D ToF .

При сомнительной достоверности идентификации человека, с учетом того, что пришедший человек прошел идентификацию охранником, предусмотрена дополнительная идентификация камерой внутри шлюза, которая запускается автоматически с помощью поверхностного охранного извещателя с формой зоны обнаружения типа «штора». Предусмотрено контактное общение с человеком в шлюзе, при необходимости, через специальное окно в пункте охраны.

Чтобы попасть на территорию объекта сотрудник прикладывает свой идентификатор к атрибутному считывателю перед турникетом. В контроллере формируется сообщение «Идентификация» с указанием уникального кода идентификатора. Если идентификация проходит успешно, то сотрудник может попасть в шлюз турникета и пройти процедуры первичной и дополнительной аутентификации. Первичная аутентификация заключается в том, что после считывания идентификатора сотрудника, его данные выводятся на монитор АРМ и охранник проводит сверку вошедшего с его фотографией на мониторе. Процедура дополнительной аутентификации проводится системой, базирующейся на технологии объемных изображений, состоящей из камеры 3D ToF и охранного извещателя с формой зоны обнаружения типа «штора», находящихся внутри шлюза турникета.

Дополнительная аутентификация заключается в следующем. Система на основе предъявленного сотрудником идентификатора находит в своей памяти его личный файл, в котором хранятся данные его биометрии. Биометрические данные предварительно заносятся в базу данных во время регистрации сотрудника в системе или при начальном этапе внедрения подсистемы, которая в зависимости от частоты прохода через КПП сотрудником, заносит и обновляет эталонный образец каждого работника, при этом система, пока не соберет достаточное количество образцов (всего должно быть 3 прохода через камеру) выводит ошибку «нет идентификатора личности». После полного укомплектования образцов каждого пользователя система перестает выводить данную ошибку. Камера включается с помощью охранного извещателя, который срабатывает после прохождения человека в шлюз турникета. Далее система осуществляет сверку вновь полученной биометрии с зарегистрированными данными. Если сверка прошла успешно на панели АРМ загорается зеленый индикатор, и охранник нажимает кнопку «ПОДТВЕРЖДЕНИЕ» для турникета, формируется сообщение «Доступ предоставлен», и турникет разблокируется (на турникете загорается зеленый индикатор, означающий, что проход разрешен). Теперь сотрудник может пройти через турникет. После преодоления сотрудником турникета в систему заносится сообщение «Проход».

Если сотрудник не проходит идентификацию по бесконтактной карточке, загорается красный индикатор. Охранник должен провести проверку данного сотрудника и принять решение о возможности санкционирования допуска. При запрете допуска охранник нажимает кнопку «ЗАПРЕТ», блокируя турникет.

Есть еще вариант, при котором загорается желтый индикатор. Это значит, что сверка шаблонов вновь полученной биометрии первого или второго типа не прошла сравнение по одному из этих типов эталонных

шаблонов. Решение проводить проверку сотрудника или нет принимает сам охранник.

При необходимости предоставить доступ человеку без идентификатора сотрудник охраны впускает его внутрь шлюза, нажав кнопку «ПОДТВЕРЖДЕНИЕ» для турникета, для выхода из него требуется нажать кнопку «ПОДТВЕРЖДЕНИЕ» уже для выхода из турникета.

При прохождении человеком в шлюз по украденной карточке, он не сможет пройти первичную и дополнительную аутентификацию. В этой ситуации он может покинуть шлюз только назад, когда охранник нажмет кнопку «ПОДТВЕРЖДЕНИЕ» для турникета. Такой посетитель должен быть тщательно досмотрен на посту охраны и при надобности вызвать службу безопасности.

При внедрении данной системы в уже существующую на производстве СКУД предвещается улучшение технических показателей распознавания, рост производительности пропускных мероприятий, минимизация человеческого фактора в идентификации личности пользователей, уменьшение вероятности ложного отказа в доступе зарегистрированному в системе пользователю и вероятности ложного доступа, когда система ошибочно дает право доступа незарегистрированному в системе пользователю.

Список использованных источников:

1. Торокин А. А. Инженерно-техническая защита информации/ Москва. – «Гелиос АРВ». – 2005.
2. Назарецкий С. А. Считыватели "proxy-key" – мультифакторная аутентификация пользователей в системах СКУД/ Системы безопасности, №3. – 2016 [электронный ресурс]. URL: https://bolid.ru/files/411/607/bolid_S.pdf (дата обращения: 20.10.2020).
3. Сидоров. М. Д. Технология объемного изображения результатов глубинного плотностного моделирования геологических структур, 2016 [электронный ресурс]. URL: https://cyberleninka.ru/article/n/tehnologiya-obemnogo-izobrazheniya-rezultatov-glubinnogo-plotnostnogo-modelirovaniya-geologicheskih-struktur/viewer (дата обращения: 20.10.2020).
4. Баша Н. С., Шульга Л. А. Система выделения подкожного кровеносного рисунка по термографическим изображениям/ Вестник МГТУ им. Баумана. – Сер. «Естественные науки». – 2012 [электронный ресурс]. URL: https://cyberleninka.ru/article/n/sistema-vydeleniya-podkozhnogo-krovenosnogo-risunka-po-termograficheskim-izobrazheniyam/viewer (дата обращения: 20.10.2020).
5. Трехмерная времепролетная камера (3D ToF) / Analog Devices [электронный ресурс]. URL: https://www.analog.com/ru/applications/technology/3d-time-of-flight.html (дата обращения: 20.10.2020).
6. Иванеицкий Р. Г. Современное метрическое тепловидение в биомедицине/ Успехи физических наук. – т. 176, № 12. – 2006.

## 2.10. Использование сетей Wi-Fi системах дистанционного мониторинга сотрудников охраны

Контроль физиологического состояния, своевременное определение местоположения сотрудника физической охраны [1] на объекте может способствовать выявлению нападения на сотрудника в процессе обхода территории, в том числе и в случае если после нападения он не в состоянии использовать средства связи.

Осуществление такого контроля возможно с использованием носимого модуля мониторинга состояния, выполненного в виде браслета, который поддерживает связь с объектом наблюдения в помещении и вне здания благодаря точкам доступа, на основе стандартов IEEE 802.11.

Одновременно такая подсистема позволяет организовать дополнительный канал связи для передачи сигнала тревоги.

В процессе функционирования браслета возможно получение следующих физиологических параметров: температура, пульс, оценки артериального давления, сатурации, степень активности, положение относительно горизонта (уровень), пиковые значения ускорений по трем осям [2].

Вариант организации мониторинга внутри здания.

От носимого устройства, находящегося в помещении объекта наблюдения, по протоколу IEEE 802.11 считываемая информация передается на точки доступа расположенные в здании, в количестве достаточном для обеспечения требуемой зоны покрытия.

По протоколу Fast Ethernet, используя витую пару информация передается на коммутатор, и далее на роутер. С роутера сигнал поступает на сервер системы мониторинга сотрудников охраны.

Вариант организации мониторинга вне здания:

От носимого устройства, находящегося в помещении объекта наблюдения, по протоколу IEEE 802.11 считываемая информация передается на точки доступа расположенные вне здания, в количестве достаточном для обеспечения требуемой зоны покрытия.

По протоколу Fast Ethernet, используя витую пару информация передается на коммутатор, далее на роутер. С роутера сигнал поступает на сервер системы мониторинга сотрудников охраны. Местоположение устройства на контролируемой территории определяется с помощью метода триангуляции исходя из принимаемого сигнала точками доступа.

Расстояние между точками доступа не должно превышать двойного радиуса зоны Wi-Fi. С учетом требования перекрытия зон Wi-Fi расстояние между двумя точками доступа не должно превышать 75 метров, так как радиус покрытия беспроводным сигналом одной точки доступа, как правило, не превышает 50 метров. Для повышения точности определения

местоположения, может быть использован анализ уровня сигнала с носимого устройства принимаемой точкой доступа, с учетом затухания сигнала в свободном пространстве. Точкам доступа необходимо выделять различные каналы Wi-Fi для исключения взаимных помех при работе.

Список использованных источников:

1. Торокин А. А. Инженерно-техническая защита информации/ Москва. – «Гелиос АРВ». – 2005.
2. Цифровые медицинские платформы сбора данных. Обзор, 29.07.2019 [электронный ресурс]. URL: https://evercare.ru/tsifrovye-platformy-dlya-sbora-meditsinskikh-danny (дата обращения: 20.10.2020).

## 2.11. Использование программно-аппаратных решений НВП «Болид» в системах дистанционного мониторинга и предотвращения аварийных ситуаций в инженерных сетях

Случайные воздействия, вызванные аварийными ситуациями при эксплуатации инженерных сетей (водоснабжение и центральное отопление), могут быть причиной потери информации или стойкого блокирования доступа к ней за счет повреждения оборудования на объектах информатизации [1].

Снижению риска возникновения таких угроз могут способствовать системы дистанционного мониторинга и выявления аварийных состояний, обладая возможностью выявления аварийных ситуаций с возможностью удаленного управления основными техническими узлами [2].

Определим основные требования, которые предъявляются к дистанционной системе мониторинга в целом. Типовая система дистанционного мониторинга должна обеспечивать:

- незамедлительное оповещение при аварийной ситуации на объекте;
- предоставление по запросу пользователя или диспетчера полной информации о показаниях и состоянии оборудования объекта;
- отображение контролируемых объектов в удобной для восприятия форме;
- возможность в оперативной форме внести изменения в работу оборудования объекта при возникновении аварийных ситуаций;
- осуществление контроля успешности выполнения команд управления и оповещение в форме сигнала тревоги при невозможности их выполнения;
- возможность анализа работы отдельных объектов системы или группы объектов по выбранным параметрам и характеристикам за указный период времени;

- возможность дистанционного осуществления настройки контроллеров системы и выполнения их диагностики;
- возможность составления и ведения отчетов (аварийных ситуаций, действий оператора).

Аппаратная часть системы дистанционного мониторинга, установленная на объекте, должна иметь возможность вариативно конфигурировать настройки в зависимости от технических особенностей объекта. Базой для аппаратной части данных систем являются контроллеры, каждый из которых должен иметь возможность подключать:

- цифровые или импульсные счетчики для контроля температуры, давления, уровня и т.п.;
- измерительные приборы на базе стандартных интерфейсов и протоколов связи;
- преобразователи интерфейсов для сопряжения с устройствами хранения и обработки информации.

В целях обеспечения высокой стабильности и корректности передачи данных со счетчиков, а также обеспечения защиты от несанкционированного изменения показаний потребления коммунальных ресурсов, данная система построена на базе технологии проводной связи.

В сфере систем автоматизации на данный момент существует три наиболее распространенных интерфейса: RS-232, RS-485, CAN. Исходя из рассмотренных выше интерфейсов передачи данных, для реализации разрабатываемой системы лучше всего подойдет RS-485, причиной этому служат, большая длина линий связи, высокая помехоустойчивость, а также тот факт, что подавляющее большинство контролеров и датчиков поддерживают данный интерфейс. Возможности сети, построенной на стандарте CAN, являются излишними для системы данного типа. А короткая длина линий связи и слабая помехоустойчивость ставят стандарт RS-232 в менее приоритетные положения относительно RS-485.

Необходимо также отметить, что интерфейс RS-485 имеет две вариации исполнения: двухпроводная и четырех проводная. Делается этого для организации полудуплексной и полнодуплексной связи соответственно. При использовании двухпроводного варианта интерфейса информация единовременно может передаваться только в одном направлении, а в четырехпроводной вариации два провода передают информацию в одну сторону, а два других в обратную, что обеспечивает полнодуплексную связь.

В случаи разработки дистанционной системы мониторинга объектов организация полнодуплексной связи по средству использования четырехпроводной схемы организации интерфейса RS-485 не является необходимостью, а только увеличит стоимость проекта и повысит сложность его реализации, так как при организации полнодуплексного режима передачи

данных существует требование жесткого указания на стадии проектирования ведущих и ведомых устройств.

Интерфейс RS-485 может иметь максимальную длину кабеля 1200 метров при максимально скорости передачи на такой длине кабеля до 100 Кбит/с. В данной системе имеет смысл использовать экранированную витую пару, для достижения максимальной помехоустойчивости.

Разрабатываемая система дистанционного мониторинга и выявления аварийных состояний объектов может быть развернута на базе сертифицированных счетчиков и контролеров компании «Болид», прошедшие все необходимые испытания для введения в эксплуатацию.

Для объединения конечных устройств и контроллеров в единую цепь могут использоваться два варианта организации сети – это двухпроводная линия связи (ДПЛС), с главным контролером двухпроводной линии связи с гальванической изоляцией С-2000-КДЛ-2И [3] и сеть, построенная на базе витой пары проводов под управлением интерфейса RS-485.

Ключевые особенностями контролера двухпроводной линии связи являются:

- возможность подключения до 127 адресных устройств;
- длина двухпроводной линии 600-700 метров;
- наличие коммуникационного порта – RS-485;
- рабочий диапазон температур от -30 до +55 градусов Цельсия;
- возможность работы с широким рядом различных датчиков и счетчиков;
- контроль вскрытия корпуса блока.

Исходя из характеристик данного контролера, можно отметит, что длины двухпроводной линии связи в 600-700 метров хватит для соединения конечных устройств в рамках одного строения, а также возможность поддержки 127 адресных устройств достаточна для подключения счетчиков водоснабжения и датчиков затопления, также в рамках одного строения.

При использовании данной системы в строениях с большим количеством помещений, где, соответственно, необходимо задействовать большое количество конечных устройств существует возможность установки нескольких контролеров С-2000 КДЛ-2И.

Рабочий диапазон температур подходит для климатических условий средней полосы России. Возможность подключения считывателей ключей, а также контроль вскрытия корпуса блока обеспечивают безопасность эксплуатации.

Технические средства системы на основе цифрового интерфейса RS-485 могут объединяться в сеть с топологией «шина». Особенностью сети, построенной на стандарте RS-485 по «витой паре» проводов, является тот факт, что при больших расстояниях и наличий ответвлений сети на боль-

шой скорости передачи данных появляются искажения сигнала, что может повлиять на целостность данных при их передаче. В качестве меры минимизации данного эффекта в данной системе используется относительно низкая скорость передачи данных около 9600 бит/с, что в конечном итоге не как не существенно повлияет на выполнение основной функции системы – сбор данных с адресных устройств.

Данная сеть является централизованной, т.е. одно устройство в сети генерирует запросы и команды для остальных устройств, в случаи рассматриваемой системы, С2000-Ethernet выполняет роль контроллера сети и выполняет удаленный опрос конечных устройств, с целью получения показаний датчиков и расходометров приборов учета.

Передача данных с адресных устройств в системе происходит по сети RS-485, и через преобразователь интерфейсов С2000-Ethernet подключается к АРМ диспетчера.

Электропитание сети RS-485 производится от резервного источника питания РИП-12 (12 В) подключенного в электрическую сеть с напряжением 220В.

Необходимо отметить, что при нехватке длины линии RS-485 для подключения всех приборов, существует возможность интеграции в систему повторителя интерфейса «С2000-ПИ», за счет использования которого, можно делать ответвления от основной магистрали RS-485 для построения топологии «звезда» и увеличения длинны линии на максимум на 1500 метров. Но стоит отметить, что ответвления не желательны, так как увеличивают искажение сигнала в линии.

Рассматривая вариант, когда пункт управления диспетчера находится в зоне покрытия сети RS-485, тогда модуль удаленного ввода вывода напрямую подключается к преобразователю интерфейсов С2000-Ethernet [4]. Далее по линии связи Ethernet данные со счетчиков передаются на компютер диспетчера (автоматизированное рабочее место), где происходит контроль адресных устройств, программирование сценариев выявления и устранения аварийных ситуаций, а также запись показателей приборов учета в базу данных.

В случае, когда пункт диспетчера расположен удаленно от объекта, может применяться медиаконверетер «Planet FT-806», который преобразует среду распростанения сигнала из Ethernet в отповолоконную среду передачи данных. Данный медиаконвертер работает как полудуплексном, так и полнодуплексном режиме в зависимости от используемой сети. При использовании стандарта 100Base-FX скорость передачи данных до 100Мбит/с при расстоянии до 20км, что позволяет подключится к удаленному диспетчерскому пункту.

Опишем основные методы выявления и устранения типовых аварийных ситуаций на объекте.

*Протечка труб отопления с горячей водой*

Причиной прорыва труб отопления может быть несколько факторов:

- плохое состояние коммуникации;
- неисправное состояние приборов отопления;
- неправильное подключение или установка приборов отопления;
- брак в оборудовании или самих трубах;
- гидравлический удар.

При этом существует два места локализации:

- общедомовая авария (прорыв или протечка труб отопления в подвале и подсобных помещениях);
- по отдельным помещениям или их группам (неисправность отопительных приборов и труб).

В данной системе для выявления данного вида аварии применяется пара общедомовых теплосчетчиков, установленных в месте подвода отопительных труб к дому, на прямой и обратный трубопровод, а также шаровый кран с электроприводом.

Алгоритм работы подсистемы построен на принципе законов электротехники, в частности первом законе Кирхгофа, одна из формулировок которого гласит, что сумма токов, втекающих в узел, равна сумме токов, вытекающих из узла. Только в рассматриваемой системе, вместо суммы токов используется мгновенный объемный расход теплоносителя (Q).

Следовательно, в данном случае, правило основанное на применении данной закона, звучит следующим образом, мгновенный объемный расход теплоносителя поступающий на объект равен мгновенному объемному расходу теплоносителя, возвращающего обратно.

Ситуация, при которой показание мгновенного объемного расхода теплоносителя общедомового теплосчетчика, установленного на обратном трубопроводе отопления, уменьшается относительно показателя на счетчике прямой подачи, свидетельствует о потере теплоносителя в системе, то есть протечке. На практике уменьшение объемного расхода теплоносителя, более чем на 0,1 м3/ч в течении половины часа указывает на наличие порыва (минимальное значение при котором система определяет появившуюся разницу объемного расхода зависит от конструктивных особенностей объекта, в данном случае взято условно среднее значение).

Переводя данное правило на программный уровень можно получить следующий алгоритм:

1. общедомовые счетчики тепла передают по запросу данные об измеряемых параметрах на компьютер-сервер;

2. полученные показатели приборов записываются в базу данных, спроектированную ранее в этой работе;

3. на основе выполнения запроса к базе данных, выполняется анализ показателей мгновенного объемного расхода общедомовых теплосчетчиков, установленных на прямой и обратный трубопровод теплоснабжения.

Суть анализа заключается в проверке условия:

Qп – Qо > 0,1 (м3/ч),

где Qп – мгновенный объемный расход теплоносителя (м3/ч) в подающем трубопроводе,

Qо – мгновенный объемный расход теплоносителя (м3/ч) в обратном трубопроводе.

4.Система предоставляет диспетчеру информацию о возникновении аварийной ситуации на объекте. Диспетчер на основе полученных данных принимает решение о перекрытии трубопровода отопительной системы. Перекрытие производится при помощи передачи сигнала на шаровой кран с электроприводом и подключенного к системе через интерфейс RS-485.

Построенные в данной работе логические принципы реализации выявления аварийных состояний объектов, на практике развертываются на базе программного продукта АРМ «Орион» компании НВП «Болид» в рамках функционала «Оперативной задачи». При дальнейшей реализации материала работы в будущих проектах существует возможность создания собственного средства реализации базовой логики системы.

Данный программный комплекс используется в системе также в качестве сопряжения каналов связи между приборами НВП «Болид».

*Затопление в помещении*

Одним из самых распространенных случаев аварийной ситуации в помещении является «потоп», который может повлечь существенный материальный ущерб исходя из ряда прямых и косвенных последствий данной аварии.

В разрабатываемой системе предусмотрен механизм выявления аварийных ситуаций данного типа, это осуществляется за счет датчика затопления С2000-ДЗ и шарового крана установленного на стояке водоснабжения.

Принцип работы заключается в следующем, датчик затопления, установленный на полу под трубами водоснабжения или в другом месте, где затопление наиболее вероятно и подключен к ДПЛС.

При обнаружении затопления С2000-ДЗ формирует адресный сигнал тревоги и передает по двухпроводной линии связи контроллеру «С2000-КДЛ-2И», сопряжение каналов связи приборов выполняется на базе программного продукта АРМ «Орион».

После этого на интерфейсе системе формируется сообщение об обнаружении аварийной ситуации. Диспетчер на основании полученной информации с адресного устройства принимает решение о перекрытии «стояка» водоснабжения, при помощи отправки адресного сигнала на шаро-

вый кран с электроприводом, для перекрытия подачи воды и остановки не контролируемой утечки водного ресурса (возможного прорыва труб водоснабжения).

Список использованных источников:

1. Торокин А. А. Инженерно-техническая защита информации: учеб. пособие для студентов, обучающихся по специальностям в области информационной безопасности. – М.: Гелиос АРВ, 2005. – 960 с.
2. Безопасность в чрезвычайных ситуациях. Структурированная система мониторинга и управления инженерными системами зданий и сооружений. Общие требования. ГОСТ Р 22.1.12-2005 – М.: Стандартинформ, 2005. – 29 с.
3. Контроллера двухпроводной линии связи С2000-КДЛ-2И: руководство по эксплуатации [электронный ресурс]. URL: https://bolid.ru/files/373/566/s2000_kdl_2i_ret_v.1.26_apr_19.pdf (дата обращения: 20.10.2020).
4. Преобразователь интерфейсов RS-485/RS-232 в Ethernet «С2000-Ethernet»: руководство по эксплуатации [электронный ресурс]. URL: https://bolid.ru/files/373/566/s2000_Ethernet_rep_nov_19.pdf (дата обращения: 20.10.2020).

# ГЛАВА 3. МЕТОДИЧЕСКИЕ АСПЕКТЫ ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ ОРГАНИЗАЦИЙ

## 3.1. Алгоритм (этапы) создания системы обработки и обеспечения безопасности персональных данных в органе власти, организации

В последнее время существенно выросло количество преступлений в сфере информационных технологий. Причем значительная часть из них связана с неправомерным использованием персональных данных. В связи с этим обеспечению безопасности персональных данных придается особое внимание. В развитие ФЗ от 27.07.2006 года № 152 «О персональных данных» (в ред. ФЗ от 25.07.2011 N 261) были приняты ФЗ от 07.02 2017 года №13, Постановления Правительства РФ от 15.09. 2008 г. N 687, от 01.11.2012 № 1119, от 21 марта 2012 г. N 211, а также Приказ ФСТЭК России от 18.03.2013 года №21. В этих документах определены требования, в первую очередь, к операторам по соблюдению правил обработки персональных данных и обеспечению их безопасности. Для многих операторов решение проблемы по созданию системы обработки персональных данных и обеспечению их безопасности, удовлетворяющей требованиям указанных документов, вызывают определенные трудности. С целью решения этой проблемы и оказания методической помощи операторам автор на основе многолетнего опыта работы в сфере информационной безопасности разработал и предлагает «Алгоритм создания системы обработки и обеспечения безопасности персональных данных в организации». Алгоритм можно условно разбить на этапы.

**1 этап. Организационные мероприятия**

1.1 Назначение сотрудника (ов) из состава руководителей, ответственного (ых) за организацию мероприятий по обработке и обеспечению безопасности персональных данных.

1.2. Назначение ответственных за выполнение мероприятий по обработке и обеспечению безопасности персональных данных (подразделения, сотрудники – специалисты по защите информации, администратор безопасности информации и пр.), определив при этом их функции, обязанности, права и ответственность в положениях о подразделениях и в должностных инструкциях сотрудников.

1.3 Назначение комиссии по оценке соответствия (аттестации) объектов информатизации требованиям безопасности информации, по классификации автоматизированных систем и установке уровней защищенности персональных данных.

1.4 Разработка организационно-распорядительных документов, определяющих содержание и порядок обработки и обеспечения безопас-

ности персональных данных в организации, а именно: «Положение об обработке и обеспечении безопасности персональных данных в организации».

Этим документом определяются правила (политика) обработки персональных данных, устанавливающие процедуры, направленные на выявление и предотвращение нарушений законодательства Российской Федерации в сфере персональных данных, а также определяющие для каждой цели обработки персональных данных содержание обрабатываемых персональных данных, категории субъектов, персональные данные которых обрабатываются, сроки их обработки и хранения, порядок уничтожения при достижении целей обработки или при наступлении иных законных оснований.

В указанном Положении кроме правил обработки персональных данных могут быть представлены:

- комплекс мероприятий по обеспечению безопасности персональных данных;
- правила рассмотрения запросов субъектов персональных данных или их представителей;
- правила осуществления внутреннего контроля соответствия обработки персональных данных требованиям к защите персональных данных, установленным Федеральным законом «О персональных данных», принятыми в соответствии с ним нормативными правовыми актами и организационно-распорядительными документами;
- правила работы с обезличенными данными;
- перечень информационных систем персональных данных;
- перечень персональных данных, обрабатываемых в организации;
- перечень должностей работников организации, ответственных за проведение мероприятий по обезличиванию обрабатываемых персональных данных;
- перечень должностей работников организации, замещение которых предусматривает осуществление обработки персональных данных либо осуществление доступа к персональным данным.

Кроме этого к указанному Положению целесообразно в качестве приложений разработать ряд перечней, инструкций и типовые формы различных документов.

- Перечень (состав) персональных данных, обрабатываемых в организации.
- Список должностных лиц, ответственных за организацию и выполнение мероприятий по обработке и обеспечению безопасности персональных данных.
- Список должностных лиц, допущенных к обработке персональных данных.

- Список должностных лиц, обеспечивающих безопасность персональных данных.
- Список должностных лиц, обслуживающих ИСПДн.
- Должностные инструкции сотрудников, ответственных за организацию обработки и обеспечение безопасности персональных данных.
- Должностные инструкции сотрудников, ответственных за выполнение мероприятий по обработке и обеспечению безопасности персональных данных, допущенных к обработке персональных данных обслуживающих ИСПДн.
- Перечень технических средств, участвующих в обработке персональных данных, помещений, в которых они установлены, либо хранятся материальные носители персональных данных, и лиц, допущенных в эти помещения
- Порядок доступа работников в помещения, в которых ведется обработка персональных данных;

Типовые формы:

- типовое обязательство сотрудников, непосредственно осуществляющих обработку персональных данных, в случае расторжения с ними контрактов прекратить обработку персональных данных, а также о неразглашении персональных данных ставших известными им в связи с исполнением должностных обязанностей;
- типовая форма согласия субъектов персональных данных на обработку его персональных данных;
- типовая форма согласия субъекта на получение его персональных данных у третьей стороны;
- типовая форма согласия субъекта на передачу его персональных данных третьей стороне;
- типовая форма запроса о предоставлении персональных данных работников предприятия (личных дел) для ознакомления;
- типовая форма разъяснения субъекту персональных данных юридических последствий отказа предоставить свои персональные данные;
- типовая форма заявления об отзыве согласия на обработку.

**2 этап. Создание ИСПДн в защищенном исполнении**

2.1 Разработка частной модели угроз безопасности персональных данных.

2.2 Разработка модели нарушителя.

2.3 Определение (установка) границ контролируемой зоны.

2.4 Определение (установка) уровня защищенности персональных данных.

2.5 Определение (установка) класса защищенности ИСПДн как АС.

2.6 Разработка технического задания на создание ИСПДн в соответствии с ГОСТ 34602.

2.7 Разработка технического задания на создание системы защиты информации в ИСПДн в соответствии с ГОСТ 51583.

2.8 Разработка проекта на создание ИСПДн в защищенном исполнении.

В состав проектной документации, как правило, входят:

- пояснительная записка;
- структурная схема электрическая;
- эксплуатационно-техническая документация.

В пояснительной записке излагаются сведения о:

- конфигурации и топологии ИСПДн в целом и ее отдельных компонентов;
- технических средствах и системах, предполагаемых к использованию в составе ИСПДн, в том числе средствах защиты информации, условиях их расположения;
- общесистемных и прикладных средствах программного обеспечения, предполагаемых к использованию в ИСПДн;
- технологии (режимах) обработки персональных данных в ИСПДн в целом и в ее отдельных компонентах;
- условиях расположения ИСПДн относительно ограждающих конструкций и границ контролируемой зоны.

На структурной схеме отображаются элементы ИСПДн – АРМы и соединяющие их линии относительно ограждающих конструкций помещений, в которых они установлены, и границ контролируемой зоны. В спецификации к структурной схеме могут указываться типы ОВТ, типы соединительных проводов, типы разъемов и т.д.

В состав эксплуатационно-технической документации, как правило, входят:

- инструкции по эксплуатации ИСПДн и АРМов, входящих в ее состав;
- руководство пользователю;
- инструкция системному администратору;
- инструкция администратору безопасности информации;
- инструкция по соблюдению правил техники безопасности (электро-, пожарной и т.д.);
- инструкция пользователю по соблюдению режима защиты информации при работе на АРМ и в ЛВС;
- инструкция пользователю о порядке отнесения информационных ресурсов Предприятия к защищаемым и об организации доступа к ним;

- инструкция пользователю в случае возникновения нештатных ситуаций;
- инструкция пользователям АРМ по организации антивирусной защиты;
- инструкция пользователям АРМ по организации парольной защиты;
- инструкция пользователю АРМ услугами электронной почты и internet
- инструкция пользователю АРМ по использованию и хранению электронных носителей с информацией ограниченного доступа в подразделениях;
- инструкция по внесению изменений в списки пользователей АРМ и наделению их полномочиями доступа к информационным ресурсам Предприятия;

а также ряд иных, в случае необходимости, инструкций.

2.9 Реализация проекта, а именно:

- закупка оборудования (СВТ, лицензионных программных продуктов, сертифицированных средств защиты информации (СЗИ), соединительных проводов, комплектующих изделий и т.д.)
- установка оборудования, прокладка кабелей и соединение элементов ИСПДн между собой, установка программных продуктов;
- настройка и отладка работы элементов и ИСПДн в целом;
- испытания и прием ИСПДн в эксплуатацию.

**3 этап. Оценка эффективности реализованных мер защиты информации в ИСПДн (аттестация)**

Формирование Акта по результатам оценки эффективности реализованных мер защиты информации (Аттестата соответствия) с приложениями:

- акт установки уровня защищенности персональных данных;
- акт классификации АС (АРМ) в составе ИСПДн;
- предписание на эксплуатацию объекта информатизации – ИСПДн;
- схема (карта с границами) контролируемой зоны;
- протокол измерения параметров заземления АРМ;
- описание технологического процесса обработки информации (каким образом обрабатывается, что и где хранится, куда выдается и т.д.);
- технический паспорт объекта информатизации – ИСПДн;
- акт установки и настройки средств и систем защиты информации;
- матрица разграничения доступа к информационным ресурсам ИСПДн;
- состав программного обеспечения, установленного в ИСПДн;

- программа и методика аттестационных испытаний;
- сертификаты (копии) на установленные СЗИ (программные и технические);
- список лиц, имеющих доступ в помещение;
- заключение по результатам оценки эффективности (аттестационных испытаний).

Выполнение отдельных мероприятий алгоритма (этапов) может проводиться одновременно, в зависимости от количества привлекаемых сотрудников и уровня их квалификации.

Реализация рассмотренного алгоритма позволит создать адекватную требованиям систему обработки и обеспечения безопасности персональных данных в организации.

Список использованных источников:

1. Федеральный закон «О персональных данных» от 27.07.2006 № 152 (в ред. ФЗ от 25.07.2011 N 261).
2. Постановление Правительства Российской Федерации от 01 ноября 2012г. № 1119 «Об утверждении требований к защите персональных данных при их обработке в информационных системах персональных данных».
3. Постановление Правительства Российской Федерации от 21 марта 2012 г. N 211 «Об утверждении перечня мер, направленных на обеспечение выполнения обязанностей, предусмотренных ФЗ «О персональных данных» и принятыми в соответствии с ним нормативными правовыми актами, операторами, являющимися государственными или муниципальными органами».
4. Приказ ФСТЭК России от 18 февраля 2013 г. № 21 «Об утверждении Состава и содержания организационных и технических мер по обеспечению безопасности персональных данных при их обработке в информационных системах персональных данных».
5. ГОСТ 34.602–89. Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы.
6. ГОСТ Р 51583–2014. Защита информации. Порядок создания автоматизированных систем в защищенном исполнении. Общие положения.

## 3.2. Методические аспекты внедрения GDPR в организации

Новые нормы общего регламента по защите данных (General Data Privacy Regulations, GDPR), принятого взамен директивы № 95/46/EC о защите прав частных лиц при обработке персональных данных, вступили в действие 25 мая 2018 г. и применяются ко всем компаниям европейского рынка, включая иностранные организации.

Целью GDPR является оказание помощи пользователю в понимании использования его персональных данных и управления доступа к ним.

Персональными данными является любая информация, относящаяся к идентифицированному или идентифицируемому физическому лицу (субъект данных), с помощью которой прямо или косвенно его можно определить (п. 1 ст. 4 [1]). Примерами такой информации может быть логин и пароль, интернет-ID, IP-адрес, а также личные данные, которые пользователь указывает в своих аккаунтах (например, пол, возраст) и данные, которые сайты собирают для рекламных целей (поисковые запросы, геолокация и т.д.).

*Внедрение GDPR в организации*

GDPR применяется экстерриториально, т.е. не имеет значения, где находится компания. Если в ней обрабатываются и хранятся данные, относящиеся к резидентам Европейского союза (ЕС), следовательно, на нее будет распространятся действие GDPR.

Организация обязана назначить себе представителя в ЕС при выполнении хотя бы одного из нижеперечисленных условий:

- обработка персональных данных происходит постоянно;
- специальные категории персональных данных обрабатываются в крупных размерах;
- производится обработка персональных данных, связанных с судимостями или преступлениями;
- имеет место высокий риск нарушения прав и свобод человека [2].

В разрезе российских компаний введение норм регулирования GDPR в первую очередь касается организаций энергетической, финансовой, транспортной и ИТ-сфер.

Работа по внедрению GDPR для компании или группы компаний происходит поэтапно:

1 этап. Проверка применимости GDPR.

2 этап. Анализ систем обработки персональных данных и их защиты в организации.

2.1. Изучение структуры компании.

2.2. Анализ договоров с компаниями, относящимся к резидентам ЕС.

2.3. Изучение технических инструментов для обработки и защиты персональных данных.

3 этап. Проверка соответствия требованиям GDPR.

3.1. Оценка несоответствия обработки персональных данных.

3.2. Определение перечня мероприятий по приведению деятельности компании в соответствие с требованиями GDPR.

4 этап. Реализация мероприятий по приведению деятельности компании в соответствие с требованиями GDPR.

5 этап. Консультационная или абонентская поддержка.

При внедрении GDPR в организации необходимо следовать принципам, которые составляют основу общего регламента по защите данных (таблица 3.1).

Таблица 3.1 – Принципы GDPR

| Принципы | Характеристика |
|---|---|
| Легитимность, справедливость, прозрачность | информация о целях, методах и объемах обработки персональных данных требует более доступного и простого изложения. |
| Ограничение цели | закрепление целей использования персональных данных в Политике конфиденциальности и их соблюдение. |
| Минимизация данных | количество собираемых данных должно совпадать с целями компании, заявленными в Политике конфиденциальности. |
| Конкретность | по требованию пользователя личные данные, которые являются неточными, должны быть обновлены или удалены. |
| Ограничение хранения | данные удаляются сразу же после использования. |
| Целостность и конфиденциальность | защита данных от несанкционированного доступа, незаконной обработки или повреждения. |

Следует отметить, что основным документом GDPR является политика конфиденциальности. В идеале это один документ, который содержит основные цели обработки персональных данных и написан простым, не юридическим языком. Из политики конфиденциальности должно быть понятно, как будут обрабатываться пользовательские данные:

- какие пользовательские данные вы собираете;
- кто собирает, каким образом и на каком основании;
- как данные используется в работе;
- будет ли они переданы третьим лицам;
- как долго данные хранятся у вас;
- может ли пользователь управлять своими данными;
- могут ли быть негативные последствия для пользователя.

Внедрение GDPR диктует необходимость реорганизации бизнес-процессов и штатной структуры внутри компании клиента.

С целью определения основных направлений, которые необходимо доработать в организации, следует провести экспресс-анализ уровня соответствия регламенту GDPR в рамках SAM проекта (Software Asset Management). Такой анализ выполняется специализированными компаниями. Реализация SAM-проекта позволяет повысить эффективность работы IT-инфраструктуры за счет проведения аудита процессов доступа и использования персональных данных; разработки рекомендаций по мерам

безопасности, направленных на предотвращение, выявление и реагирование на угрозы, связанные с уязвимостями и утечкой конфиденциальных данных; разработки практических рекомендаций по управлению данными [3,4].

В целом, по оценке экспертов, компании нужно около 40 документов для того, чтобы вести свою работу в соответствии с положениями GDPR. Большинство из них являются внутренними документами, и лишь незначительная их часть должна размещаться на сайте. Эти документы используются для того, чтобы доказать регуляторным органам подотчетность компании, показать, что она ведет свою работу по GDPR. При этом важно прописать не только политику приватности, которая излагается в публичный доступ на сайте, но внутренние документы о роли и ответственность при обработке данных внутри компании.

Чтобы избежать штрафов и потери клиентов, необходимо подготовить «минимальный пакет» документов:

- Privacy Notice (уведомления об обработке персональных данных), его обычно объединяют с Privacy Policy;
- Privacy Policy (общая информация по обработке персональных данных, отношение к обработке данных несовершеннолетних и информация по обработке специальных категорий персональных данных, в том числе биометрических данных). Она составляется в доступной форме на языках тех стран, с которыми сотрудничает организация;
- Cookie Policy (политика в отношении сбора cookie-файлов, их обработки и целей использования);
- политику защиты персональных данных;
- согласия на обработку персональных данных (желательно отдельное согласие для каждой цели обработки cookie);
- дисклеймер об обработке Cookie (письменный отказ от ответственности перед предоставлением той или иной услуги) [5].

GDPR не регламентирует язык, на котором должно быть написано соглашение об обработке данных (DPA, Data Processing Agreement), поэтому будет достаточно прописать соглашение на английском языке. При этом к документу предъявляется ряд требований: об обеспечении безопасности персональных данных, возможности проведения внешнего аудита для проверки исполнения документа, предупреждении контроллера о смене субпроцессоров (ваших подрядчиков при исполнении основного договора) и уведомлении об инцидентах.

Следует отметить, что преимуществом GDPR является отсутствие необходимости контроллерам отправлять уведомления об операциях в каждый локальный орган, ответственный за защиту персональных данных, в которых зачастую были свои требования к оформлению. Теперь же,

за некоторым исключением, достаточно типового договора и регламента учета видов деятельности по обработке данных.

Рассмотрим основные норма регулирования GDPR.

*Нормы защиты персональных данных в рамках GDPR*

Нормы GDPR рассматривают все данные с персонифицированной информацией, то есть сведениям об организациях и физических лицах. Речь идет, в частности о маркетинговых данных, при хранении которых большинство компаний относятся менее требовательно, чем к информации, например, финансового или медицинского характера.

Приведение в соответствии с регламентом GDPR позволяет обрабатывать персональные данные, связанные с предложением товаров или услуг гражданам ЕС (независимо от того, требуется ли оплата); проводить мониторинг поведения пользователей в ЕС (маркетинг, отслеживание в соцсетях и т.д.).

Кроме того, нельзя пользоваться длинными неудобочитаемыми пользовательскими соглашения. Они должны быть для пользователя представлены формами соглашений на простом языке.

При чем пользователь имеет возможность быстро отозвать свое согласие и обязать компанию удалить его персональные данные. Это право, названное Data Erasure, дает возможность прекращения распространения и обработки персональных данных, а также обязывает их убрать. Проверка таких запросов дает возможность регуляторам в качестве оценки рассматривать «общественный интерес к доступности информации», то есть право Data Erasure может быть аннулировано, если, например, свои данные пытается скрыть злоумышленник.

В качестве примера можно рассмотреть следующее. Провайдер облачных услуг предоставляет вычислительные услуги в облачной среде, поэтому он действует от имени клиента и выступает в роли процессора. В данном случае клиент является контроллером, так как принимает решение об использовании того или иного облачного сервиса для обработки персональных данных. Контроллер отвечает за защиту данных. В отдельных случаях провайдер облачных услуг (процессор) может быть и контроллером [6].

Компании, находящиеся за пределами ЕС, должны иметь, в случае необходимости, представителя для работы с европейскими регуляторами. Также следует отметить, что в соответствии с регламентом GDPR, регуляторы наделяются правом затребовать информацию о том, как, где и с какой целью используются персональные данные.

Контроллеры предоставляют субъекту данных копию информации в электронном формате – это радикальный переход к прозрачности информации, хотя он и влечет за собой очень много проблем. Следует отметить, что передача любых данных по требованию имеет риск утечки. В ряде

случаев компании не изъявляют желания, чтобы пользователи получали данные о том, какие сведения о них собирают. В то же время это частично снимает проблему непрозрачности данных облачных сервисов, которая создает угрозу для контроллеров и субъектов данных.

Еще один пункт регламента GDPR, на который нужно обратить внимание в первую очередь, – это требование обеспечить защиту данных еще на этапе разработки, то есть конфиденциальность должна быть изначально заложена в дизайн продукта [6].

*Сравнительный анализ регламента GDPR и закона 152-ФЗ*

По сравнению с российским законом в регламенте GDPR прописаны более высокие требования к обработке персональных данных, но следует отметить, что есть совпадения в документации и технических решениях.

В таблице 3.2 приведен сравнительный анализ требований и положения о персональных данных [7].

Таблица 3.2 – Сравнение регламента GDPR и закона 152-ФЗ

| Требования регламента | | Персональные данные | |
|---|---|---|---|
| 152 - ФЗ | GDPR | 152 - ФЗ | GDPR |
| Получить явное согласие у пользователя на сбор, обработку и хранение персональных данных | Получить явное согласие у пользователя на сбор, обработку и хранение персональных данных | E-mail | Логин/псевдоним |
| Зарегистрируйтесь в Роскомнадзоре в качестве оператора, работающего с персональными данным | Предоставить пользователю возможность отказаться от согласия на обработку персональных данных | Номер мобильного телефона | Номер мобильного телефона |
| Указать, в какой стране находится хостинг сайта. Хостинг должен располагаться на территории Российской Федерации | Возможность продемонстрировать согласие пользователя на обработку персональных данных | Фамилия, имя, отчество | Медицинская информация |
| Назначить лицо, ответственное за защиту персональных данных пользователей | Назначить лицо, ответственное за защиту персональных данных пользователей | Год, месяц, дата и место рождения | Онлайн-идентификатор (IP-адрес, cookie и т.д.) |
| Обеспечить возможность пользователю удалять все свои персональные данные, хранящиеся на сайте | Обеспечить возможность пользователю удалять все свои персональные данные, хранящиеся на сайте | Адрес места регистрации и проживания | Адрес места регистрации и проживания |

Окончание табл. 3.2

| Требования регламента | | Персональные данные | |
|---|---|---|---|
| 152 - ФЗ | GDPR | 152 - ФЗ | GDPR |
| Обеспечить надежное хранение персональных данных. Например, шифрование данных, псевдонимизация (хранение данных, которые можно идентифицировать с конкретным человеком отдельно от данных, которые к нему относятся) | Обеспечить надежное хранение персональных данных. Например, шифрование данных, псевдонимизация (хранение данных, которые можно идентифицировать с конкретным человеком отдельно от данных, которые к нему относятся) | Семейное, социальное, имущественное положение | Текущее местоположение (координаты на карте) |
| Обеспечить сбор минимального количества персональных данных | Обеспечить сбор минимального количества персональных данных | Образование, профессия, доходы | Образование, профессия, доходы |
| | Обеспечить возможность пользователю просмотреть все свои персональные данные, хранящиеся на сайте | Паспортные данные | |
| | Уведомление пользователей об инцидентах, связанных с утечкой их персональных данных | Другая информация о человеке, которая на прямую или косвенно может его идентифицировать | Другая информация о человеке, которая на прямую или косвенно может его идентифицировать |

Следует отметить, что регламент GDPR позволяет пользователям знать принципы, права и обязанности защиты их данных.

GDPR позволяет решить вопросы:

- какие типы персональных данных собирает компания;
- в какой форме происходит сбор данных;
- как эти данные можно использовать;
- кто имеет к ним доступ;
- есть возможность использования данных для автоматического составления портрета пользователя;
- по каким критериям определяются сроки хранения.

В пользовательском соглашении компании не обязаны прописывать все нюансы использования персональных данных. Однако при запросе пользователя информации, организация обязана ее выслать в течение месяца (с объяснением причины и бесплатно) в удобной для пользователя форме.

Основные отличия GDPR:

1. Компании должны четко и коротко объяснить, какие его данные компания будет собирать и для чего использовать. Основная причина, почему никто не читает пользовательские соглашения – обилие юридических терминов и сложных конструкций. Теперь формулировки должны быть максимально точными.
2. Компании обязаны по просьбе пользователя предоставлять электронную копию его данных. Это правило полезно для тех, кто хочет сменить сервис на аналогичный, не теряя возможности получать основанные на предпочтениях рекомендации.
3. Пользователь имеет право не согласиться на обработку своих данных или отозвать согласие в любой момент.
4. Дети-граждане ЕС младше 16 лет могут регистрироваться в интернете только с согласия родителей или опекунов.
5. Если произошла утечка данных, компания обязана сообщить о ней клиентам и регулирующим органам в течение 72 часов.

GDPR предусматривает право на забвение. Это значит, что пользователь может потребовать у компании удалить свои личные данные. Ранее право распространялось только на поисковые системы, теперь оно касается всех сайтов и сервисов, собирающих личные данные. Правом можно воспользоваться, если:

- данные использованы по назначению и больше не нужны;
- пользователь отозвал согласие на обработку;
- информация собиралась незаконно.

*Штрафы за нарушение норм GDPR*

Работа по GDPR предполагает выполнение ряда требований, нарушение которых влечет за собой серьезные штрафы. Согласно GDPR, одним из основных обязательств является предоставление уведомления в течение трех суток о нарушении конфиденциальности данных (чем выше степень риска, тем быстрее нужно сообщать).

То есть компания, обрабатывающая персональные данные резидентов ЕС для того, чтобы избежать штрафов, обязана иметь полную копию трафика в объеме за последние 72 часа как минимум. Вместе с тем важно проиндексировать трафик и обеспечить быстроту поиска, так как объем этих данных может быть очень большим, порядка сотен терабайт. Данная информация позволяет выявить источник проблемы, если необходимо

воспроизвести трафик и после устранения угрозы проверить соответствие внутренним политикам безопасности организации.

За несоответствие требованиям GDPR на компании могут быть наложены штрафы двух типов:

1) за нарушение правил о согласии на обработку данных ребенка и безопасности персональных данных – до €10 млн или 2% от годового дохода.
2) за нарушение основных прав субъектов данных, принципов обработки и передачи персональных данных и правил согласия - до €20 млн или 4% от годового дохода.

За 2019 год наибольшее количество штрафов получили Венгрия, Испания, Чехия, Болгария, Румыния. В соответствии с нормами GDPR, размеры штрафов пропорциональны нарушению. Наиболее крупные штрафы – за утечку паспортных данных, в области здравоохранения и т.п. [8].

Примером одного из самых крупных штрафов за 2019 г. является штраф в 183 миллиона фунтов стерлингов, предъявленный за нарушение обработки данных авиакомпании British Airways. В сентябре 2018 года у данной авиакомпании были украдены персональные данные примерно 500 000 клиентов авиакомпании (имена, номера банковских карт и их коды CVV, а также адреса электронной почты).

По данным на январь 2020 года сумма штрафов достигла 114 млн евро, регуляторами зафиксировано 160 000 нарушений [8].

За два года существования GDPR уже проявились самые популярные причины штрафов:

- недостаточные технические и организационные меры по обеспечению информационной безопасности;
- недостаточная правовая база для обработки данных;
- несоблюдение принципов обработки данных;
- недостаточное выполнение информационных обязательств.

При применении GDPR компания должна быть уверена, что публичные процедуры, например, процедуры информирования, получения согласия на обработку персональных данных и т.п. доведены до совершенства. Таким образом, целесообразным будет сократить область применения GDPR на периметр отдельных процессов, т.е. локализовать контур применения GDPR.

Список использованных источников:

1. Общий регламент защиты персональных данных (GDPR) Европейского союза. [электронный ресурс]. URL: https://gdpr-text.com/ (дата обращения: 05.09.2020).
2. Заведенская А. Влияние GDPR на российских операторов персональных данных. [электронный ресурс]. URL: https://habr.com/ru/post/423733/ (дата обращения: 06.09.2020).

3. Информационная безопасность цифрового пространства / под ред. Е.В. Стельмашонок, И.Н. Васильевой. – СПб.: Изд-во СПбГЭУ, 2019. – 155 с.
4. GDPR SAM-проект: комплексная оценка на соответствие общему регламенту Евросоюза по защите данных [электронный ресурс]. URL: https://www.infosec.ru/upload/medialibrary/edf/SAM-GDPR-Assessment-Customer-Flyer.pdf (дата обращения: 08.09.2020).
5. Козлов С. Штрафов GDPR еще не было? Будут! Все, что нужно знать о защите персональных данных [электронный ресурс]. URL: https://protocol.ua/ua/shtrafov_gdpr_eshche_ne_bilo_budut_vse_chto_nugno_znat_o_zashchite_personalnih_dannih/ (дата обращения: 10.09.2020).
6. GDPR – что это [электронный ресурс]. URL: https://aizas.ru/gdpr-chto-eto/ (дата обращения: 11.09.2020).
7. Хорошевский А. Регламент GDPR и 152 ФЗ. Требования, сравнение [электронный ресурс]. URL: https://aleksius.com/joomla/rasshireniya/gdpr-i-152-fz (дата обращения: 12.09.2020).
8. General Data Protection Regulation. Регламент Евросоюза о персональных данных [электронный ресурс]. URL: https://www.tadviser.ru/ (дата обращения: 14.09.2020).
9. GDPR: Guidelines, Recommendations, Best Practices [электронный ресурс]. URL: https://edpb.europa.eu/our-work-tools/general-guidance/gdpr-guidelines-recommendations-best-practices_en (дата обращения: 17.09.2020).
10. Guide to the General Data Protection Regulation (GDPR) [электронный ресурс]. URL: https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr (дата обращения: 29.09.2020).
11. Privacy news [электронный ресурс]. URL: https://iapp.org/news (дата обращения: 29.09.2020).

## 3.3. Подходы к разработке типовой модели нарушителя информационной безопасности организации

Модель нарушителя является важной частью обеспечения информационной безопасности организации, в частности – составная часть модели угроз безопасности информации [1, 4]. Поэтому изучение данного вопроса является актуальным.

Согласно ГОСТ РВ 51987-2002 «Типовые требования и показатели качества функционирования информационных систем», под нарушителем безопасности информации понимается субъект, случайно или преднамеренно совершивший действие, следствием которого является возникновение и/или реализация угроз нарушения безопасности информации [2].

В [2] авторы рассматривают следующие критерии классификации нарушителей АС:

- по уровню возможностей;
- по отношению к АС;
- по времени действия;
- по предположениям безопасности.

При этом, необходимо учитывать характеристики случайности и преднамеренности реализации атаки нарушителем, которые определяют его статус. Если атака осуществляется преднамеренно, то нарушитель называется злоумышленником.

При разработке модели нарушителя необходимо принимать во внимание то обстоятельство, что система защиты АС строится на основе модели нарушителя с учетом применяемой технологии обработки информации, условий функционирования и расположения АС. Определяемая модель должна быть адекватна реальному нарушителю.

Под моделью нарушителя понимается его абстрактное (формализованное или неформализованное) описание. Неформальная модель нарушителя, по мнению авторов, отражает его практические и теоретические возможности, время и место действия.

При разработке модели нарушителя определяются [2]:

- предположения о категориях лиц, к которым может принадлежать нарушитель;
- предположения о мотивах действий нарушителя (преследуемых нарушителем целях);
- предположения о квалификации нарушителя и его технической оснащенности (об используемых для совершения нарушения методах и средствах);
- ограничения и предположения о характере возможных действий нарушителя.

В методических документах регуляторов сферы ИБ РФ при моделировании угроз большое внимание уделяется описанию потенциального нарушителя. Так, в проекте «Методика определения угроз безопасности информации в информационных системах» от ФСТЭК РФ подробно описаны виды нарушителей и их мотивация. В банке угроз ФСТЭК РФ, для каждой угрозы задан тип и потенциал нарушителя, который может ее реализовать. В общем, модель нарушителя перестает быть простой формальностью и начинает оказывать большое влияние на перечень актуальных угроз.

Согласно [4] структурными составляющими модели нарушителя являются:

- типы, виды и потенциал нарушителей, которые могут обеспечить реализацию угроз безопасности информации;
- цели, которые могут преследовать нарушители каждого вида при реализации угроз безопасности информации;
- возможные способы реализации угроз безопасности информации.

Авторами [3] представлена модель нарушителя с учетом разделения его уровней на поиск и использование уязвимостей. На рисунке 3.1 представлена схема потенциальных возможностей нарушителя.

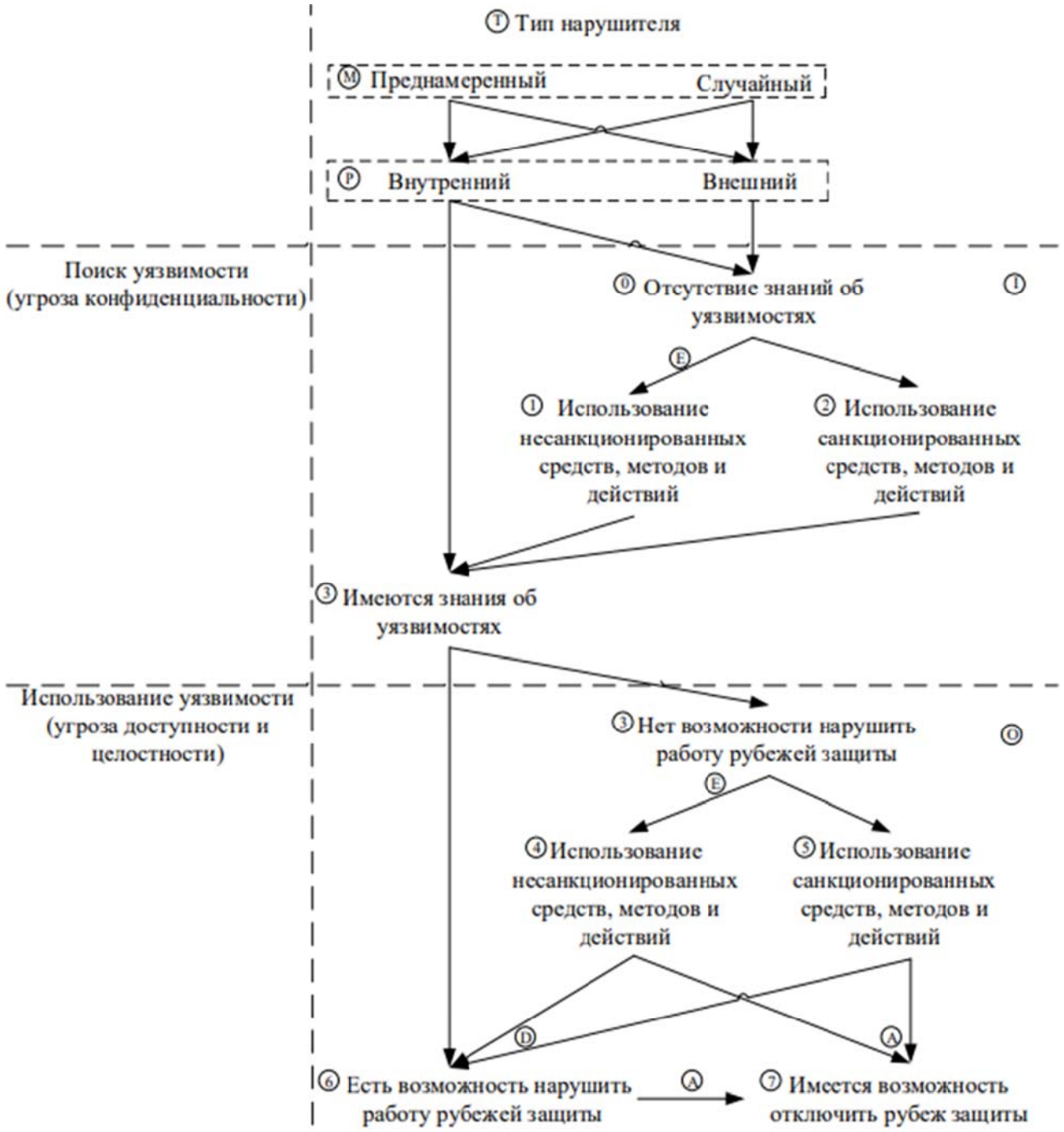



Рисунок 3.1 – Схема отображения параметров нарушителя по [3].

Предположения о том, кем может являться нарушитель для каждого уровня представлены в [3]:

0 уровень – простой внешний нарушитель с очень ограниченным доступом, у которого нет мотивации;

1 уровень – простой внутренний нарушитель с очень ограниченным доступом, (например, электрик), у которых нет мотивации;

2 уровень – внешний нарушитель с очень ограниченным доступом, которые для выявления угроз используют несанкционированные средства для получения информации об уязвимостях в рубежах защиты;

3 уровень – нарушитель, который использует свое положение чтобы собирать информацию об уязвимостях в рубежах защиты, используя санкционированные методы;

4 уровень – нарушитель, обладающий информацией об уязвимостях, может быть, как сотрудником, имеющим отношение к конструированию данного рубежа защиты, так и одним из нарушителей, ранее имевших 2 или 3 уровень;

5 уровень – внутренний нарушитель, имеющий достаточно информации про уязвимости в рубеже, но не имеющий возможности нарушить или преодолеть защиту рубежа, используя свой уровень допуска и использующий для этого санкционированные средства;

6 уровень – внутренний нарушитель имеющий достаточно информации про уязвимости в рубеже, имеет возможность нарушить или преодолеть, используя для этого несанкционированные средства;

7 уровень – внутренний нарушитель, который для достижения своих целей использует санкционированные методы;

8 уровень – внутренний нарушитель с высоким уровнем доступа, имеющий возможность нарушить работу рубежей защиты, пользуясь своим служебным положением;

9 уровень – изначально внутренний нарушитель с очень высоким уровнем доступа, имеющий возможность отключить рубеж защиты, используя свое служебное положение.

Таким образом, существование единой универсальной модели имеет понятное преимущество в виду того, что неполное описание влечет за собой необходимость доработки модели нарушителя под нужды конкретной организации. Возможная избыточность модели не нанесет вреда, но пробелы в описании вероятных нарушителей могут оставить «дыры» в системе безопасности, которые в следствие могут воспользоваться злоумышленники.

Список использованных источников:

1. Гатчин Ю.А., Сухостат В.В., Куракин А.С., Донецкая Ю.В. Теория информационной безопасности и методология защиты информации – Санкт-Петербург: Университет ИТМО, 2018. – 100 с.
2. Грибунин В.Г., Чудовский В.В. Комплексная система защиты информации на предприятии [Электронный ресурс]. URL: https://obuchalka.org/2017082795990/kompleksnaya-sistema-zaschiti-informacii-na-predpriyatii-gribunin-v-g-chudovskii-v-v-2009.html (дата обращения 10.09.2020).
3. Егошин Н.С., Конев А.А., Шелупанов А.А. Формирование модели нарушителя [Электронный ресурс]. URL: https://bit.mephi.ru/index.php/ bit/article/view/273 (дата обращения 20.09.2020).
4. Методика определения угроз безопасности информации в информационных системах (Проект) [Электронный ресурс]. URL: https://fstec.ru/component/attachments/download/812 (дата обращения 20.10.2020).

## 3.4. Об аттестации информационной системы по принципу выделения перечня «типовых сегментов»

С развитием современных технологий, каждый шаг человека становится частью информационной сферы. Даже самые простые вещи уже давно стали частью электронного мира.

Без сомнения, информационные технологии на сегодняшний день играют важнейшую роль в современном мире. Они занимают уникальное положение в нашем обществе и не просто оказывают влияние на его экономические и социальные институты, но и является двигателем глобального экономического роста, проникая во все сферы производственной деятельности и позволяя строить эффективные системы управления.

Мировые тенденции развития общества направлены на постоянное увеличение информационных потребностей, в связи с этим возрастают объемы обработки, передачи и хранения информации, появляются новые методы ее получения и анализа, увеличивается производительность технических средств. Информация становится одним из основных средств решения проблем и задач государства, политических партий и деятелей, различных коммерческих структур и отдельных людей.

Сложно представить жизнь современного человека без информационных систем. С их помощью мы можем получить практически любую государственную услугу: оформить социальные льготы, записаться на прием к врачу, встать в очередь в детский сад, получить или сменить паспорт и многое другое. В связи с возрастающей важностью информационных систем все острее встает вопрос о их безопасности. Задачи обеспечения информационной безопасности государственных и иных организаций России, по мере перехода к цифровым методам управления, становятся все более важными, а в некоторых сферах (оборона, правоохранительная деятельность, дипломатические отношения, экономика и т.п.) – критически важными. Надежное обеспечение информационной безопасности может быть достигнуто только при создании системы защиты информации. Важным этапом в создании элементов защиты информации в государственных и муниципальных информационных системах является аттестация – она является обязательным условием ввода их в эксплуатацию.

На сегодняшний день для государственных информационных систем Приказом ФСТЭК от 11.02.2013 № 17 «Об утверждении Требований о защите информации, не составляющей государственную тайну, содержащейся в государственных информационных системах» [1] определена обязательная их аттестация до ввода в эксплуатацию. А государственные информационные системы – это не статичные объекты информатизации, а большие динамичные и постоянно изменяемые системы регионального или даже федерального масштаба.

Каким же образом аттестовать подобную систему, если она постоянно дополняется и оптимизируется? А, учитывая, что аттестация для государственных информационных систем обязательна, то единственным решением в этой ситуации будет аттестация по принципу выделения перечня «типовых сегментов».

Данное понятие было введено вышеупомянутом Приказом ФСТЭК от 11.02.2013 № 17 и стандартом ГОСТ РО 0043-003-2012 «Защита информации. Аттестация объектов информатизации. Общие положения» в 2012 году. Но для понимания правил распространения аттестата соответствия на типовые сегменты будет достаточно разъяснений в пункте 17.3 Приказа ФСТЭК № 17«Об утверждении Требований о защите информации, не составляющей государственную тайну, содержащейся в государственных информационных системах»: «допускается аттестация информационной системы на основе результатов аттестационных испытаний выделенного набора сегментов информационной системы, реализующих полную технологию обработки информации.» [1]

Но на первый взгляд прочитав п. 17.3 Приказа ФСТЭК № 17 покажется, что все конкретно и понятно. Однако, на практике, в ходе ее проведения возникают «подводные камни» с которыми встречаются обладатели защищаемой информации. По понятным причинам все проблемные вопросы невозможно изложить в приказе, тогда видимо настало время подготовить ФСТЭК-ом методического документа, который бы изложил полностью процедуру проведения аттестационных испытаний выделенного набора сегментов информационной системы.

Помимо приказа, ФСТЭК утвердил методический документ от 11 февраля 2014 г. «Меры защиты информации в государственных информационных системах», где говорится, что сегментирование информационной системы проводится с целью построения многоуровневой (эшелонированной) системы защиты информации путем построения сегментов на различных физических доменах или средах». [2]

Также методический документ поясняет: принципы сегментирования информационной системы определяются оператором с учетом функциональных и технологических особенностей процесса обработки информации и анализа угроз безопасности информации и должны заключаться в снижении вероятности реализации угроз и (или) их локализации в рамках одного сегмента. [2]

Таким образом, типовой сегмент – это выделенный набор сегментов информационной системы, реализующих полную технологию обработки информации, необходимыми соответствиями которых являются: такой же класс защищенности, определены такие же угрозы безопасности информации и реализованы одинаковые проектные решения по самой системе и по системе защиты информации. При этом, соответствие нового сегмента уже выданному аттестату подтверждается в ходе приемочных испытаний.

Аттестация информационной системы проводится в соответствии с программой и методиками аттестационных испытаний до начала обработки информации, подлежащей защите в ГИС. Для проведения аттестации применяются национальные стандарты, а также методические документы, разработанные и утвержденные ФСТЭК России в соответствии с подпунктом 4 пункта 8 Положения о Федеральной службе по техническому и экспортному контролю, утвержденного Указом Президента Российской Федерации от 16.08.2004 г. N 1085.

До начала аттестационных испытаний необходимо утвердить организационно-распорядительные документы по защите информации.

При проведении аттестационных испытаний проводится анализ уязвимостей средств защиты информации, технических средств и программного обеспечения. В случае обнаружения уязвимостей необходимо выполнить их устранение или снизить возможность их применения за счет настройки имеющихся средств защиты. Отчет по анализу уязвимостей как правило оформляется в виде приложения к заключению по результатам аттестационных испытаний.

После проведения аттестационных испытаний разрабатывается протокол проведения аттестационных испытаний и заключение по результатам аттестационных испытаний.

Аттестация выполняться в соответствии со следующими требованиями:

при проведении аттестационных испытаний применяется подход аттестации типовых сегментов;

оформляемая документация учитывает возможность распространение аттестата соответствия на другие сегменты при условии их соответствия сегментам, прошедшим аттестационные испытания и реализующих полную технологию обработки информации;

особенности аттестации на основе результатов аттестационных испытаний выделенного набора ее сегментов, а также условия и порядок распространения аттестата соответствия на другие сегменты информационной системы обязательно будут определены в программе и методиках аттестационных испытаний, заключении по результатам аттестационных испытаний и аттестате соответствия.

При успешном прохождении аттестационных испытаний выдается Аттестат соответствия.

Таким образом, аттестация по принципу типовых сегментов – закрепленный в нормативно-правовых актах единственный подходящий способ аттестации государственной информационной системы, как для проведения аттестационных испытаний, так и для дальнейшего поддержания безопасности системы.

Список использованных источников:

1. Приказ ФСТЭК от 11.02.2013 № 17 «Об утверждении Требований о защите информации, не составляющей государственную тайну, содержащейся в государственных информационных системах».
2. Методический документ ФСТЭК от 11 февраля 2014 г. «Меры защиты информации в государственных информационных системах».
3. Постановление Правительства Российской Федерации от 01.11.2012 г. № 1119 «Об утверждении требований к защите персональных данных при их обработке в информационных системах персональных данных» //Собрание законодательства Российской Федерации от 5 ноября 2012 г.

## 3.5. Экономическая эффективность затрат на инфраструктуру защиты информации промышленного предприятии

Современное состояние информационных технологий поддержки бизнес-процессов характеризуется массовым производством программно-аппаратных средств, начиная от систем класса ERP, заканчивая отдельными бизнес-приложениями. Это многообразие порождает информационную разобщенность между отдельными средствами, и, как следствие, снижение уровня защищенности информации.

С другой стороны, требования международных стандартов информационной безопасности предполагают создание единой системы управления информационной безопасностью всей бизнес- системы предприятия.

В результате указанных причин возникло острое противоречие между возросшими возможностями методов и средств информационных технологий и возможностями методов и средств защиты информационных ресурсов.

В первую очередь это касается инфраструктуры защиты информации бизнес-процессов, которая в свете современных тенденций организации бизнеса играет решающую роль в достижении успеха хозяйствующим субъектом.

На основе проведенного анализа влияния информационной инфраструктуры на реализацию бизнес-процессов [1], анализа содержания информационной безопасности бизнеса, определив цели и основные принципы функционирования инфраструктуры, можно определить ряд предъявляемых требований к системе ее защиты в отношении бизнес-процессов (рисунок 3.5).


Рисунок 3.5 – Роль и место инфраструктуры защиты информации в бизнес – системе

Концептуальная модель инфраструктуры защиты информации бизнес-процессов отражает принципиальные подходы к организации такой системы на промышленном предприятии (рисунок 3.6).

Основным показателем экономической эффективности затрат на инфраструктуру защиты информации промышленного предприятия, как любого инвестиционного проекта является чистая приведенная стоимость (*NPV*) в период времени от *t* до *T*:

$$NPV = \sum_{t=1}^{T} \frac{\Delta if_t(R) - \Delta of_t(R)}{(1+E)^t} - K_R$$

где: $\Delta if_t(R)$ – изменение входного денежного потока с учетом проведения мероприятий по защите информации;

$\Delta of_t(R)$ – изменение выходного денежного потока с учетом проведения мероприятий по защите информации;

$K_R$ – внеоборотные и оборотные информационные активы инфраструктуры защиты информации;

$E$ – норма прибыли на капитал.

А теперь посмотрим, как эффективная инфраструктура защиты информации изменяет основные показатели производственно-хозяйственной деятельности предприятия. Организация инфраструктуры защиты информации на промышленном предприятии безусловно влияет на результаты его хозяйственной деятельности. Основными показателями хозяйственной деятельности промышленного предприятия являются:

1. основной финансовый результат (прибыль или убыток);

2. рентабельность производственных фондов и продукции;
3. стоимость предприятия (балансовый подход, доходный подход);
4. фактическая (актуальная) ликвидность.

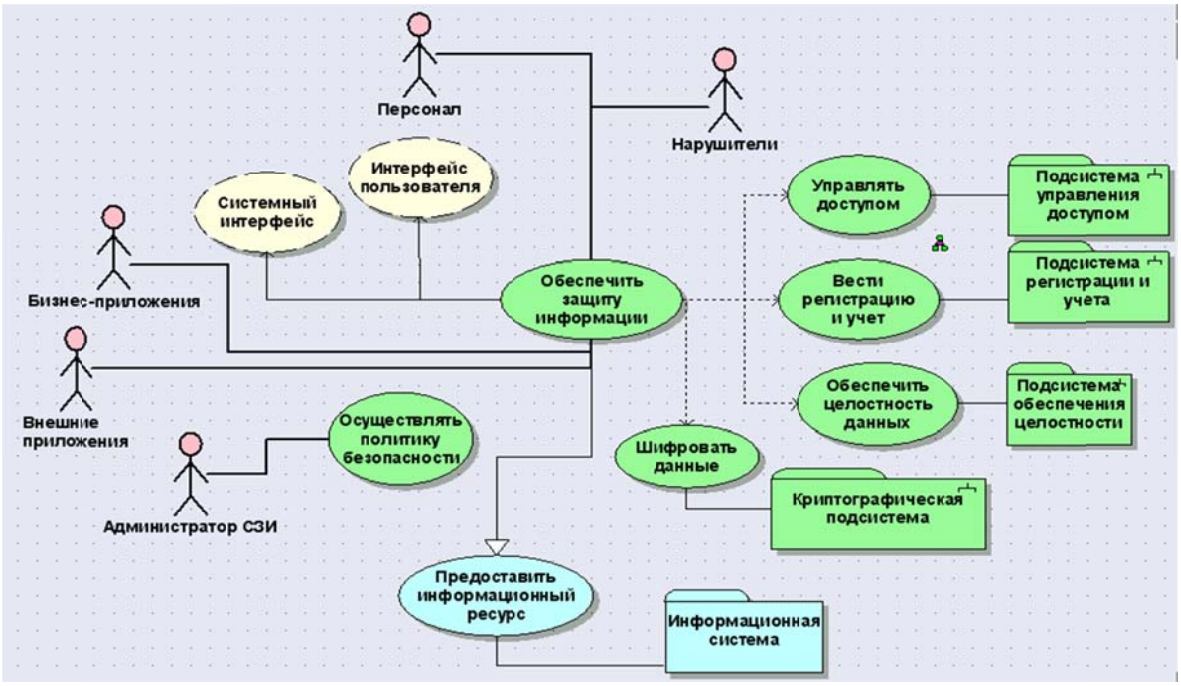



Рисунок 3.6 – Концептуальная модель инфраструктуры защиты информации бизнес-процессов

Рассмотрим влияние затрат на инфраструктуру защиты информации на прибыль предприятия.

В результате проводимых мероприятий по защите информации на промышленном предприятии прибыль должна увеличиться по сравнению с базовым вариантом, не предусматривающим такой защиты.

$$\Delta П(R) = П(R) - П,$$

где: $\Delta П(R)$ – годовой прирост прибыли в результате мероприятий по защите информации;

$П(R)$ – прибыль при условии проведения мероприятий по защите информации за год;

$П$ – прибыль в условиях отсутствия защиты информации (базовый вариант) за год.

Затраты на инфраструктуру защиты информации содержат две составляющие: единовременные (инвестиции) и эксплуатационные (текущие). Затраты на инфраструктуру защиты информации будут оправданы при соблюдении следующего условия:

$$\Delta П(R) \geq C_R + E \cdot K_R,$$

где $C_R$ – годовые эксплуатационные затраты на защиту информации.

Проводимые мероприятия по защите информации должны положительно сказаться на показателе рентабельности:

$$Q_б = \frac{П}{Ф_{пр}}$$

$$Q(R) = \frac{П + \Delta П(R) - C_R}{Ф_{пр} + K_R},$$

где : $Q_б$ – рентабельность базовая, в условиях отсутствия мероприятий по защите информации;

$Q(R)$ – рентабельность с учетом мероприятий по защите информации;

$Ф_{пр}$ – стоимость производственных фондов.

Затраты на инфраструктуру защиты информации экономически с точки зрения рентабельности оправданы при соблюдении условия:

$$\frac{П + \Delta П(R) - C_R}{Ф_{пр} + K_R} \geq \frac{П}{Ф_{пр}}.$$

Обратимся к условию эффективности затрат на инфраструктуру защиты информации на основе доходного подхода к оценке стоимости предприятия.

Как известно, приведенная стоимость предприятия ($PV$) учитывает временную ценность денег (в период времени от $t$ до $T$):

$$PV = \sum_{t=1}^{T} \frac{if_t - of_t}{(1+E)^t} + \frac{PV_T}{(1+E)^T}.$$

С учетом создания и функционирования эффективной инфраструктуры защиты информации приведенную стоимость предприятия $PV(R)$ можно оценить:

$$PV(R) = \sum_{t=1}^{T} \frac{if_t + \Delta if_t(R) - of_t - \Delta of_t(R)}{(1+E)^t} + \frac{PV_T + \Delta PV_T(R)}{(1+E)^T},$$

где: $PV$ – приведенная стоимость предприятия (базовый вариант) в условиях отсутствия мероприятий по защите информации;

$PV(R)$ – приведенная стоимость предприятия с учетом проведения мероприятий по защите информации;

$if_t$ – входной денежный поток;

$of_t$ – выходной денежный поток;

$E$ – ставка дисконта.

Следующее условие эффективности затрат на инфраструктуру защиты информации может быть определено соотношением:

$$\sum_{t=1}^{T} \frac{if_t + \Delta if_t(R) - of_t - \Delta of_t(R)}{(1+E)^t} + \frac{PV_T + \Delta PV_T(R)}{(1+E)^T} \geq \sum_{t=1}^{T} \frac{if_t - of_t}{(1+E)^t} + \frac{PV_T}{(1+E)^T}$$

Фактическая (актуальная) ликвидность определяется величиной сальдо накопления денежных средств на расчетном счете предприятия для каждого подпериода. Очевидно, что капитальные и эксплуатационные затраты на инфраструктуру защиты информации должны не нарушать условия, при котором сальдо накопленных денежных средств на конец каждого подпериода ( $S_t^k$ ) должно быть не меньше заданной величины.

$$S_t^к = (S_1^н - K_R)(1+E)^t + \sum_{\tau=1}^{t}(if_\tau + \Delta if_\tau(R))(1+E)^{t-\tau} - \sum_{\tau=1}^{t}(of_\tau + \Delta of_\tau(R))(1+E)^{t-\tau} -$$

$$- \sum_{\tau=1}^{t} of_\tau (1+E)^{t-\tau} - K \geq S_{доп\,t}^к \quad , \quad t = \overline{1,T}$$

где: $S_1^н$ – сальдо накопленных денежных средств на расчетном счете на начало первого подпериода;

$if_\tau$ – приток денежных средств в $\tau$-ый подпериод на расчетный счет;

$\Delta if_\tau(R)$ – дополнительный приток денежных средств в $\tau$-ый подпериод на расчетный счет в условиях функционирования системы защиты информации;

$of_\tau$ – отток денежных средств в $\tau$-ый подпериод с расчетного счета;

$\Delta of_\tau(R)$ – дополнительный отток денежных средств в $\tau$-ый подпериод с расчетного счета с учетом затрат на систему защиты информации;

$S_{доп\,t}^к$ – допустимый остаток денежных средств на расчетном счете предприятия на конец $\tau$-ого подпериода.

Результаты анализа наиболее прогрессивных тенденций развития организации управления хозяйственной деятельностью предприятия убедительно свидетельствуют о необходимости организации инфраструктуры защиты информации бизнес-процессов, как одного из вспомогательных (инфраструктурных) бизнес-процессов бизнес-системы. Это обеспечит организацию инфраструктуры защиты информации в тесной взаимосвязи с проектированием других бизнес-процессов, что увеличит их интегрированность, гибкость, сбалансированность и управляемость.

Список использованных источников:

1. Стельмашонок Е.В. Информационная инфраструктура поддержки и защиты корпоративных бизнес-процессов: экономико-организационные проблемы. – СПб.: СПбГИЭУ, 2005. – 151с.

## 3.6. Гуманитарные аспекты информационной безопасности: специфика научного знания в социальном контексте

Проблемы научного познания в определенных ситуациях влияют на развитие научных дисциплин и результаты научного знания неотделимы от общества, приумножают его блага. Само же научное знание представляет собой тот пласт информации, специфика обращения с которым требует определенной подготовки, знаний в виде базиса. Отсутствие у индивида необходимых качеств для взаимодействия с научным знанием влечет за собой негативные последствия в формировании картины мира и восприятия действительности.

Особенность научного знания заключается в том, что оно выходит за рамки содержания и формы представления информации, а понимание «научности» в социуме окружено ореолом авторитета. В случае неумения оперировать научным знанием в ходе рассуждений привязка собственной аргументации к научной сфере являет собой манипуляцию человеческим сознанием, приводит к информационному искажению у индивида.

Специфичность научного знания находится в тесной связи с феноменами других наук. Социологический феномен элитарности знания, как следствие сложности и необычности, легко сопоставим со знанием научным. Для большинства людей, не связывающих свою жизнь с научной деятельностью, такая информация не является актуальной, что снижает ее качество в терминах информационной безопасности.

Таким образом, особенности взаимодействия с научной информацией в социо-гуманитарной сфере, обусловливают актуальность безопасности такого взаимодействия в контексте информационной безопасности.

В данном параграфе исследуются примеры взаимосвязи социо-гуманитарных феноменов с угрозами информационной безопасности индивида и ключевыми особенностями научного знания, конкретизируется проблематика такой взаимосвязи, предлагаются первичные методы упреждения дестабилизирующих воздействий.

*Особенности научного знания в контексте исследования*

В рамках работы уместно выделить три основополагающих критерия научного знания, поясняющих взаимосвязь социо-гуманитарной сферы и области информационной безопасности.

Во-первых, научное знание обладает собственным методологическим аппаратом, развивающейся терминологией, философской базой, что предопределяется *узконаправленностью знания*, получаемого в ходе научной деятельности. Научный институт в свою очередь обособляется от субъективного восприятия и работает объективными методами познания. Иные картины мира, с которыми сталкивается человек, не отличаются та-

кой узостью и конкретностью изучаемых вопросов и обозначающихся проблем.

Во-вторых, множество взглядов, концепций и школ, продвигающих свою версию объяснения и описания сущностей предметов исследования научной дисциплины, обусловливает идейно-концептуальный *плюрализм*. Примерами могут служить психологическая, социологическая, экономическая науки.

Третьим отличающим критерием научного знания является *авторитетность*, связанная с научной деятельностью в целом. Принципы научной деятельности, проявляющиеся в системности, проверке данных на валидность, объективности излагаемых суждений, а также следствия таких принципов – результат применения синтезируемого знания, зарекомендовали научный подход как подход, на результаты которого стоит полагаться в подавляющем большинстве случаев. Не понимая особенностей научной деятельности, постановки вопросов ее исследований, граница применимости научного метода и уместность такого применения может оказаться значимо размытой. Таким образом, ярлык научности приобретает авторитет сам по себе, для индивида его наличие оказывается достаточным фактором для принятия решения.

*Научное знание в обыденной жизни*

Оперирование научными знаниями не ограничивается сферой научной практики. Общедоступность таких знаний через образовательные и информационные источники приводит к непосредственному взаимодействию индивидов с научным аппаратом. Здесь важно отметить глубину такого знакомства. Обыденное знание включает в себя и здравый смысл, и приметы, и назидания, и рецепты, и личный опыт, и традиции. Оно, хотя и фиксирует истину, но делает это не систематично и бездоказательно. Его особенностью является то, что такое знание используется человеком практически неосознанно и в своем применении не требует предварительных систем доказательств [5].

Важно то, что субъективное восприятие повседневной жизни (феноменологический анализ) воздерживается от причинных и генетических гипотез так же, как и от утверждений относительно онтологического статуса анализируемых феноменов. Обыденное сознание содержит много до- и квазинаучных интерпретаций повседневной жизни, которые считаются само собой разумеющимися. Поэтому при описании повседневной реальности прежде всего следует обращаться к такого рода интерпретациям как само собой разумеющиеся, хотя и в рамках феноменологических скобок [2].

Хрестоматийность отличий научного и обыденного знания характеризуется тем, что в ходе обмена научными знаниями на бытовом уровне, индивиды практически не получают прагматической пользы. Если про-

цесс обмена мнениями и подчерпнутыми знаниями является частью социализации, самопрезентования и досуга, то неприменимость самих знаний на практике делает их бесполезными для простого обывателя. При этом неформальный разговор влияет на саму структуру используемого в диалоге знания, появляется риск исказить последнюю за счет неосведомленности говорящих о нормах развития научной мысли, системы аргументации и доказательства.

Далее следует отметить, что теория как пласт научного знания обладает особенностью идеализированного представления вещей в рамках исследуемого вопроса. Однако реальность очень нечасто позволяет использовать теоретические наработки без серьезных корректировок и даже пренебрежений. В этой связи людям (специалистам в профессиональной сфере) не нужно прибегать к научной составляющей их ремесла за ее ненадобностью для выполнения своих функций как работников.

*Негативный аспект обыденного применения научного знания*

Использования ярлыка научности, исходя из ее авторитетности, экономически выгодно. Такой подход используется как в рекламной деятельности, так и в качестве сомнительной аргументации для достижения определенных целей. Зачастую к такому методу прибегают СМИ, заинтересованные в формировании определенного взгляда на вещи у целевой аудитории [1].

Любая идеология стремится объяснить и обосновать тот социальный и политический порядок, который она защищает, через апелляцию к естественным законам. При этом конечными аргументами, безотказно действующими на публику, являются следующие фразы: «так устроен мир», «такова природа человека». Поэтому идеологи тщательно создают модель человека, используя всякий идущий в дело материал: научные сведения, легенды, верования, даже дичайшие предрассудки. Разумеется, для современного человека убедительнее всего звучат фразы, напоминающие смутно знакомые со школьной скамьи научные формулы и изречения великих ученых.

Опасность манипулятивных средств не является чем-то кардинальным новым, гораздо важнее – понимать, откуда опасность может исходить. Истинную опасность представляет из себя возможная установка о беспрекословной ценности научного знания в сознании индивида, отсутствие самостоятельного критического мышления, неумение анализировать и отбирать источники информации [3].

Таким образом, актуальность проблемы неполезности научной информации в обыденной среде ввиду ее неприменимости в быту и в большинстве сфер профессиональной деятельности заключается в том, что общее снижение уровня полезности информации снижает качество ин-

формации. Следовательно, возникает потенциальная информационная угроза для человека и его продуктивной деятельности.

*Наука и кентавризм для простого обывателя*

С точки зрения историко-философского анализа развития знаний, предпринятого В.С. Хомяковой, безопасность имеет гуманитарную сущность. На ранних этапах истории, на этапе мифологического мировоззрения безопасность понимается через отношение человека с природой, опасные явления которой кажутся ему непостижимыми и безропотно принимаются, а сами опасности персонифицируются. В мифе человек имеет дело с одушевленными существами, от которых зависит его существование, а познание безопасности протекает в форме их «узнавания» [6]. Зарождение и стремительное развитие научной картины внесло в миропонимание много конкретики. Ранние этапы синтезы научного знания подразумевают множество равноправных теорий, поиск подтверждения которых формируют современную догму, на которой строятся дальнейшие исследования, реализуются эмпирическая сфера науки, делаются значимые для технологического прогресса выводы.

Состояние науки в ряде дисциплин (экономика, психология, социология) подводит нас к ряду интерпретаций реальности. Схожую ситуацию мы наблюдаем при анализе феномена кентавризма – социологического феномена, рассмотренного Тощенко. В таком контексте уместно провести параллель между исследованием Хомяковой и дескриптивным аспектом научного знания. Объяснение окружающего мира – все такая же актуальная человеческая потребность. При этом противоречащие друг другу научные взгляды на мир продолжают одновременно сосуществовать, развиваться за счет оппозиции, которую представляет другая сторона. Так научные идеи превращаются в кентавр-идеи [4].

Говоря о кентавризме, мы прежде всего говорим об исторически сформировавшемся шаблоне поведения социальной группы, основанный на приписывании наблюдаемым объектам и феноменам несовместимых черт и свойств, порождая парадоксы и логические противоречия.

Потенциальный вред кентавр-идей проявляется в возможной манипуляции человеческим сознанием. В одной из работ феномен уже был исследован в таком контексте, зависимость между применением кентавр-идей в качестве средства манипуляции и целенаправленного изменения мнения масс была установлена [1].

Следует заключить, что работа с парадоксальными сущностями требует особой подготовки. В противном случае индивид или склонится в сторону одной из них, или же потеряет суть исследуемого. И в том и в другом случае налицо – негативное влияние на информационную картину мира индивида, которая может распространять искажения в ходе бытовых обсуждений на других социальных агентов.

Мышление индивида развивается за счет получаемых им знаний, связей, которые он может воссоздать у себя в голове, самостоятельно закрепить их. Абстрактный уровень мышления, значимым проявлением которого являются и наукоемкие суждения, формирует образ мышления, меняет не информационный базис индивида, но тот инструмент, посредством которого базис будет меняться в дальнейшем. Данный процесс не происходит осознанно, ровно, как и невозможно целенаправленное развитие личности без наличия особой для того информации. Научное знание здесь выступает определенным полем для обучения со своими примерами и вопросами, на которые каждый волен попытаться ответить самостоятельно [3].

Последним важным моментом в рамках данной темы является необходимость обновления кадрового научного потенциала. Ограничение доступа к научному знанию приведет к истощению потока интересующихся и подающих надежд юных умов, чьи старания впоследствии смогут самостоятельно двигать науку дальше.

Список использованных источников:

1. Бурзунов Д.Д., Крайковский Г.А. Разработка концепции системы автоматического анализа сетевого трафика на основе моделей самообучения и использования статистических методов классификации данных // Сб. статей «Цифровые технологии обработки и защиты информации» / Под ред. Е.В. Стельмашонок, И.Н. Васильевой. – Санкт-Петербург, СПбГЭУ, 2020. – С. 26-29.
2. Питер Бергер, Томас Лукман. Социальное конструирование реальности [электронный ресурс] URL: https://studopedia.ru/20_63029_piter-berger-tomas-lukman-sotsialnoe-konstruirovanie-realnosti.html (дата обращения 20.10. 2020).
3. Теплов Э.П., Гатчин Ю.А., Нырков А.П., Коробейников А.Г., Сухостат В.В. Гуманитарные аспекты информационной безопасности: основные понятия, логические основы и операции. – СПб, Университет ИТМО, 2016. – 122 с.
4. Тощенко Ж.Т. Кентавр-проблема (Опыт философского и социологического анализа): [монография] / Ж.Т. Тощенко. М.: Новый хронограф, 2011 // Социологический журнал, 2013. – С. 172-178.
5. Философия науки в вопросах и ответах: Учебное пособие для аспирантов / В.П. Кохановский [и др.]. – Ростов н/Д: Феникс, 2006. – 352 с.
6. Хомякова, В.С. Безопасность как фактор устойчивого развития (социально-философский аспект): автореферат дис. на соискание ученой степени кандидата философских наук / В.С. Хомякова. – Чита, 2007. – 18 с.

# ГЛАВА 4. ОБРАЗОВАНИЕ И ВОПРОСЫ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

## 4.1. Основные требования к безопасности информационных процессов при организации обучения с использованием дистанционных образовательных технологий

В условиях предупреждения распространения новой коронавирусной инфекции особенно актуальным становится вопрос о корректировке образовательного процесса, связанной с реализацией обучения в дистанционном формате. Для проведения всех видов аудиторных занятий в удаленном режиме между преподавателями и обучающимися должен устанавливаться режим видеоконференцсвязи. С этой целью используются различные программные средства унифицированных коммуникаций в информационно-телекоммуникационной сети Интернет. Из существующих на сегодняшний день такого вида программ по предоставлению услуг удаленной конференц-связи наибольшей популярностью пользуется Zoom, разработанная компанией Zoom Video Communications. По различным данным, количество пользователей сервиса увеличилось с 10 млн человек ежедневно в декабре 2019 года до 200 млн человек в начале апреля 2020 года. Необходимость соблюдать карантин привела к тому, что сервис использовали более 90000 школ в 20 различных странах мира [1]. Вместе с тем возникает закономерный вопрос обеспечения конфиденциальности и целостности обрабатываемой информации, а также ее доступности в ходе реализации основных образовательных программ.

Известен случай, когда сервис видеоконференцсвязи Zoom уличили в неправильном утверждении об использовании оконечного (end-to-end) шифрования, выяснилось, что платформа фактически использует свое собственное определение термина, которое позволяет серверу Zoom получать доступ к незашифрованному видео и аудио с видеоконференций [2]. При оконечном шифровании обеспечиваются условия, когда доступ к исходному сообщению имеется только у отправителя и получателя информации [3].

На этом фоне компания в официальном блоге извинилась перед пользователями: «Мы хотим начать с извинений за путаницу, которую мы вызвали неправильным предположением, что сборки Zoom могут использовать оконечное шифрование. Zoom всегда стремился использовать шифрование для защиты контента в максимально возможном количестве сценариев, и в этом духе мы использовали термин оконечное шифрование», после этого были внесены изменения в документацию [4].

Другой проблемой является принципиальная открытость телеконференции, возможность подключения к ней произвольных участников, что

на практике означает, что путем простого подбора, случайного генерирования ссылки можно подсоединиться к активной в данный момент времени видеоконференцсвязи. При этом у ведущего имеется возможность заблокировать несанкционированного участника. Однако, если его подсоединение не будет вовремя замечено, у нарушителя имеется возможность сделать запись или транслировать свой контент. Что и происходило в период карантинных мероприятий, когда тысячи записей видеозвонков Zoom попали в открытый доступ [5,6].

Кроме того, специалистами был обнаружен ряд уязвимостей и сомнительных функций, таких как автоматическая установка на компьютер без участия пользователя, автоматическое добавление в контакты посторонних лиц, под управлением операционной системы MacOS разрешение удаленного подключения веб-камеры пользователя для любого вредоносного веб-сайта, автоматическое преобразование путей к файлам в кликабельные ссылки, отправка данных в Facebook, даже если у пользователя отсутствует учетная запись в этой социальной сети [7,8]. На этом фоне ряд крупных компаний и государственных правительств во всем мире отказались от использования Zoom. Среди них SpaceX – американская компания, производитель космической техники [9], Правительство Китайской Республики (Тайвань) [10], Министерство иностранных дел Германии [11].

В этих условиях становиться затруднительным с точки зрения информационной безопасности использовать Zoom в образовательном процессе. Приоритетом должно являться отечественное программное обеспечение. Однако, к такому виду программ должен выдвигаться ряд требований. Прежде всего, это обеспечение соблюдения нормативно-правовых актов Российской Федерации [12-14]. Должна иметься возможность управления доступом, в частности, подключения ограниченного круга авторизованных пользователей. В случае необходимости использования в государственных информационных системах средства криптографической защиты информации должны быть сертифицированы по соответствующему классу. Программное обеспечение должно позволять осуществлять работу на всех доступных платформах: Windows, MacOS, Linux, IOS, Android. Должна создаваться возможность объединять видеосвязью учебные классы, полигоны, лаборатории, рабочие места, мобильные устройства и браузеры.

В ходе внедрения выстраивается инфраструктура с развертыванием программного продукта на определенное количество лицензий на базе серверов образовательной организации высшего образования. На основании расписания учебных занятий ведущий преподаватель должен иметь возможность заходить в любое время в клиентское приложение. Для организации занятия лекционного типа в режиме видеоконференцсвязи долж-

на иметься возможность создания групповой конференции. Причем она может иметь как симметричный характер, позволять обеспечивать интерактивное занятие с возможностью взаимного общения аудитории, так и асимметричный, позволять осуществлять проведение лекции с возможностью говорить только преподавателю. Должна быть предусмотрена возможность проведения занятия несколькими преподавателями в асимметричном формате.

Аудитории для проведения учебных занятий в дистанционной форме должны быть спроектированы по принципу переговорных комнат. В их оснащение должны входить профессиональные устройства шумо- и эхо подавления. Необходимо обеспечить политику Quality of Service (QoS)[1], с размещением трафика видеоконференцсвязи в отдельную подсеть с настройкой приоритезации и маркировкой трафика, чтобы он передавался в первую очередь, а запросы от браузеров, почтовых клиентов и прочий некритичный к задержкам трафик имел приоритет ниже. Это позволит существенно повысить качество обслуживания (QoS), разделив доступную пропускную способность между приложениями. Установив правила (QoS), потоковое видео будет воспроизводиться на достаточно высоком уровне, позволяющем обучающимся более качественно усваивать учебный материал.

Таким образом, проанализировав практику использования наиболее распространенного программного средства для организации дистанционного обучения, были выявлены основные недостатки этого продукта. Основываясь на них, сформулированы основные требования по обеспечению безопасности информационных процессов для программных средств при организации обучения с использованием дистанционных образовательных технологий.

Список использованных источников:

1. Популярность программ групповых звонков выросла во время самоизоляции [электронный ресурс] // Информационное агентство КРАСНАЯ ВЕСНА, 05.04.2020. URL: https://rossaprimavera.ru/news/284bd478 (дата обращения: 19.10.2020).
2. Micah Lee, Yael Grauer. Zoom Meetings Aren’t End-to-End Encrypted, Despite Misleading Marketing [электронный ресурс] // The Intercept, 31.03.2020. URL: https://theintercept.com/2020/03/31/zoom-meeting-encryption/ (дата обращения: 19.10.2020).
3. Информационная безопасность цифрового пространства: коллективная монография /под ред. Е.В. Стельмашонок, И.Н. Васильевой. – СПб.: Изд-во СПбГЭУ, 2018. – С. 20-36.

[1] Quality of Service (в перев. с англ. «качество обслуживания») – технология предоставления различным классам трафика различных приоритетов в обслуживании

4. The Facts Around Zoom and Encryption for Meetings/Webinars [электронный ресурс] // Zoom Blog, 01.04.2020. URL: https://blog.zoom.us/facts-around-zoom-encryption-for-meetings-webinars/ (дата обращения: 19.10.2020).
5. Тысячи записей видеозвонков Zoom попали в открытый доступ [электронный ресурс] // Коммерсантъ, 04.04.2020. URL: https://www.kommersant.ru/doc/4314611 (дата обращения: 19.10.2020).
6. Российским школам посоветовали отказаться от занятий в Zoom после показа порно [Электронный ресурс] // Lenta.ru, 15.04.2020. URL: https://lenta.ru/news/2020/04/15/saratov/ (дата обращения: 19.10.2020).
7. Zoom is Leaking Peoples' Email Addresses and Photos to Strangers // Vice, 01.04.2020 [Электронный ресурс]. URL: https://www.vice.com/en/article/k7e95m/zoom-leaking-email-addresses-photos (дата обращения: 19.10.2020).
8. Patrick Wardle. The 'S' in Zoom, Stands for Security. Uncovering (local) security flaws in Zoom's latest macOS client, 30.03.2020 [электронный ресурс]. URL: https://objective-see.com/blog/blog_0x56.html (дата обращения: 19.10.2020).
9. Munsif Vengattil, Joey Roulette. Elon Musk's SpaceX bans Zoom over privacy concerns -memo [электронный ресурс] // Reuters, 02.04.2020. URL: https://www.reuters.com/article/us-spacex-zoom-video-commn/elon-musks-spacex-bans-zoom-over-privacy-concerns-memo-idUSKBN21J71H (дата обращения: 19.10.2020).
10. Zoom развеяла сомнения в информационной безопасности по поводу отправки зашифрованных сообщений напрямую в Китай? [электронный ресурс] // Taiwan news. 06.04.2020. URL: https://www.taiwannews.com.tw/ch/news/3911165 (дата обращения: 19.10.2020).
11. German foreign ministry restricts use of Zoom over security concerns [электронный ресурс] // Reuters, 08.04.2020. URL: https://www.reuters.com/article/us-health-coronavirus-germany-zoom/german-foreign-ministry-restricts-use-of-zoom-over-security-concerns-report-idUSKBN21Q1SC (дата обращения: 19.10.2020).
12. Федеральный закон от 27 июля 2006 г. № 149-ФЗ «Об информации, информационных технологиях и о защите информации» [электронный ресурс] // Гарант.ру. Информационно-правовой портал. URL: https://base.garant.ru/12148555/ (Дата обращения 14.10.2020).
13. Федеральный закон от 27 июля 2006 г. № 152-ФЗ «О персональных данных» [электронный ресурс] // Гарант.ру. Информационно-правовой портал. URL: https://base.garant.ru/12148567/ (Дата обращения 14.10.2020).
14. Федеральный закон от 27.12.2002 № 184-ФЗ «О техническом регулировании» [электронный ресурс] // Гарант.ру. Информационно-правовой портал. URL: http://base.garant.ru/12129354/ (Дата обращения 14.10.2020).

## 4.2. Использование интерактивной среды Jupyter Lab для формирования аналитических навыков у бакалавров информационной безопасности

Несмотря на утверждение в 2016 году профстандартов [1] на сегодняшний день недостаточно внимания уделяется аналитическим навыкам при подготовке бакалавров информационной безопасности (ИБ). В связи с

этим в параграфе приводится авторский опыт преподавания дисциплины «Программно-аппаратные средства защиты информации», устраняющий этот недостаток.

Набор навыков современного специалиста по ИБ крайне широк, но условно его можно разделить на две области (в терминах военных учений) – Blue и Red Teams. Каждая из этих команд решает определенные задачи.

Red Team («Красная команда») занимается исследованием компьютерных систем на наличие уязвимостей, иногда в их сферу деятельности входит проникновение на территорию компании с помощью методов социальной инженерии, например, под видом коммунальных служб. Red Team разрабатывает набор инструментов и эксплойтов для тестирования системы на проникновение. «Красной команде» достаточно найти одно слабое звено в системе, чтобы получить доступ, например, к учетным записям пользователей и другим конфиденциальным сведениям. Навыки Red Team можно приобрести, участвуя в соревнованиях по кибербезопасности (CTF).

Blue Team («Синяя команда») ищет следы инцидентов ИБ и реагирует на них. В их компетенцию входит анализ логов (сетевого трафика), собранных от всех критических элементов компьютерной системы, установка и настройка систем защиты информации (SIEM), активная реакция на инциденты и пр. Современные APT-атаки распределены во времени: злоумышленник обычно устанавливает бэкдор, способный обновлять свои компоненты и собирать информацию о целевой системе (см. Kill Chain [2]). Отсюда участникам Blue Team важно обладать аналитическим складом мышления и навыками расследования компьютерных преступлений. Полный перечень навыков «Синей команды» представлен в [3].

На взгляд автора, в вузе целесообразно сосредоточиться на формировании у обучающихся аналитических навыков Blue Team, и помочь в этом может интерактивная среда разработки Jupyter Lab [4], которая входит в состав дистрибутива Anaconda [5].

Впервые широко о применении Jupyter Lab в деятельности специалистов по ИБ заговорил Джон Ламберт [6, 7], инженер компании Microsoft.

Jupyter Lab [8] – это расширяемая клиент-серверная среда разработки, которая поддерживает более 40 языков программирования (или kernels): Python, R, C++ и пр. [9]. Клиентская часть открывается через браузер и позволяет в отдельных ячейках запускать код на выбранном языке программирования.

Исходный код в Jupyter Lab группируется в Блокноты (Notebooks), идею которых переняли из Wolfram Mathematica [10]. Блокноты состоят из набора ячеек. Каждая ячейка может быть исполняемой или содержать

комментарии в формате Markdown, включая LATEX. Таким образом, каждый Блокнот превращается в интерактивную статью с возможностью верификации полученных результатов (рисунок 4.1). Благодаря мировому сообществу ученых Jupyter Lab превратился в одну из самых популярных сред для проведения анализа данных [11].

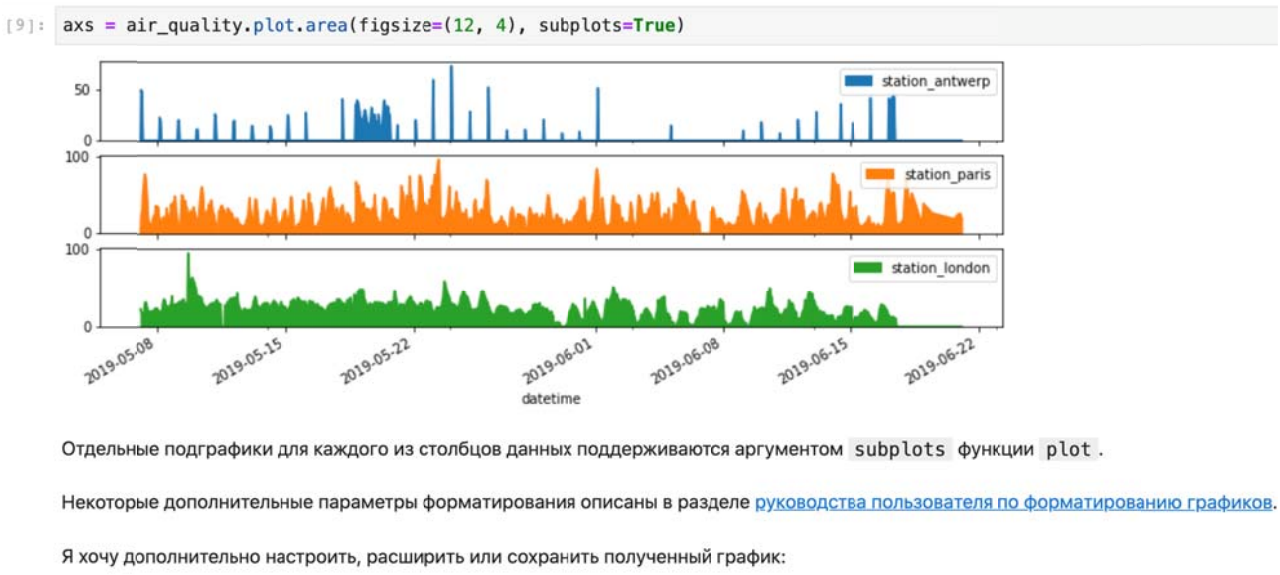



Рисунок 4.1 – Пример Блокнота Jupyter Lab с графиком и комментариями

Jupyter Lab поддерживает десятки языков программирования, но на сегодняшний день наиболее перспективным для решения задач автоматизации является Python [12]. Он создавался в начале 90-ых с одной стороны, как замена командной оболочке bash, а с другой – как язык для обучения программированию [13]. В связи с этим будем рассматривать схему работы Jupyter Lab (рисунок 4.2) на примере языка Python.

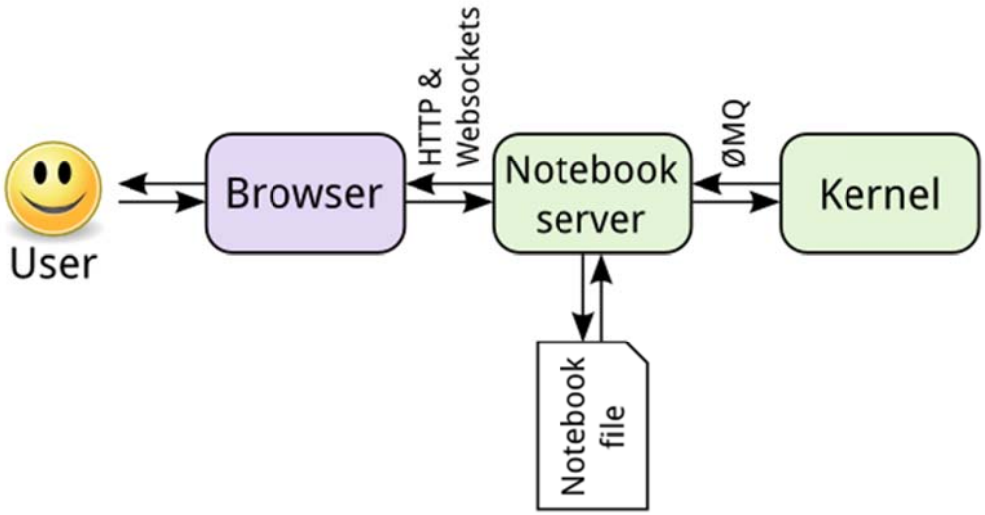



Рисунок 4.2 – Архитектура среды Jupyter Lab [14]

Пользователь в браузере выполняет код на языке Python. Notebook-сервер обращается к Kernel, в качестве которого выступает приложение IPython – расширенная оболочка языка Python (историю ее создания см. в [15]). Затем, получив результат от IPython, Notebook-сервер генерирует файл в формате JSON с расширением *.ipynb* (Блокнот) и отправляет его обратно клиенту. Таким образом внутри Jupyter Lab происходит интерактивное взаимодействие между браузером и оболочкой IPython.

Помимо встроенных возможностей языка Python приложение IPython (Kernel на рисунке 4.2) позволяет в интерактивном режиме обращаться к командному интерпретатору операционной системы (ОС).

Рассмотрим несколько примеров работы в Jupyter Lab.

Первый – получение идентификаторов процессов, принадлежащих пользователю. Данная задача разбивается на несколько шагов.

1) Вывести информацию о всех выполняющихся в ОС процессах:

```
In [1]: !ps aux
USER               PID  %CPU %MEM      VSZ    RSS   TT  STAT STARTED      TIME COMMAND
dm_fedorov        1133  18,6  2,6  5121356 213948   ??  S    27сен20 103:49.27 /Applications/Go
dm_fedorov       40741  13,7  1,8  8903832 150876   ??  S     1:51    0:05.27 /Applications/Go
…
```

2) Выполнить фильтрацию (в методе *grep* указывается аргумент-столбец, по которому производится поиск строки 'dm_fedorov'):

```
In [2]: ps = !ps aux
        ps.grep('dm_fedorov', field=0).fields(1)
['1133',
 '40741',
 '1135',
 '1139',
 '1179',
 '415',
 '39767'
…
```

Детальное описание этого кейса с примером Блокнота приведено в [16].

Второй пример заключается в определении списка подкаталогов в текущем каталоге. Эту задачу с помощью Python можно решить несколькими способами [17], но остановимся только на возможностях, предоставляемых средой Jupyter Lab.

1) Вывести информацию о содержимом текущего рабочего каталога:

```
In [1]: !ls -a ./
.            bash_infosec    log_analysis    scapy
..           elf             networks        traffic-analysis
```

2) Получить содержимое каталога и отфильтровать только подкаталоги:

```
In [2]: import os
        file_list = !ls -a ./
        file_list.grep(os.path.isdir)
['.', '..', 'bash_infosec', 'log_analysis', 'scapy', 'elf, networks', 'traffic-analysis']
```

Представленный способ выполнения команд можно перенести на деятельность специалистов по ИБ. Рассмотрим несколько кейсов, которые демонстрируются в дисциплине «Программно-аппаратные средства защиты информации».

В первом примере Блокноты Jupyter Lab используются для написания YARA-правил [18], служащих одним из форматов индикаторов компрометации [19].

YARA-правило, которое ищет подозрительные строки в любом файле, выглядит следующим образом:

```
rule suspicious_strings
{
strings:
  $a = "Synflooding"
  $b = "Portscanner"
  $c = "Keylogger"
condition:
  ($a or $b or $c)
}
```

В Блокноте [20] приведенное правило (в файле *suspicious_01.yara*) с помощью утилиты *yara* применяется к различным файлам, например:

```
In [34]: !yara -r yara-rules/suspicious_01.yara samples
```

Второй учебный кейс относится к анализу исполняемых PE-файлов и содержится в Блокноте по ссылке [21]. В процессе анализа файла вычисляется его хэш-сумма:

```
In [13]: import hashlib
         content = open("samples/task-1.exe","rb").read()
         print(hashlib.md5(content).hexdigest())
a82a243ff5dbf90677c64eae4f0b6a8e
```

Далее с помощью открытого API сервиса VirusTotal [22] на языке Python выполняется поиск файла в базе данных malware по значению хэш-суммы:

```
In [17]: import requests
         api_url = 'https://www.virustotal.com/vtapi/v2/file/report'
         params = dict(apikey='<key>',
resource='a82a243ff5dbf90677c64eae4f0b6a8e')
         response = requests.get(api_url, params=params)
         if response.status_code == 200:
            result=response.json()
            print(result)
```

```
{'response_code': 0, 'resource': 'a82a243ff5dbf90677c64eae4f0b6a8e',
'verbose_msg': 'The requested resource is not among the finished, queued or
pending scans'}
```

Приведенные Блокноты можно клонировать в собственные github-репозитории для дальнейшего изучения и модификации.

Таким образом, в статье были представлены кейсы по использованию наглядных Блокнотов Jupyter Lab для формирования аналитических навыков у бакалавров ИБ. В качестве перспективы можно отметить возможность построения в Блокнотах моделей машинного обучения для данных, подготовленных учащимися.

Список использованных источников:


1. Федоров Д. Ю., Стельмашонок Е. В. Компетенции Ворлдскиллс, трудовые функции профстандартов и повышение качества образования студентов в области защиты информации // Конвергенция цифровых и материальных миров: экономика, технологии, образование. Сборник научных статей международной научной конференции. 21–22 июня 2018 г. Санкт-Петербург. Conference of St.-Petersburg State University of Economics. / Под ред. проф. В.В. Трофимова, В.Ф. Минакова. – СПб.: Изд-во СПбГЭУ, 2018. – С. 269-273
2. Убийственная цепочка или что такое Kill Chain. Блог Алексея Лукацкого [электронный ресурс]. URL: https://lukatsky.blogspot.com/2016/10/kill-chain.html (дата обращения: 06.10.2020)
3. Как стать SOC-аналитиком. Блог Алексея Лукацкого [электронный ресурс]. URL: https://lukatsky.blogspot.com/2020/01/soc_10.html (дата обращения: 06.10.2020)
4. Официальный сайт Project Jupyter [электронный ресурс]. URL: https://jupyter.org/ (дата обращения: 06.10.2020)
5. Официальный сайт дистрибутива Anaconda [электронный ресурс]. URL: https://www.anaconda.com/ (дата обращения: 06.10.2020)
6. John Lambert. The Githubification of InfoSec. [электронный ресурс]. URL: https://medium.com/@johnlatwc/the-githubification-of-infosec-afbdbfaad1d1 (дата обращения: 06.10.2020)

7. Джон Ламберт. Гитхабификация Информационной Безопасности. [электронный ресурс]. URL: https://habr.com/ru/company/microsoft/blog/487584/ (дата обращения: 06.10.2020)
8. Ian Rose, Grant Nestor. JupyterLab: The Evolution of the Jupyter Notebook [электронный ресурс] URL: https://www.youtube.com/watch?v=NSiPeoDpwuI (дата обращения: 06.10.2020)
9. Официальный сайт Project Jupyter [электронный ресурс]. URL: https://jupyter.org/try (дата обращения: 06.10.2020)
10. Официальный сайт системы Mathematica [электронный ресурс]. URL: https://www.wolfram.com/mathematica/ (дата обращения: 06.10.2020)
11. Jeffrey M. Perkel. Why Jupyter is data scientists' computational notebook of choice. An improved architecture and enthusiastic user base are driving uptake of the open-source web tool. Nature 563, 145-146 (2018) [электронный ресурс]. URL: https://www.nature.com/articles/d41586-018-07196-1 (дата обращения: 06.10.2020)
12. Рейтинг языков программирования TIOBE [электронный ресурс]. URL: https://www.tiobe.com/tiobe-index/ (дата обращения: 06.10.2020)
13. Федоров Д.Ю. Программирование на языке высокого уровня Python : учеб. пособие для прикладного бакалавриата. – 2-е изд., перераб. и доп. – М.: Издательство Юрайт, 2019. – 161 с.
14. Официальная документация Project Jupyter [электронный ресурс]. URL: https://jupyter.readthedocs.io/en/latest/projects/architecture/content-architecture.html (дата обращения: 06.10.2020)
15. Эволюция командной оболочки Python. Персональный блог Дмитрия Федорова [электронный ресурс]. URL: http://blog.dfedorov.spb.ru/all/evolyuciya-komandnoy-obolochki-python/ (дата обращения: 06.10.2020)
16. Получить идентификаторы процессов, принадлежащих пользователю. Персональный блог Дмитрия Федорова [электронный ресурс]. URL: http://blog.dfedorov.spb.ru/all/poluchit-identifikatory-processov-prinadlezhaschih-polzovatelyu/ (дата обращения: 06.10.2020)
17. Определяем подкаталоги в текущем каталоге. Персональный блог Дмитрия Федорова [электронный ресурс]. URL: http://blog.dfedorov.spb.ru/all/opredelyaem-podkatalogi-v-tekuschem-kataloge/ (дата обращения: 06.10.2020)
18. Пишем YARA правила. Персональный блог Дмитрия Федорова [электронный ресурс]. URL: http://blog.dfedorov.spb.ru/all/pishem-yara-pravila/ (дата обращения: 06.10.2020)
19. Индикатор компрометации (IoC). Персональный блог Дмитрия Федорова [электронный ресурс]. URL: http://blog.dfedorov.spb.ru/all/indikator-komprometacii-ioc/ (дата обращения: 06.10.2020)
20. Notebook [электронный ресурс]. URL: https://nbviewer.jupyter.org/github/dm-fedorov/infosec/blob/master/re-tools/yara-%D0%BF%D1%80%D0%B0%D0%B2%D0%B8%D0%BB%D0%B0.ipynb (дата обращения: 06.10.2020)
21. Notebook [электронный ресурс]. URL: https://nbviewer.jupyter.org/github/dm-fedorov/infosec/blob/master/re-tools/hashes%20%D0%B8%20PE-%D1%84%D0%B0%D0%B9%D0%BB%D1%8B.ipynb (дата обращения: 06.10.2020)
22. API Scripts and client libraries [электронный ресурс]. URL: https://support.virustotal.com/hc/en-us/articles/360006819798-API-Scripts-and-client-libraries (дата обращения: 06.10.2020)

## 4.3. Подход к созданию испытательной лаборатории по пентестингу

В процессе подготовки специалистов по информационной безопасности важное значение имеет получение навыков по проведению тестов на проникновение в информационную систему (ИС) с целью обнаружения в ней уязвимостей. Так называемый «пентестинг» – это метод оценки безопасности ИС путем моделирования атак нарушителя. Оценка безопасности может производиться как в рамках активного аудита, так и по непосредственному соглашению между пентестерами и руководством ИС для проведения инструментальных тестов. Тесты на проникновение должны включать, в том числе, выявление уязвимостей сетевого и системного уровня, анализ защищенности беспроводных сетей и анализ защищенности веб-приложений.

Пентестер, или специалист по проникновению в компьютерные системы и сети, должен обладать не только общими знаниями в области информационной безопасности, но, прежде всего, практическими навыками поиска уязвимостей и моделирования атак на проникновение, использующих найденные уязвимости системы. Это предъявляет повышенные требования к подготовке, которая должна включать:

- знание требований регуляторов;
- знание сетевых протоколов и их уязвимостей;
- знание рынка современных сертифицированных средств защиты информации и средств сканирования сети;
- опыт администрирования ОС Windows и UNIX-совместимых систем;
- опыт работы с операционной системой (ОС) Kali Linux и ее сервисами;
- опыт программирования на языках C, Python, PHP, Ruby и др.;
- работа в командных интерпретаторах типа bash и PowerShell;
- и т.д.

Для подготовки такого специалиста необходимо создать практическую возможность осуществления тестов на проникновение в условиях, максимально приближенных к реальным. Такую возможность может предоставить специально созданная испытательная лаборатория. Совершенно очевидно, что и в производственных условиях многие тесты нужно сначала выполнять в лабораторной среде, прежде, чем переносить их в производственную, чтобы избежать возможных проблем. Например, при выполнении тестов на проникновение в рамках традиционных лабораторных практикумов не следует использовать доступные в Интернете уязвимые серверы без письменного разрешения, сканирование портов также может быть воспринято как пассивная атака и т.п.

Созданием подобных испытательных лабораторий и, в широком смысле, киберполигонов, занимаются в сфере ИБ уже не первый год.

За рубежом тематика киберполигонов давно вышла на государственный уровень. При этом создать собственный киберполигон и поддерживать его в актуальном состоянии – наукоемкая и ресурсозатратная задача.

Известен опыт ряда научно-исследовательский лабораторий по созданию киберполигонов [1].

Однако в условиях учебного заведения с ограниченным бюджетом достаточно на первом этапе создать испытательную лабораторию, в которой будут сосредоточены ресурсы (физические, виртуальные или их комбинация) для проведения практикумов по пентестингу. Лаборатория может быть использована также для проведения олимпиад по защите информации в формате CTF, а также для научно-исследовательской и самостоятельной работы студентов.

Известно несколько тестовых лабораторий, которые эмулируют ИТ-инфраструктуру реальных компаний и на базе которых можно повышать навыки тестирования на проникновение в онлайн-режиме.

В частности, Pentestit – российская компания, предоставляющая услуги в области анализа защищенности, в том числе предоставляет доступ к лабораторным машинам по VPN (https://lab.pentestit.ru/).

Другая российская компания Hacktory предлагает «Иммерсивную[2] образовательную платформу по кибербезопасности», имеющую курсы по веб-безопасности и по безопасному программированию на языке Java (https://hacktory.ai/).

Несколько странным выглядит полностью англоязычное описание функционала указанных российских лабораторий, что может стать препятствием для использования их в российских университетах, преподавание в которых ведется на русском языке, даже несмотря на то, что специалисты по ИТ должны владеть техническим английским языком.

Известна английская платформа HackTheBox, на которой свободно размещены стенды как с ОС семейства Unix, так и с Windows. Доступ к стендам происходит путем VPN-туннелей. Есть возможность взаимодействия с университетами со скидками на ряд услуг (https://www.hackthebox.eu/).

Использование перечисленных платформ в конкретном образовательном учреждении затруднительно, так как придется адаптировать образовательные программы под использование сторонних ресурсов. Это может вызвать несоответствие государственным образовательным стандартам по направлению «Информационная безопасность». Ну и, что немаловажно, при подключении в чужую сеть всегда есть вероятность, что вы будете подвергнуты атакам из этой сети на ваш собственный компьютер.

Разработка и постепенное развитие собственной платформы для испытательной лаборатории позволит, во-первых, адаптировать ее под нор-

[2] *Иммерсивность* (от англ. immersive – «создающий эффект присутствия, погружения»)

мативную базу действующих в вузе образовательных программ, во-вторых, создать различные уровни сложности освоения пентестинга, в-третьих, придаст интерес студентам к образовательному процессу, в том числе, в форме гейминга.

Как указывалось, тестирование на проникновение в зависимости от используемых инструментов может быть довольно дорогим. Особенно если использовать коммерческие инструменты. Но, учитывая множество доступных в Kali Linux программ с открытым исходным кодом, без коммерческих инструментов можно обойтись.

При создании лабораторной среды необходимо для каждой операционной системы, включая основную ОС и все виртуальные машины, соблюсти хотя бы минимальные рекомендуемые требования. Для комфортной работы без ошибок, связанных с недостатком оперативной памяти, было бы правильно иметь запас оперативной памяти больше рекомендуемого. Учитывая, что большинство ОС на базе Linux, требуют порядка 2 Гбайт оперативной памяти, выполнить это требование возможно.

Гипервизор для среды виртуализации может быть в принципе любым, но предпочтительнее использовать бесплатные версии:

- vSphere Hypervisor, созданный на базе VMware vSphere ESXi, компактной и устойчивой архитектуры для виртуализации серверов и консолидации приложений при управлении ИТ-инфраструктурой;
- VirtualBox, дружественный, хорошо развитый и постоянно обновляемый гипервизор от компании Oracle.

В качестве хостовой операционной системы может быть использована одна из последних ОС от Microsoft: Windows 10 Enterprise.

Поскольку в среде гипервизора должно быть установлено несколько уязвимых серверов, память физического сервера должна быть не менее 24 Гбайт, из которых порядка 6 Гбайт отводится под хостовую систему.

Уязвимые серверы должны быть сконфигурированы таким образом, чтобы позволить решать следующие задачи пентестинга:

- тесты на уязвимости WEB-серверов;
- тесты на уязвимости приложений;
- тесты на проникновение в сеть;
- тесты на проникновение в беспроводную сеть;
- тесты на уязвимости мобильных приложений.

В качестве модульной платформы для тестирования на проникновение может использоваться Metasploit Framework, которая позволяет писать, тестировать и выполнять код эксплойта. Metasploit Framework содержит набор инструментов, которые можно использовать для тестирования уязвимостей безопасности, выполнения атак и уклонения от обнаружения [2]. По своей сути Framework представляет собой набор часто используемых инструментов, которые предоставляют полную среду для тестирования на проникновение и разработки эксплойтов (рис. 4.3). Это очень мощный инструмент,

который может использоваться пентестерами (но, к сожалению, и киберпреступниками), для выявления систематических уязвимостей в сетях и серверах. Поскольку это среда с открытым исходным кодом, ее можно легко настроить и использовать с большинством операционных систем. Часть инструментов Metasploit встроена в ОС Kali Linux.

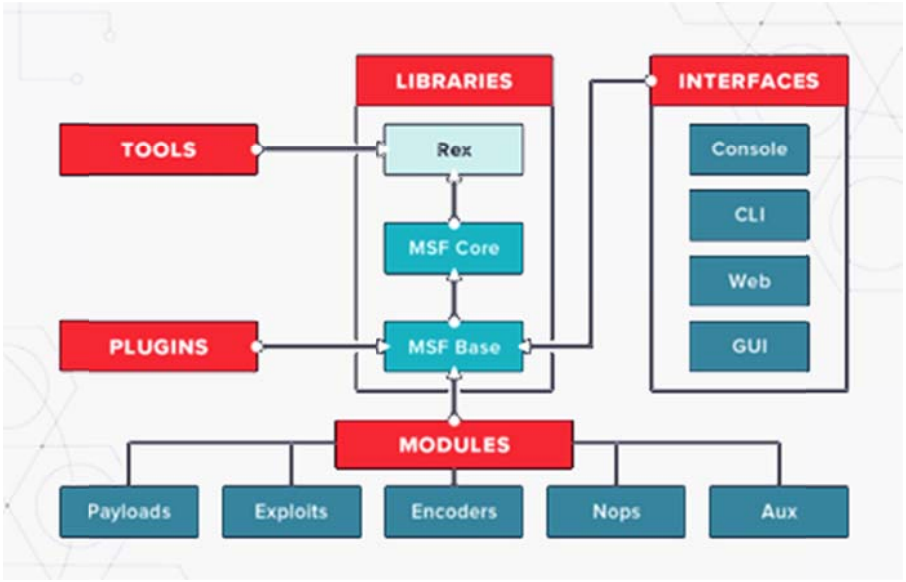



Рисунок 4.3 – Архитектура Metasploit

Metasploit Framework состоит из таких модулей, как: эксплойты, полезные нагрузки и сетевые сканеры, что позволяет сочетать их друг с другом (рис. 4.4).

Metasploit включает в себя более 1677 эксплойтов, организованных на 25 платформах, включая Android, PHP, Python, Java, Cisco и многие другие [3]. База данных эксплойтов содержит всю информацию о доступных эксплойтах, открытых портах, службах и данные проверяемого хоста. С помощью нее можно получить эксплойты, которые подойдут для конкретного сканируемого веб-приложения. Также она позволяет узнать номер уязвимости, используемую определенным эксплойтом и все основные данные о ней.

Структура также содержит около 500 полезных нагрузок:

- полезные нагрузки командной оболочки, которые позволяют пользователям запускать сценарии или произвольные команды для хоста;
- динамические полезные нагрузки, которые позволяют пентестерам генерировать уникальные полезные нагрузки для обхода антивирусного программного обеспечения;
- полезные нагрузки Meterpreter, которые позволяют пользователям контролировать устройства при пост-эксплуатации системы в случае удачного запуска эксплойта;

– статические полезные нагрузки, обеспечивающие переадресацию портов и обмен данными между сетями.

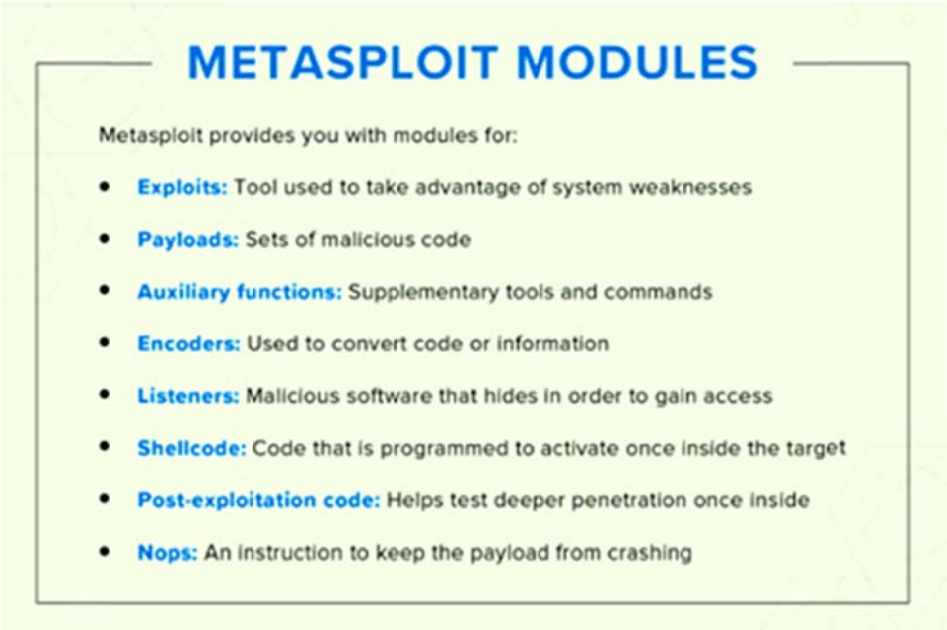



Рисунок 4.4 – Модули Metasploit Framework

Пентестер, формируя поисковые запросы Google вида «сервис версия» +vulnerability +exploit, находит страницы с описанием уязвимостей и эксплойтов [4].

Известные базы уязвимостей [5]:

– www.exploit-db.com;
– www.rapid7.com/db;
– nvd.nist.gov;
– www.cvedetails.com.

Для эксплуатации уязвимостей в сетевых сервисах и прикладном ПО используются эксплойты из раздела exploit Metasploit Framework. Подходящие эксплойты можно найти с помощью команды search по коду CVE, названию или версии сервиса.

В качестве уязвимых серверов могут использоваться как хорошо известные системы, например, Metasploitable 2, Metasploitable 3, BadStore, так и серверы собственной разработки.

На первом этапе подготовки испытательной лаборатории был разработан уязвимый сервер, на котором развернуты ssh, ftp, веб-сервер с приложением для сохранения заметок, веб-сервер с приложением для мониторинга. Работа с сервером предполагает два варианта:

– воспользоваться уязвимостями в одном из веб-приложений для получения исполнения команд на стороне сервера от непривилегированного пользователя;

– найти способ получения исполнения команд от суперпользователя root.

В качестве операционной системы, на которой развертывалось все программное обеспечение использована ОС Ubuntu Server 20.04. Она поставляется без графической оболочки, что существенно уменьшает размер занимаемого дискового пространства и количество необходимой оперативной памяти.

К стенду осуществляется доступ по протоколу ssh, который предоставляет доступ к консольной оболочке bash.

На порт 21 дан анонимный доступ к размещенным на стенде файлам. Анонимный доступ подразумевает вход в систему без использования пароля. В размещенных на ftp-сервере файлах можно найти некоторые файлы исходного кода веб-приложения, работающего на порту 80. Данные файлы могут оказаться полезными пользователям стенда для дальнейшей атаки на веб-сервер.

На порту 80 располагается веб-приложение, написанное на языке программирования Python, в котором оставлены две уязвимости веб-приложений – SSTI и XSS.

Уязвимость SSTI – server side template injection (переводится как инъекция в шаблоны на стороне сервера) позволяет с помощью специально сконструированной строки символов исполнять код на стороне сервера посредством внедрения строки в шаблон, который возвращает данные на страницу веб-приложения. С помощью этой уязвимости пользователь стенда может прочитать или создать файлы на сервере и многое другое.

Уязвимость XSS – cross-site scripting (переводится как межсайтовый скриптинг) позволяет исполнять javascript код в браузере пользователя, что чревато кражей cookie-файлов пользователя, которые позволяют авторизоваться в приложении в качестве атакованного пользователя. В данном стенде эта уязвимость может быть проэксплуатирована на пользователе admin, имеющем доступ к конфиденциальной информации.

На порту 8089 доступно веб-приложение, позволяющее контролировать работу сервера или отдельных приложений. Оно содержит опубликованную уязвимость, код для эксплуатации которой пользователь может найти в сети Интернет.

В качестве перспектив развития данного сервера можно указать возможность контейнеризации компонентов стенда с использованием ПО docker, kubernetes, что даст возможность облегчить развертывание стенда в обновляемых системах.

Отметим, что в описании каждого уязвимого сервера, помимо описания самих уязвимостей, должно быть описание методики эксплуатации этих уязвимостей с тем, чтобы после этапа гейминга студент мог иметь возможность провести самостоятельный разбор хода пентестинга и этим завершить этап обучения.

Установка уязвимых виртуальных серверов испытательной лаборатории предполагается на физических серверах университета. Однако последние события с пандемией коронавируса показали необходимость реализации удаленного доступа к данному сервису. Это может быть выполнено при развертывании университетского корпоративного облака или с использованием российских облачных сервисов, например, Yandex.Cloud.

Список использованных источников:

1. Хочешь мира – готовься к войне или зачем нужен киберполигон Анна Олейникова, ведущий аналитик «Ростелеком-Солар» [электронный ресурс]. URL: https://www.securityl ab.ru/analytics/512874.php (дата обращения: 14.10.2020).
2. Тест на проникновение с помощью Metasploit Framework: базовое руководство для системного администратора [электронный ресурс]. URL: https://habr.com/ru/company/echelon/blog/347702/ (дата обращения: 14.10.2020).
3. What is Metasploit? The Beginner's Guide [электронный ресурс]. URL: https://www.varonis.com/blog/what-is-metasploit/ (дата обращения: 14.10.2020).
4. Vulners – Гугл для хакера. Как устроен лучший поисковик по уязвимостям и как им пользоваться [электронный ресурс]. URL: https://xakep.ru/2016/07/08/vulners/ (дата обращения: 14.10.2020).
5. Общий обзор реестров и классификаций уязвимостей (CVE, OSVDB, NVD, Secunia) [электронный ресурс]. URL: https://safe-surf.ru/specialists/article/5228/607311/ (дата обращения: 14.10.2020).

## 4.4. Контрольная точка с элементами деловой игры

В данном параграфе излагаются особенности методики преподавания дисциплин «Информационная безопасность» и «Защита информации» для студентов гуманитарного и юридического факультета.

Студенты нашего университета изучают множество различных дисциплин, как специальных, так и общеобразовательных. Среди них важное место занимают дисциплины «Информационная безопасность» и «Защита информации». Они преподаются, как специалистам, так и студентам некоторых направлений «непрофильных» факультетов (гуманитарный, юридический и др.). Особенность преподавания этих дисциплин на вышеназванных факультетах заключается в том, что студенты исходно имеют очень низкий уровень математической подготовки.

На лекции был проведен эксперимент по расшифрованию зашифрованного с помощью шифра простой замены (шифр Цезаря) [1] латинского изречения (рис. 4.5). Процесс расшифрования прошел с большим трудом, хотя для облегчения процесса в шифрованном сообщении была даже оставлена запятая, разделяющая две части фразы.

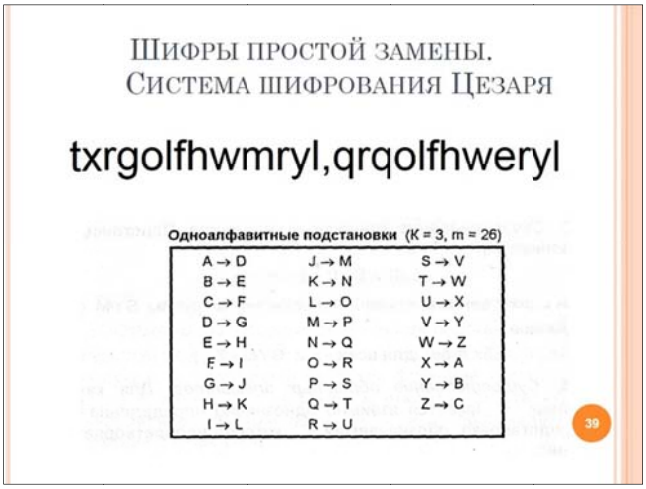



Рисунок 4.5 – Шифр Цезаря

Наконец, расшифрование было закончено и полученное сообщение разделено на слова [2]:

Quod licet Jovi, non licet bovi.

Следующим «камнем преткновения» оказалось то, что студенты не знают этого изречения и не могут его перевести, причем даже юристы, у которых изучение основ латыни входит в учебный план.

В свете проведенного эксперимента встал вопрос о практической части и контрольных точках, которые необходимо выполнить студентам по рабочей программе дисциплины. И если лабораторный практикум содержит блок лабораторных работ, которые могут быть выполнены вышеозначенным контингентом студентов, то контрольные точки.

Проблему первой контрольной точки можно решить написанием эссе на заданную тему. Найти информацию в интернете, обработать ее и представить в требуемом формате гуманитарий должен уметь.

Вторая контрольная точка должна быть посвящена шифрованию. Шифр Цезаря уже использовался. Появилась мысль использовать одно из направлений цифровой стеганографии. Сокрытие текста в изображении семантически очень достоверно демонстрирует суть современных способов скрытия информации. Но доступные бесплатные ресурсы не отвечают даже самым скромным запросам. Соединение текста и картинки превращает последнюю в набор пикселей, в котором с трудом угадывается исходное изображение. Кроме того, на этих ресурсах велика вероятность «подхватить» вредоносную программу.

Результатом кропотливого поиска решения, стала идея использовать один из методов симметричного шифрования.

Алгоритмы симметричного шифрования основаны на том, что и для шифрования сообщения, и для его расшифровки используется один и тот же (общий) ключ (рисунок 4.6).

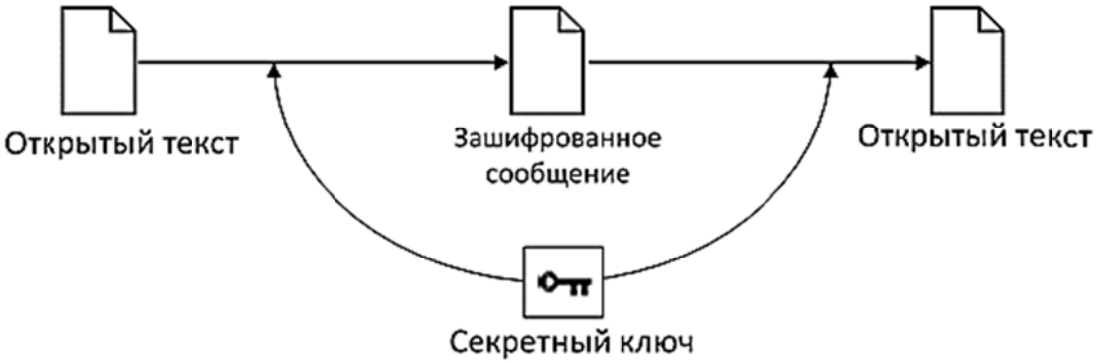



Рисунок 4.6 – Симметричная схема шифрования

Одно из главных преимуществ симметричных методов – быстрота шифрования и расшифровки, а главный недостаток – необходимость передачи секретного значения ключа получателю. Неизбежно возникает проблема: как передать ключ и при этом не позволить злоумышленникам перехватить его.

Преимущества криптографии с симметричными ключами:

– *Высокая производительность*.
– *Высокая стойкость*. При прочих равных условиях стойкость криптографического алгоритма определяется длиной ключа. При длине ключа 256 бит необходимо произвести $10^{77}$ переборов для его определения.

Недостатки криптографии с симметричными ключами:

– *Проблема распределения ключей*. Так как для шифрования и расшифровки используется один и тот же ключ, требуются очень надежные механизмы для их распределения (передачи).
– *Масштабируемость*. Так как и отправитель, и получатель используют единый ключ, количество необходимых ключей возрастает в геометрической прогрессии в зависимости от числа участников коммуникации.
– *Ограниченное использование*. Криптография с секретным ключом используется для шифрования данных и ограничения доступа к ним, с ее помощью невозможно обеспечить такие свойства информации, как аутентичность и неотрекаемость [3].

В лабораторном практикуме есть лабораторная работа «Изучение шифра «Поворотные решетки Кардано» [4], в которой наглядно демонстрируются возможности симметричного шифрования.

Решетка Кардано – это прямоугольная или квадратная карточка с четным числом строк и столбцов $2k \times 2m$. В ней проделаны отверстия (трафарет) таким образом, что при последовательном отражении или поворачивании и заполнении открытых клеток карточки постепенно будут заполнены все клетки листа.

Получатель должен знать трафарет и наложить его в той же последовательности, что и при шифровании [4].

Осталось придумать под каким соусом это блюдо подать. А почему бы не поиграть в «шпионов»? Обмен шифрованными сообщениями между партнерами с учетом особенностей присущих алгоритму симметричного шифрования, использование различных каналов связи для передачи различных частей сообщения, отчет о проделанной работе. Всего этого достаточно для того, чтобы контрольная точка превратилась в тренинг по работе с шифрованными сообщениями.

Для изучения и тренировки студентам рекомендуется самостоятельно выполнить лабораторную работу «Изучение шифра «Поворотные решетки Кардано». После выполнения лабораторной работы, каждому студенту в группе предлагается выбрать себе vis a vis, с которыми он будет обмениваться шифрованными сообщениями.

Студент должен придумать текст сообщения (если текст больше возможностей шифратора, возможно частичное купирование текста, не влияющее на смысловое содержание) и произвести шифрование сообщения.

Шифрованное сообщение пересылается vis a vis по электронной почте (имитирует слабозащищенный канал передачи). Кроме того, vis a vis получает ключ расшифрования, отправляемый по SMS (защищенный канал связи), примерно такого содержания: «*Сов. Секретно: 2,4,8…, +/-, cup*».
Где:

- 2,4,8… – расположение клеток трафарета;
- +/- – направление вращения трафарета;
- cup – признак купированного сообщения (опционально).

Такое же сообщение и SMS студент получает от vis a vis.

Результат выполнения контрольной точки оформляется в виде отчета, который присылается преподавателю и включает:

- результаты выполнения лабораторной работы (шифруемая фраза, шифрованная фраза, решетка кардано в начальном положении),
- придуманная студентом фраза, процесс шифрования, шифрованная фраза, sms с ключом.
- полученная от vis a vis шифрованная фраза, SMS с ключом, процесс расшифрования и результат расшифрования.

Если группа хорошо воспринимает и усваивает материал, группе может быть предложен расширенный вариант контрольной точки, включающий элементы деловой игры. Суть игры состоит в том, что из группы выделяется (назначается) студент, который будет заниматься дешифровкой «перехваченных» сообщений. Перехват сообщений имитируется добавлением электронного адреса студента, назначенного «дешифровщиком» (decipherer) (рисунок 4.7)

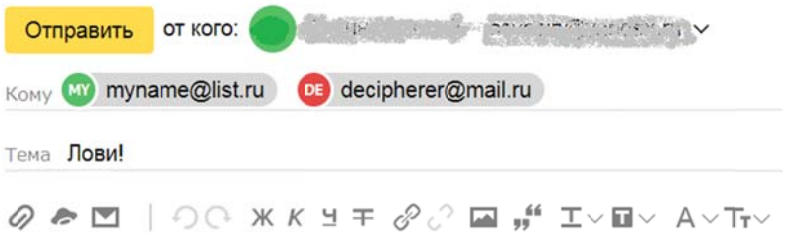



Рисунок 4.7 – Имитация «перехвата» сообщения

Дешифровщик получает только шифрованный текст. Он не знает ключа, который применялся при шифровании (объяснить студентам, что ключ «контрразведчику» присылать нельзя). Он знает только метод шифрования. Зная правила зашифрования сообщения с помощью решетки Кардано, дешифровщик должен правильно расшифровать как можно большее количество перехваченных сообщений. Результаты дешифрования «контрразведчик» присылает преподавателю, указав электронные адреса «перехвата» и дешифрованные сообщения. Это и будет его отчет по контрольной точке. Дешифровщиков может быть несколько. Тогда они могут перераспределять нагрузку между собой, обмениваться сообщениями, кооперироваться для выполнения работы.

Подобная организация изучения процесса шифрования/ расшифрования/ дешифрования будет способствовать выработке навыка работы с документами, содержащими конфиденциальную информацию.

Список использованных источников:

1. Васильева И.Н. Криптографические методы защиты информации. Учебник и практикум. Сер. 58 Бакалавр. Академический курс (1-е изд.) – Москва: Изд-во Юрайт, 2019. – 348 с.
2. Латинские пословицы и крылатые выражения с транскрипцией (транслитерацией) и ударениями [электронный ресурс]. URL: http://graecolatini.bsu.by/htm-proverbs/proverbs-latin-transcription-170-ru.htm (дата обращения: 15.10.2020).
3. Васильева И.Н. Защита информации: учебное пособие / И. Н. Васильева; М-во образования и науки Российской Федерации, Гос. образовательное учреждение высш. проф. образования «Санкт-Петербургский гос. инженерно-экономический ун-т». Санкт-Петербург, 2011. – 162 с.
4. Защита информации: Методические указания к изучению дисциплины и выполнению контрольной работы для студентов заочной формы обучения направлений подготовки 030900 Юриспруденция, 031900 Международные отношения, 080200 Менеджмент (без профиля), квалификация - бакалавр. Файл 11887.doc / Сост.: И.Н.Васильева. – СПб: СПбГИЭУ, 2012 – 76 с.

## Заключение

Монография посвящена вопросам обеспечения информационной безопасности и защиты информации при внедрении новых цифровых технологий. Важным аспектом реагирования на угрозы в информационной сфере является выявление деструктивных воздействий и злоупотреблений в компьютерных системах. Рассмотрены подходы к выявлению недекларированных возможностей программного кода и модели анализа аномалий сетевого трафика.

Проведен анализ отдельных технологий и связанных с ними проблем безопасности и средств защиты информации, в частности, NoSQL базы данных, платформа разработки веб приложений Node.js, протоколы криптографической защиты DNS, протоколы цифровых валют. Рассмотрены средства аутентификации, стеганографии, криптографической защиты информации, а также подходы к обеспечению физической безопасности информационных систем.

Изложены методические рекомендации по построению и оценке экономической эффективности системы защиты информации в рамках отдельных хозяйствующих субъектов. Особое внимание уделено соблюдению требований российских и европейских регуляторов по обеспечению безопасности персональных данных. Рассмотрены гуманитарные аспекты информационной безопасности.

В заключение рассмотрены проблемы безопасности при организации учебного процесса с применением дистанционных образовательных технологий, а также изложен опыт использования активных методов обучения по вопросам информационной безопасности и защиты информации.

Монография может быть полезна преподавателям, студентам, магистрантам, аспирантам и специалистам в области компьютерных систем, информационной безопасности и защиты информации, а также всем, кто интересуется вопросами внедрения цифровых технологий.

Научное издание

**ЦИФРОВЫЕ ТЕХНОЛОГИИ**
**И ПРОБЛЕМЫ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ**

Под редакцией Е.В. Стельмашонок, И.Н. Васильевой

Подписано в печать 17.02.2021. Формат 60×84 1/16.
Усл. печ. л. 9,75. Тираж 500 экз. Заказ 893.

Издательство СПбГЭУ. 191023, Санкт-Петербург, Садовая ул., д. 21.

Отпечатано на полиграфической базе СПбГЭУ