МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение высшего образования
«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ ГИДРОМЕТЕОРОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра Информационных технологий и систем безопасности

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
(дипломная работа)

На тему «Разработка модели защиты информации от несанкционированного доступа в локальных сетях»

Исполнитель: Погудин Савелий Андреевич
(фамилия,имя,отчество)

Руководитель: Бурлов Вячеслав Георгиевич
(фамилия, имя, отчество)

«К защите допускаю»
Заведующий
кафедрой_____
(подпись)

_____
(ученая степень,ученое звание)

_____ Бурлов В.Г. _____
(фамилия,имя,отчество)

«» 25г

Санкт-Петербург
2025

РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ
ГИДРОМЕТЕОРОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»
Кафедра Информационных технологий и систем безопасности

«УТВЕРЖДАЮ»
Заведующий кафедрой
Бурлов В.Г.
(подпись) (фамилия, имя,отчество)
« _ «_____2025 года

Задание
на выпускную квалификационную работу

_____
(фамилия, имя, отчество)

1. **Тема** Разработка модели защиты информации от НСД в локальных сетях _____

закреплена приказом ректора Университета от «___» _____2025 года, № ____

2. Срок сдачи законченной работы «___» _____2025 года

3.Исходные данные к выпускной квалификационной работе:
«Функционирование ЛВС», «Анализ существующих угроз» «Методы обеспечения информационной безопасности «, «Проектирование ресурса», «Методические рекомендации по улучшению системы безопасности ЛВС»_____

Перечень вопросов, подлежащих разработке (краткое содержание работы(проекта):

- Введение. Актуальность темы, цели и задачи выпускной квалификационной работы.
- Глава1. Теоретическая часть.
  (наименование главы)
- Глава2. Аналитическая часть.
  (наименование главы)
- Глава3. Практическая часть.
  (наименование главы)
- Заключение. Выводы по работе.
  (наименование главы)

4.Перечень материалов, представляемых к защите:

- Пояснительная записка;

5.Дата выдачи задания:»_____«_____202 года
Руководитель выпускной квалификационной работы

(должность, ученая степень, ученое звание, фамилия,имя,отчество) (подпись)

Задание принял к исполнению «_____«_____2025 года

Студент: Погудин Савелий Андреевич, ИБ-С20-1 __
(фамилия, имя, отчество ,учебная группа) (подпись)

ВВЕДЕНИЕ

0.1. Основные тенденции развития информационных технологий и их влияние на обеспечение безопасности локальных сетей

В условиях интенсивного развития информационных технологий и расширения масштабов локальных сетей (ЛВС) возрастает потребность в обеспечении их информационной безопасности. Основные факторы, формирующие потребности в защите данных, включают рост числа угроз (НСД, кибератаки, вредоносное ПО), усложнение сетевой инфраструктуры и увеличение интенсивности внутреннего обмена информацией. Эти процессы требуют разработки системных моделей защиты, способных адаптироваться к современным вызовам, обеспечивая устойчивость и безопасность корпоративных и государственных систем.

Рассматриваются и выявляются основные механизмы обеспечения безопасности, связанные с анализом угроз, автоматизацией обнаружения и нейтрализацией атак, а также оценкой уязвимых элементов инфраструктуры.

Данная совокупность факторов определяет актуальность настоящей работы, целью которой является «Выбрать, обосновать и реализовать условия обеспечения безопасности локальных сетей в условиях современных угроз на основе использования математического аппарата, технологий и средств информационной безопасности».

0.2. Постановка задачи исследования в целом

Для достижения поставленной цели в работе сформулирована и решена задача разработки модели системы защиты информации в локальных сетях, которая учитывает особенности угроз, механизмы их появления и методы противодействия. В рамках исследования необходимо провести анализ существующих методов, моделей и технологий обеспечения безопасности, выявить их слабые стороны и разработать новые подходы к управлению процессами защиты.

Важная научная задача:

Дано – локальная сеть.

Требуется - разработать математическую модель системы защиты, которая обеспечивает автоматическое обнаружение, идентификацию и нейтрализацию угроз, а также интегрировать процессы обеспечения безопасности в единую систему.

Возникшие трудности:

- Анализ сложности угроз и разнообразия методов атак;
- Формализация процессов защиты на базе математического аппарата;
- Интеграция различных методов и средств защиты в единую модель.

Трудности преодолены за счет применения системного анализа, моделирования угроз и разработки методов автоматизированного управления.

Решение поставленной задачи требует ее декомпозиции на четыре подзадачи:

1. Проанализировать существующие методы, модели, технологии и особенности обеспечения безопасности локальных сетей на основе математического аппарата, технологий и средств защиты.

2. Разработать модель системной интеграции процессов обнаружения, идентификации и нейтрализации угроз в локальных сетях.

3. Разработать технологию управления процессами обеспечения информационной безопасности.

4. Предложить меры по совершенствованию системы защиты, учитывая новые угрозы и возможности технологий.

0.3. Характеристика подзадач

- Первая подзадача заключается в анализе существующих методов и технологий защиты, выявлении их преимуществ и недостатков, а также возможностей их применения в современных условиях.

- вторая подзадача состоит в разработке математической модели системной интеграции процессов обнаружения и нейтрализации угроз, которая позволяет автоматизировать управление безопасностью.

- третья подзадача включает создание технологии управления процессами защиты.

- четвертая подзадача предполагает разработку предложений по повышению эффективности системы защиты и расширению ее функциональных возможностей.

0.4. Состояние вопроса

Проводится анализ основных подходов к решению задачи обеспечения безопасности локальных сетей. Выделяются направления:

- «Традиционные методы», основанные на статических правилах и сигнатурах;

- «Динамические модели», использующие поведенческий анализ и машинное обучение;

- «Интеграционные подходы», объединяющие обнаружение и реагирование в единую систему.

Указывается, что традиционные методы не полностью решают задачу по свойствам «адаптивности» и «скорости реакции», динамические - по «точности» и «масштабируемости», а интеграционные - по «эффективности» и «устойчивости». Поэтому в работе предлагается системная модель, объединяющая сильные стороны указанных подходов на основе математического аппарата, что позволит повысить уровень защиты в локальных сетях.

0.5. Содержание ВКР

Для решения поставленной задачи в рамках предлагаемого подхода содержание работы включает:

- Введение, в котором изложены актуальность, цели и задачи исследования;

- первый раздел посвящен анализу существующих методов и технологий защиты информации в локальных сетях;

- второй раздел содержит разработку математической модели системной интеграции процессов обеспечения безопасности;

- третий раздел посвящен разработке технологии управления процессами защиты;

- четвертый раздел включает предложения по совершенствованию системы защиты и рекомендации по их внедрению;

- в заключении сформулированы основные выводы и направления дальнейших исследований.

0.6. Основные положения, выносимые на защиту

1. Постановка и формализация задачи создания системы защиты информации в локальных сетях на основе математического моделирования и системной интеграции процессов.

2. Разработка модели системной интеграции процессов обнаружения, идентификации и нейтрализации угроз, реализуемой на базе математического аппарата и технологий информационной безопасности.

3. Создание технологии управления процессами обеспечения безопасности.

4. Предложения по совершенствованию системы защиты с учетом современных угроз и возможностей технологий.

ГЛАВА 1. НАУЧНО-ТЕХНИЧЕСКИЕ ОСНОВЫ ОБЕСПЕЧЕНИЯ БЕЗОПАСНОСТИ ИНФОРМАЦИИ ОТ НЕСАНКЦИОНИРОВАННОГО ДОСТУПА В ЛОКАЛЬНЫХ СЕТЯХ

Информационная среда - это совокупность процессов и средств, реализуемых внутри локальной сети для обеспечения безопасного хранения, передачи и обработки конфиденциальных данных. В современных условиях развитие локальных сетевых инфраструктур и их интеграция с внешними системами требуют внедрения эффективных моделей защиты информации от НСД. Рынок информационных технологий сталкивается с многочисленными угрозами, вызывающими деструктивные воздействия на безопасность локальных сетей, что требует системного и научно обоснованного подхода к обеспечению защиты.

Для использования информационных ресурсов в условиях угроз необходимо уметь моделировать процессы защиты с заранее заданными свойствами [1]. В основе разработки таких моделей лежит системный подход, разработанный научной школой «Системная интеграция процессов управления». Эта школа зарегистрирована в Реестре ведущих научно-образовательных школ Правительства СПБ [2-3].

Процесс защиты информации в локальных сетях строится на системной интеграции следующих элементов:

- целевого информационного процесса;
- процесса формирования угроз;
- процесса идентификации угроз;
- процесса нейтрализации угроз;
- показателя информационной безопасности.

Показателем безопасности является вероятность того, что каждая угроза, связанная с несанкционированным доступом, будет точно идентифицирована и нейтрализована в рамках конкретного информационного процесса.

Технология обеспечения защиты информации от НСД основывается на следующих принципах:

- определении требуемого уровня защиты;

- моделировании угроз, характерных для локальной сети;

- формировании методов аутентификации, авторизации и шифрования;

- внедрении систем мониторинга и обнаружения попыток несанкционированного доступа.

В рамках системной интеграции формируется условие существования модели защиты, выраженное уравнением с двумя неизвестными:

- обобщенной характеристикой процесса идентификации угроз;

- обобщенной характеристикой процесса нейтрализации угроз.

Формирование этих характеристик исходя из заданного уровня защиты позволяет обеспечить выполнение условий модели и достичь необходимого уровня информационной безопасности. Такой подход позволяет разработать аналитическую математическую модель, которая служит основой для автоматизированных систем защиты в локальных сетях, обеспечивая своевременное обнаружение и предотвращение НСД.

При проектировании системы защиты в первую очередь следует обеспечить надежную организацию механизмов аутентификации и контроля доступа к ресурсам сети. Важными элементами являются внедрение криптографических методов защиты каналов передачи данных, систем обнаружения вторжений и постоянный мониторинг состояния сети для своевременного реагирования на попытки несанкционированного проникновения.

1.1. Процесс функционирования информационной системы и обеспечение защиты от НСД в локальных сетях

В современных организациях основой информационной инфраструктуры является локальная сеть (ЛВС), в которой осуществляется обмен данными между различными устройствами - серверами, рабочими станциями, принтерами и

другими устройствами. Для безопасного функционирования системы критически важны не только грамотная организация взаимодействия ее компонентов, но и создание эффективной модели, предотвращающей несанкционированный доступ (НСД) к информации.

Основу такой модели в ЛВС составляют различные механизмы проверки подлинности (аутентификации) и предоставления прав (авторизации) пользователей, средства шифрования данных и системы контроля доступа. Каждый пользователь при подключении проходит процедуру идентификации, подтверждающую его право на доступ к конкретным сетевым ресурсам. Это является фундаментом для ограничения НСД и обеспечения конфиденциальности информации.

Ключевым элементом безопасности выступает управление доступом к ресурсам сети. Для этого используются специализированные системы контроля доступа (Access Control Systems). Они, основываясь на заданных политиках безопасности, определяют и регулируют уровень привилегий для каждого пользователя или устройства. К примеру, сотрудники IT-отдела могут обладать полным доступом к серверному оборудованию, в то время как рядовые пользователи - лишь к определенным общим папкам или принтерам.

Технические механизмы защиты внедряются как на сетевом уровне, так и на уровне отдельных устройств. В пределах локальной сети активно применяются межсетевые экраны (файрволы), системы обнаружения и предотвращения вторжений (IDS/IPS), а также технологии шифрования для защиты данных во время передачи. Эти меры позволяют существенно снизить риски НСД, даже в случае проникновения злоумышленника в сеть.

Для повышения надежности также используются методы сегментации сети, такие как создание виртуальных локальных сетей (VLAN) и разделение административных и пользовательских сегментов. Это затрудняет горизонтальное перемещение злоумышленника и сдерживает распространение угрозы внутри организации. Дополнительными важными мерами являются

регулярное обновление программного обеспечения, постоянный мониторинг сетевой активности и детальный аудит происходящих событий.

Особая роль отводится разработке и внедрению комплексной модели безопасности. Она должна включать в себя регулярное выявление потенциальных уязвимостей, внедрение современных методов шифрования, строгой аутентификации и многофакторной проверки, а также постоянное обучение персонала основам информационной безопасности. Такой подход позволяет не только предотвращать попытки НСД, но и оперативно реагировать на возможные инциденты безопасности.

Таким образом, эффективная модель защиты информации от НСД в локальных сетях требует интегрированного подхода. Он должен гармонично сочетать в себе современные технические средства, продуманные организационные меры и непрерывные процессы мониторинга. Только такой комплекс способен обеспечить устойчивую безопасность данных и инфраструктуры в условиях постоянно эволюционирующих угроз.

1.2. Анализ угроз для обеспечения защиты информации от НСД в локальных сетях

Как и любые информационные системы, локальные вычислительные сети (ЛВС) обладают определенными уязвимостями и подвержены разнообразным угрозам, создающим риск несанкционированного доступа (НСД). Создание эффективной защитной модели требует обязательного выявления и всестороннего анализа потенциальных угроз для принятия своевременных мер по их нейтрализации.

К основным угрозам и уязвимостям в среде ЛВС относятся:

- внутренние угрозы: Действия злоумышленников или недобросовестных сотрудников, уже имеющих легитимный доступ. Они могут использовать свои права для незаконного получения конфиденциальных данных, изменения критически важных настроек или саботирования работы систем.

- внешние угрозы: Атаки злоумышленников извне организации, которые эксплуатируют уязвимости в сетевом оборудовании или программном обеспечении для проникновения в периметр сети и получения НСД к ее ресурсам.

- уязвимости оборудования и ПО: Ошибки в конфигурации маршрутизаторов, коммутаторов, серверов, а также программные баги в приложениях и операционных системах, которые могут быть использованы для обхода стандартных средств защиты.

- атаки социальной инженерии: Методы психологического манипулирования персоналом, такие как фишинг, вишинг или претекстинг, с целью получения учетных данных, распространения вредоносного ПО или обманом спровоцировав на опасные действия.

- физический доступ: Несанкционированное проникновение в помещения, где размещено сетевое оборудование или серверы, что позволяет напрямую подключиться к инфраструктуре, похитить или повредить аппаратные компоненты.

При разработке модели защиты необходимо учитывать следующие ключевые виды угроз:

- некорректная конфигурация безопасности: Использование слабых или стандартных паролей, отсутствие сегментации сети, несвоевременное обновление и установка исправлений (патчей) для системного и прикладного ПО.

- НСД к сетевым ресурсам: Компрометация и использование украденных учетных данных, а также злоупотребление избыточными привилегиями пользователей.

- вредоносное программное обеспечение: Вирусы, черви, троянские программы, шпионское и ransomware-ПО, предназначенные для кражи, повреждения или блокирования данных.

- атаки на сетевые протоколы и сервисы: Эксплуатация уязвимостей в таких протоколах, как SMB, RDP, SNMP и других, для получения контроля над системами или перехвата трафика.

- DDoS: организованные атаки, приводящие к отказу в обслуживании сетевых ресурсов или целых сегментов инфраструктуры путем создания чрезмерной нагрузки.

- ошибки в настройке политик безопасности: неполное или противоречивое определение правил доступа, аутентификации и аудита, создающее бреши в защите.

1.2.1. Вредоносное программное обеспечение

Вредоносное программное обеспечение (ВПО) представляет собой специально созданные программы, основной целью которых является нанесение ущерба информационной системе, получение несанкционированного доступа или хищение конфиденциальных данных.

Виды вредоносного ПО

Вирусы: программы, способные к самокопированию и внедрению в другие файлы или системы, что приводит к их повреждению, сбоям в функционировании и утечкам информации.

- троянские программы (трояны): вредоносные приложения, маскирующиеся под легитимный софт. Их задача - создать скрытые бэкдоры (лазейки) для проникновения злоумышленников в систему и получения контроля над ней.

- сетевые черви: автономные вредоносные программы, которые распространяются по сети без необходимости действий пользователя, часто перегружая каналы связи, выводя из строя сервисы или повреждая данные.

- шпионское ПО: программы, скрытно собирающие информацию о действиях пользователей, их персональные данные, учетные записи и пароли с последующей передачей злоумышленникам.

- рекламное ПО: навязчивое программное обеспечение, демонстрирующее нежелательную рекламу. Помимо раздражающего воздействия, оно может замедлять работу системы и служить вектором для загрузки более опасных угроз.

- программы-вымогатели: вредоносное ПО, которое шифрует файлы на зараженных компьютерах или блокирует доступ к системе, требуя выкуп за восстановление данных, что парализует бизнес-процессы.

Способы проникновения ВПО в локальные сети

- эксплуатация уязвимостей: использование ошибок и слабых мест в операционных системах, прикладном программном обеспечении или конфигурации сетевого оборудования для тайной установки вредоносного кода.

- физический носитель: занесение ВПО через подключение инфицированных внешних устройств хранения данных (USB-флешки, внешние диски) или при прямом физическом доступе к компьютерам.

- удаленные подключения: использование слабых или украденных учетных данных, а также уязвимых служб удаленного доступа (например, RDP - Remote Desktop Protocol) для внедрения вредоносного программного обеспечения в сеть организации.

Основные угрозы, создаваемые ВПО

- кража конфиденциальных данных: шпионское ПО и трояны могут осуществлять хищение критически важной информации: интеллектуальной собственности, финансовых данных, персональных записей сотрудников и клиентов.

- нарушение работоспособности инфраструктуры: вирусы и черви способны вызывать отказы в работе серверов, сетевого оборудования и рабочих станций, приводя к длительным простоям и сбоям бизнес-операций.

- обеспечение удаленного контроля: троянские программы часто предоставляют злоумышленникам полный или частичный контроль над зараженной системой, позволяя совершать несанкционированные действия от имени пользователя.

- латеральное распространение по сети: многие виды ВПО (особенно черви) способны быстро и самостоятельно перемещаться между устройствами в пределах локальной сети, увеличивая масштаб заражения.

- финансовые и репутационные потери: инциденты с утечкой данных или длительным простоем IT-систем ведут к прямым финансовым убыткам, штрафам, а также наносят значительный ущерб деловой репутации организации в глазах клиентов и партнеров.

1.2.2 Фишинг и Социальная инженерия

Фишинг - методы психологического воздействия и обмана, используемые злоумышленниками для получения несанкционированного доступа к информации или системам внутри локальной сети. В условиях разработки модели защиты информации от НСД в локальных сетях важно учитывать эти угрозы, поскольку они могут стать первым этапом проникновения злоумышленников внутрь организации.

Виды фишинга и социальной инженерии

- электронный фишинг - рассылка поддельных электронных писем, имитирующих известные компании или внутренние службы организации, с целью заставить пользователя перейти по вредоносной ссылке или ввести конфиденциальные данные.

- веб-фишинг - создание поддельных сайтов, похожих на легитимные ресурсы, для похищения учетных данных при входе пользователя.

- телефонный фишинг - звонки от злоумышленников, выдающих себя за представителей службы поддержки или руководства, с просьбой раскрыть пароли или предоставить доступ к системам.

- социальная инженерия через личные контакты - злоумышленники используют личные связи или доверительные отношения, чтобы получить доступ к конфиденциальной информации или системам.

- инсайдерская атака - злоумышленник, находясь внутри организации, использует доверие для получения доступа к системам и данным.

Угрозы, связанные с фишингом и социальной инженерией

- несанкционированный доступ к сетевым ресурсам - злоумышленник, получив учетные данные через фишинг, может подключиться к внутренней сети и получить доступ к конфиденциальной информации.

- кража конфиденциальных данных - учетные записи, пароли, корпоративные секреты могут быть похищены, что приводит к утечке информации или ее компрометации.

- распространение вредоносного ПО - при переходе по вредоносным ссылкам или открытии зараженных вложений внутри сети злоумышленники могут установить вредоносное ПО (например, трояны или шпионское ПО).

1.2.3 Атаки на пароли

Атаки на пароли являются одним из наиболее распространенных способов несанкционированного доступа к информационным системам и данным внутри локальной сети. В рамках разработки модели защиты информации от несанкционированного доступа важно учитывать методы атак на пароли, их виды и угрозы, которые они создают.

Способы и виды атак на пароли

Подбор паролей (брутфорс) - автоматизированный перебор возможных комбинаций паролей с целью найти правильный. Используются словари, списки часто используемых паролей или полное перебирание всех вариантов.

Атаки методом перебора (грубой силы) - попытки входа с различными комбинациями паролей, зачастую автоматизированные, с помощью специальных программ или скриптов, направленные на подбор правильного пароля.

Атаки через украденные учетные данные - использование украденных или скомпрометированных логинов и паролей, полученных из утечек данных, для автоматического входа в системы.

Перехват данных при передаче - захват паролей, передаваемых по сети, с помощью сниффинга или межсетевых атак, что позволяет злоумышленникам получить доступ к учетным записям.

Угрозы, вызванные атаками на пароли

Несанкционированный доступ к системам и данным - злоумышленник, получив пароль, может управлять внутренними ресурсами, просматривать или изменять конфиденциальную информацию.

Распространение вредоносных программ - используя полученные учетные записи, злоумышленники могут распространять вредоносное ПО внутри сети или на внешних ресурсах.

Кража корпоративных секретов и утечка данных - компрометация учетных данных может привести к утечкам важной информации, что влияет на безопасность и репутацию компании.

Подрыв доверия и финансовые потери - злоупотребление учетными записями может вызвать серьезные убытки, штрафы и снижение доверия клиентов и партнеров.

1.2.4 Атаки на приложения и веб-сервисы

Атаки на приложения и веб-сервисы являются одним из распространенных методов проникновения в локальную сеть, особенно с учетом широкого применения веб-приложений и сервисов внутри корпоративных инфраструктур.

Виды и способы атак на приложения и веб-сервисы

Эксплуатация уязвимостей в программном обеспечении (ПО) - использование известных ошибок или недочетов в коде приложений и сервисов, таких как SQL-инъекции, межсайтовый скриптинг (XSS), выполнение удаленного кода.

SQL-инъекции - внедрение вредоносных SQL-запросов через формы ввода для получения несанкционированного доступа к базе данных и внутренним ресурсам.

Межсайтовый скриптинг (XSS) - внедрение вредоносных скриптов в веб-страницы, что позволяет злоумышленнику выполнить произвольный код у пользователей и похитить их данные или сессии.

Атаки через уязвимости в протоколах и API - использование слабых мест в интерфейсах взаимодействия приложений, таких как REST или SOAP, для получения доступа или выполнения нежелательных действий.

Угрозы, связанные с атаками на приложения и веб-сервисы

Несанкционированный доступ к внутренним системам - злоумышленники могут получить контроль над приложениями, что ведет к утечке данных, повреждению информации или нарушению работы сети.

Кража конфиденциальной информации - получение доступа к базам данных, внутренним ресурсам или пользовательским данным.

Распространение вредоносного ПО - внедрение вредоносных скриптов, вирусов или программ через уязвимости, что может привести к заражению сети.

Финансовые потери - в результате утраты данных и прерывания бизнес-процессов.

1.2.5 DDoS-атаки

DDoS - атаки на отказ в обслуживании, направленные на перегрузку сети или серверов с целью нарушения их доступности для легитимных пользователей. Эти атаки являются серьезной угрозой для инфраструктуры, так как могут привести к остановке работы ключевых сервисов и затруднить контроль доступа.

Основная суть DDoS-атаки заключается в использовании множества зараженных устройств (ботнетов) для одновременного отправления большого объема трафика на целевую систему. В результате этого происходит исчерпание ресурсов сети или вычислительных мощностей серверов, что делает ресурсы недоступными.

Виды DDoS-атак

Объемные атаки - эти атаки характеризуются насыщением канала связи огромным объемом данных, например, UDP-флуд, ICMP-флуд. Их цель - исчерпать пропускную способность сети или сервера, препятствуя обработке легитимных запросов.

Атаки на инфраструктуру - злоумышленники используют уязвимости в сетевых устройствах, таких как маршрутизаторы или межсетевые экраны, чтобы вызвать сбои или отказ в работе всей инфраструктуры.

Атаки на прикладной уровень - эти атаки нацелены на конкретные сервисы или приложения (например, HTTP-атаки), создавая большое число запросов, чтобы исчерпать ресурсы сервера и сделать веб-сервисы не доступными.

Использование ботнетов - зараженные устройства, объединенные в сеть (ботнет), управляются злоумышленником для одновременного запуска масштабных атак, что значительно усложняет защиту.

«Slow»-атаки - например, «Slowloris», при которых злоумышленник медленно устанавливает соединения или посылает минимальный объем данных, истощая ресурсы сервера без большого объема трафика.

Угрозы, связанные с DDoS-атаками

Нарушение доступности сервисов и приложений - в результате атаки внутренние системы, веб-сервисы и информационные ресурсы становятся недоступными для пользователей, что нарушает бизнес-процессы.

Прерывание бизнес-функций - масштабные сбои мешают выполнению критичных задач, вызывая задержки и потери данных.

Финансовые потери - в результате простоев и необходимости реагирования на инциденты организации несут значительные материальные убытки.

Ослабление инфраструктуры безопасности - масштабные атаки могут отвлечь ресурсы специалистов, снизить эффективность систем мониторинга и усложнить обнаружение других угроз.

Усиление риска проникновения (НСД) - DDoS-атакой могут отвлечь внимание службы безопасности, позволяя злоумышленникам провести скрытые атаки или проникновения в сеть.

1.2.6 Неправильная настройка безопасности.

Одной из распространенных причин уязвимости локальных сетей является неправильная конфигурация систем безопасности. Такой сбой зачастую приводит к тому, что потенциальные злоумышленники могут получить несанкционированный доступ к сетевым ресурсам, данным и системам. В рамках разработки модели защиты информации от несанкционированного доступа в локальных сетях важно учитывать аспекты правильной настройки и управления системами безопасности.

Например, неправильная настройка правил межсетевых экранов (firewall) может оставить открытыми критические порты, что позволяет злоумышленникам легко проникнуть в сеть. Также не оптимальные параметры доступа и слабые политики паролей, установленные по умолчанию, значительно увеличивают риск компрометации учетных записей. В случае неправильной настройки систем аутентификации и авторизации злоумышленники могут воспользоваться уязвимостями, чтобы получить доступ к защищенным ресурсам.

Кроме того, неправильная конфигурация систем обновлений и патчей может привести к использованию устаревших версий программного обеспечения, содержащих известные уязвимости. Это создает возможность для атак через эксплойты, которые могут полностью скомпрометировать систему или сеть.

Несвоевременное исправление ошибок конфигурации может позволить злоумышленникам использовать уязвимости для получения несанкционированного доступа, что значительно снижает уровень защиты локальной сети. Поэтому правильная настройка систем безопасности должна стать непрерывным процессом, интегрированным в систему управления информационной безопасностью организации.

1.3. Анализ возможностей подходов для обеспечения безопасности локальной сети

Давайте поговорим про безопасность локальных сетей. Это не просто «поставил фаервол и забыл». Сейчас это целая философия, которая сильно изменилась за последние годы.

Раньше все было проще. Работал так называемый периметровый подход. Представьте свою сеть как крепость: вы строите высокие и толстые стены (это фаерволы), ставите дозорных, которые следят за подозрительной активностью (это системы обнаружения вторжений, IDS/IPS), и делаете один хорошо охраняемый вход (шлюз безопасности). Этот подход отлично работал против атак извне, когда враг был там, за стенами. И он до сих пор важен, никто не отменял необходимость крепких «стен».

Но жизнь стала сложнее. Угрозы эволюционировали. Теперь «враг» может уже быть внутри, притворяясь своим. Например, сотрудник по неосторожности открыл вредоносное письмо, и вирус теперь бродит по внутренней сети. Получается, что одних «стен» уже недостаточно.

Поэтому сейчас все говорят о комплексном, или многоуровневом, подходе. Суть в том, что нельзя полагаться на что-то одно. Нужна целая система мер, которые страхуют и дополняют друг друга. Это как в безопасности аэропорта: есть и рамки на входе, и досмотр ручной, и сканеры багажа, и наблюдение по периметру, и обученный персонал.

Что входит в такой комплекс?

- Защита периметра: те самые «стены» - фаерволы, шлюзы.
- Защита внутри: системы, которые следят, что творится внутри сети, даже если трафик «свой». Они ищут аномалии, подозрительное поведение устройств или пользователей (DLP-системы, чтобы данные не утекли, системы анализа поведения).
- Обнаружение и реакция: не просто засечь угрозу, но и автоматически на нее среагировать - заблокировать подозрительное устройство, изолировать сегмент сети.
- Управление уязвимостями: постоянный аудит и «залатывание дыр» в программном обеспечении.

- А самое главное - люди и процессы: обучение сотрудников, четкие инструкции на случай инцидента.

А что говорит государство?

В России, особенно для госорганов, бюджетных учреждений, компаний из важных отраслей (транспорт, связь, финансы) или работающих с гостайной, подход к безопасности строго регламентирован. Здесь на сцену выходят ФСТЭК России (Федеральная служба по техническому и экспортному контролю) и ФСБ России.

- ФСТЭК - это главный «архитектор» требований по защите информации, которая не является гостайной, но важна для работы государства и общества (персональные данные, коммерческая тайна, служебная информация). Их основные «библии» - это приказы № 17, № 21, № 239 и другие. Эти документы как раз и продвигают тот самый комплексный подход. Они буквально предписывают: чтобы защитить информационную систему, нужно не просто купить какой-то один «волшебный» прибор. Нужно создать систему защиты информации (СЗИ), которая включает и технические средства (отслеживающие, управляющие, защищающие), и организационные меры. По сути, ФСТЭК законодательно закрепляет необходимость перехода от старого периметрового» мышления к современному комплексному.
- ФСБ занимается защитой сведений, составляющих государственную тайну. Их требования (например, приказы, регулирующие использование средств криптографической защиты информации - СКЗИ, вроде знаменитых «КриптоПро») еще строже. Там речь идет об обязательном шифровании, использовании специальной защищенной техники, глубокой проверке персонала. Для таких сетей безопасность - это не рекомендация, а обязательное условие существования.

Современная безопасность сети - это уже не просто «забор». Это комплексная экосистема, которая должна видеть угрозы снаружи и изнутри, быстро на них реагировать и постоянно адаптироваться. А в России для целого ряда организаций этот комплексный подход не просто лучшая практика, а

прямое требование закона, сформулированное ФСТЭК и ФСБ. Поэтому сегодня грамотная защита это всегда синергия технологий, грамотных специалистов и следования актуальным стандартам.

Устранение уязвимостей: проверка и тестирование

Устранение уязвимостей в локальных сетях является критически важным этапом в обеспечении безопасности информационных систем. Это направлено на выявление и исправление остаточных уязвимостей, возникших при проектировании, реализации и эксплуатации сетевой инфраструктуры. Для эффективной защиты информации от несанкционированного доступа (НСД) необходимы различные методы проверки и тестирования:

Статическая проверка:

Анализ проектирования: Проверка на соответствие требованиям безопасности, установленным в Федеральных государственных стандартах (ФГС) и нормативных документах по информационной безопасности, таких как ГОСТ Р 56939-2016 «Защита информации. Общие требования» и нормативных актов ФСТЭК России по обеспечению безопасности критической информационной инфраструктуры.

Проверка сети: Анализ сетевого кода на наличие уязвимостей, таких как SQL-инъекции или XSS, с учетом рекомендаций по безопасной разработке программного обеспечения, изложенных в ГОСТ Р 57580-2017 «Информационная безопасность. Требования к безопасной разработке программных средств».

Проверка конфигурации: Проверка конфигураций коммутаторов, маршрутизаторов и сервисов на соответствие внутренним политикам безопасности, нормативным актам и стандартам ФСТЭК России, а также требованиям по защите информации в государственных информационных системах.

Динамическая проверка

Тестирование на уязвимости: Использование методов тестирования, таких как «белый» хакинг, с соблюдением требований и методик, установленных в нормативных документах ФСТЭК России.

Анализ безопасности: включает анализ логов, трафика и поведения сетевых ресурсов для выявления потенциальных уязвимостей и нарушений безопасности, в соответствии с нормативами по мониторингу и аудиту информационных систем, установленными ФСТЭК России и Роскомнадзором.

Совокупное тестирование

Универсальное тестирование: использование комбинации статических и динамических методов для обеспечения всеобъемлющего уровня защиты, в соответствии с требованиями нормативных актов по управлению информационной безопасностью.

Повторное тестирование: ррегулярные проверки уязвимостей после внесения изменений или обновлений в инфраструктуру, в соответствии с планами и требованиями нормативных актов по обеспечению информационной безопасности, регулирующих деятельность государственных и муниципальных органов.

Защита информации от несанкционированного доступа требует систематического подхода к выявлению, классификации, уведомлению и исправлению уязвимостей, основываясь на российских стандартах и нормативных документах, таких как ГОСТ Р 56939-2016, ГОСТ Р 57580-2017, а также требованиях Федеральной службы по техническому и экспортному контролю (ФСТЭК России) и Федеральной службы по надзору в сфере связи, информационных технологий и массовых коммуникаций (Роскомнадзор).

Предотвращение, обнаружение вторжениям

Современные локальные сети, интегрированные в глобальные сети Интернета, представляют собой сложные системы, потенциально уязвимые для несанкционированного доступа. Для обеспечения безопасности

информационных ресурсов важно внедрить эффективные меры предотвращения, обнаружения и устойчивости к вторжениям:

Предотвращение вторжений:

- современные методы аутентификации и авторизации: Использование многофакторной аутентификации, биометрических данных и других методов для обеспечения подлинности пользователей и ограничения доступа к ресурсам.

- улучшение безопасности внедрения: Проверка и анализ всех внедренных компонентов на наличие уязвимостей, использование либо безопасных протоколов либо защищенных каналов.

Обнаружение вторжений:

- современные системы обнаружения вторжений: использование комплексных систем для выявления аномальных активностей и потенциальных вторжений в сеть.

- мониторинг безопасности: регулярное наблюдение за сетевыми вычислениями и системами для быстрого обнаружения и реагирования на потенциальные угрозы.

Для более надежного обнаружения вторжений используются так называемые «цифровые телохранители» сетей, которые непрерывно мониторят трафик для выявления и предотвращения кибератак, работая на основе анализа сигнатур (известных угроз) и аномалий (отклонений от нормального поведения), применяя машинное обучение и искусственного интеллекта (далее - ИИ) для адаптивного реагирования и блокировки угроз, включая распределенные (DDoS) и продвинутые атаки. Есть два типа систем обнаружения IDS (Intrusion Detection System) и IPS (Intrusion Prevention System).

IDS обнаруживает и оповещает о подозрительной активности, не блокируя ее самостоятельно.

IPS не только обнаруживает, но и активно блокирует угрозы в реальном времени (например, обрывает соединение), действуя как «телохранитель».

Что касается метода обнаружения то они делятся на три типа. Первое это по сигнатурам, система ищет совпадение с шаблонами известных атак. Второе

по аномалиям, система выявляет отклонения от установленной так сказать «нормы» поведения сети или пользователя. Третье гибридное, оно сочетает в себе оба подхода.

Для большей эффективности в систему интегрируют искусственный интеллект. Это позволяет обнаружить неизвестные угрозы при помощи выявления отклонение от нормального поведения, даже если атака не имеет известных сигнатур. Также эта интеграция позволяет уменьшить количество ложный срабатываний и прогнозировать атаки на основе исторических данных.

Устойчивость к вторжениям:

- разработка и реализация планов реагирования: планирование стратегий ответа на ситуацию в случае обнаружения вторжения или несанкционированного доступа.

- обучение и тренировки: регулярное обучение и тренировки сотрудников для эффективного реагирования в случае вторжений.

Устойчивость к вторжениям является критически важным аспектом защиты информации, поскольку она обеспечивает быстрое обнаружение и реагирование на потенциальные угрозы, снижая риск вторжения и сохраняя целостность информации.

Оценка методов защиты информации от НСД

Различные методы защиты, рассмотренные в предыдущих разделах, являются важнейшими компонентами обеспечения безопасности локальных сетей и предотвращения несанкционированного доступа (НСД). Эффективность этих методов напрямую зависит от уровня угроз, существующих в конкретной сети, а также от способности контрмер своевременно обнаруживать и нейтрализовать попытки вторжения. Чем больше остаточных уязвимостей и чем выше частота атак, тем более важна правильная оценка эффективности применяемых мер защиты.

Для оценки эффективности методов защиты информации от НСД используется систематический подход, включающий проведение специальных экспериментов, тестирований и проверок. Основная задача - определить,

насколько успешно реализуемые меры снижают риск несанкционированного доступа и насколько быстро и точно системы обнаруживают и предотвращают попытки вторжений.

1.4. Выбор и обоснование методологии решения задачи

Для разработки эффективной модели защиты информации от несанкционированного доступа в локальной сети выбран системный подход, основанный на концепциях «системной интеграции процессов управления». в рамках этого подхода была сформирована математическая модель, связывающая показатели идентификации угроз и их нейтрализации с уровнем информационной безопасности. методология включает следующие этапы:

- анализ угроз и уязвимостей с использованием методов оценки рисков и моделирования сценариев атак.
- формирование уравнений, отражающих процессы обнаружения и нейтрализации угроз, с учетом вероятностей ошибок и пропусков.
- построение аналитической модели, позволяющей предсказывать эффективность внедряемых средств защиты в конкретных условиях сети.
- внедрение автоматизированных систем мониторинга и управления, использующих математические показатели для оценки текущего уровня безопасности.

обоснование выбора данной методологии заключается в ее универсальности, научной обоснованности и высокой адаптивности к изменениям угроз и технических условий. использование математических моделей и системного анализа позволяет количественно оценивать уровень защиты, выявлять слабые места и оптимизировать меры защиты.

Выводы по разделу:

Для разработки системы защиты информации от несанкционированного доступа в локальной сети выбран методологический подход, основанный на принципах системной интеграции процессов управления. Данный подход, адаптированный для задач информационной безопасности (ИБ), позволяет

перейти от реактивного устранения инцидентов к прогнозируемому и управляемому обеспечению защищенности.

Обоснование выбора методологии

- сложность современных угроз. Традиционные подходы часто оказываются эффективными лишь против изолированных, простых угроз. Для противодействия сложным скоординированным атакам в локальной сети, использующим комбинации уязвимостей, требуется системный взгляд, объединяющий все этапы жизненного цикла угрозы.

- основа методологии составляет формализованная интеграция пяти ключевых процессов: целевой информационный процесс (работа сетевых сервисов и данных). процесс возникновения угрозы. процесс идентификации (обнаружения) угрозы. Процесс нейтрализации угрозы. Количественный показатель уровня информационной безопасности.

Математическая связь этих процессов позволяет свести задачу обеспечения ИБ к нахождению баланса между обобщенной эффективностью обнаружения угроз и обобщенной эффективностью их нейтрализации для достижения заданного целевого уровня защиты.

Применимость к архитектуре локальной сети. Локальные сети часто строятся по клиент-серверной архитектуре, которая, с одной стороны, предлагает преимущества для безопасности (централизованное управление, контроль точек доступа), а с другой - создает критически важные элементы (серверы), атака на которые парализует всю сеть. Предлагаемая методология позволяет количественно оценить риски для таких ключевых активов и оптимизировать распределение средств защиты.

Адаптивность к изменяющимся условиям. Поскольку невозможно заранее знать все векторы атак и уязвимости, система защиты не может быть статичной. Разрабатываемая аналитическая модель на основе системной интеграции позволяет: оценивать эффективность как существующих, так и планируемых к внедрению средств защиты (межсетевые экраны, системы обнаружения вторжений, антивирусные). Выявлять слабые звенья в защите локальной сети.

Прогнозировать уровень безопасности при изменениях в сетевой инфраструктуре или появлении новых типов угроз.

Таким образом, выбранная методология предоставляет научно обоснованный и структурированный инструмент для проектирования не просто набора защитных средств, а целостной, измеримой и адаптивной системы защиты информации от НСД в локальной сети. Она позволяет перейти от качественных описаний к количественным оценкам и управленческим решениям по оптимизации безопасности.

ГЛАВА 2. РАЗРАБОТКА МОДЕЛИ УПРАВЛЕНИЯ ПРОЦЕССАМИ ОБЕСПЕЧЕНИЯ БЕЗОПАСНОСТИ

2.1 Общий подход к разработке метода

В основе обеспечения информационной безопасности локальных сетей лежит принятие решений человеком. Эти решения формируются на основе модели - представления или описания объекта защиты, которое помогает понять его основные свойства и характеристики. В контексте защиты информации от несанкционированного доступа (НСД) модель должна учитывать особенности сетевой инфраструктуры, угрозы, уязвимости и меры защиты.

Решение представляет собой модель процесса защиты, с которой работает специалист по информационной безопасности. Сам процесс - это активная деятельность, выполняемая при определённой цели, направленная на предотвращение или минимизацию рисков НСД. Для достижения целей по обеспечению безопасности важно уметь формировать процессы с заданными свойствами, что требует условий для их существования и моделирования [ИСТОЧНИК].

Без методологических основ разработки и реализации мер защиты информации невозможно гарантировать достижение целей безопасности. Отсутствие таких основ может привести к тому, что принятые меры не дадут ожидаемых результатов, что особенно важно при борьбе с НСД в локальных сетях. Решение задач защиты информации требует системного подхода, основанного на моделировании и управлении процессами, обеспечивающими целостность, конфиденциальность и доступность данных [ИСТОЧНИК].

Основной проблемой является несоответствие результатов реализуемых мер безопасности ожиданиям руководителей. Они действуют, исходя из трёх ключевых категорий: системы, модели и предназначения. В разработке систем защиты информации используют два подхода: анализ и синтез. В рамках синтеза для обеспечения целостности и эффективности систем защиты применяется закон сохранения целостности объекта (ЗСЦО), который обеспечивает достижение целей защиты. ЗСЦО - это устойчивое, объективное и

повторяющееся соотношение свойств объекта и его действий при определённом предназначении [ИСТОЧНИК].

Для создания эффективных систем защиты информации на базе модели необходимо уметь синтезировать адекватные модели, отражающие состояние и угрозы локальной сети. Эффективная работа систем защиты возможна только при построении правильной системы (ППС) и использовании соответствующих моделей. В данной работе ЗСЦО рассматривается как условие существования и функционирования систем защиты информации в локальных сетях.

Модель функционирования системы защиты основана на системной интеграции четырёх процессов:

- целевой процесс (функционирование локальной вычислительной сети),
- формирование проблемы (возникновение угрозы НСД),
- распознавание проблемы (идентификация угрозы),
- устранение проблемы (нейтрализация угрозы).

В данной выпускной квалификационной работе управленческое решение представлено в виде математической модели:

$P = F (ТЭ, \Delta t_{пп}, \Delta t_{ип}, \Delta t_{нп})$,

где:

$\Delta t_{пп}$ - среднее время обнаружения угрозы НСД.

$\Delta t_{ип}$ - среднее время идентификации угрозы.

$\Delta t_{нп}$ - среднее время нейтрализации угрозы.

ТЭ - временная характеристика, требуемая для прогнозирования состояния защищенности ЛВС.

Данное уравнение является условием существования процесса управления безопасностью ЛВС от НСД. Графическая иллюстрация процесса формирования модели решения представлена на Рисунке 1.

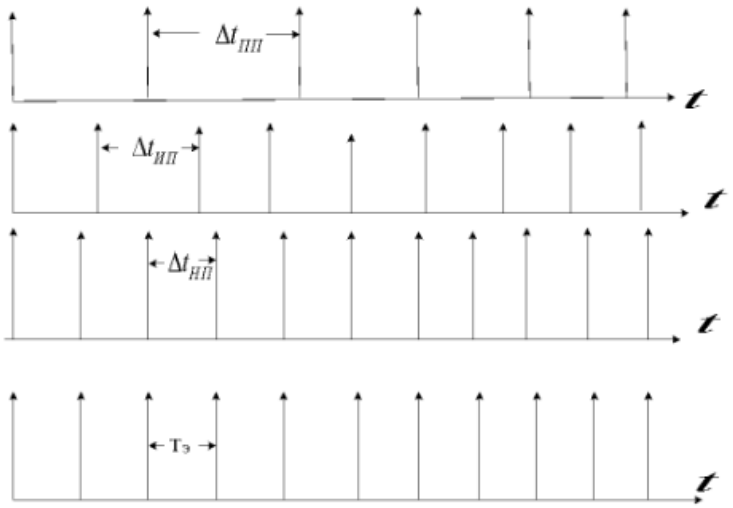

Рисунок 1 - Диаграмма проявления базовых элементов формирования модели решения

Рассмотренные четыре процесса, изображённые на диаграммах, отражают жизненный цикл системы защиты локальной вычислительной сети (ЛВС). Эти процессы показывают последовательность этапов, необходимых для поддержания и обеспечения безопасности сети на протяжении всего её функционирования.

Модель функционирования ЛВС можно представить в виде графа (см. Рисунок 2), который включает два основных состояния:

- «1» - начальное состояние, характеризующееся нормальной работой системы и её готовностью к выполнению задач защиты.
- «2» - конечное состояние, связанное с выполнением поставленной задачи по обеспечению безопасности или завершением определённого этапа деятельности.

Этот граф отображает переходы между состояниями и последовательность процессов, формирующих жизненный цикл системы защиты, что позволяет лучше понять динамику и управление безопасностью в локальной сети.

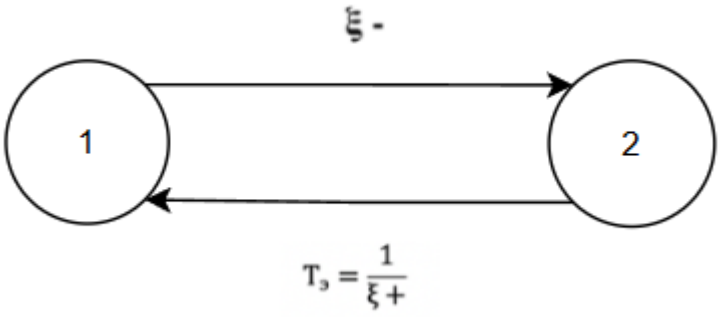

Рисунок 2 - Граф состояний

Среднее время обработки запроса в сети обозначается как «Тэ=1/ξ+». В процессе работы возможны сбои и атаки, которые приводят к несанкционированному доступу (НСД). Этот аспект описывается через частоту возникновения срыва выполнения запроса «ξ-». Надёжность локальной вычислительной сети (ЛВС) характеризуется частотой успешных запросов «Тэ=1/ξ+». Деятельность злоумышленников создает угрозу нарушению функционирования ЛВС. В связи с этим возникает важная задача: как связать процесс работы ЛВС с системой защиты от НСД.

Для решения этой задачи применяется естественно-научный подход (ЕНП), который основывается на интеграции свойств мышления человека, окружающего мира и процесса познания.

Процесс принятия решений включает следующие этапы: постановка цели, сбор и обработка информации, подготовка вариантов решений, координация действий, принятие окончательного решения и контроль его выполнения.

Также необходимо учитывать логику работы системы управления безопасностью ЛВС и сократить время, затрачиваемое специалистом на принятие решений при обнаружении уязвимостей или атак, с помощью средств автоматического управления. Для внедрения автоматического управления важно определить условия существования процесса и обеспечить механизм обратной связи. Структурная схема данного процесса изображена на рисунке 3.

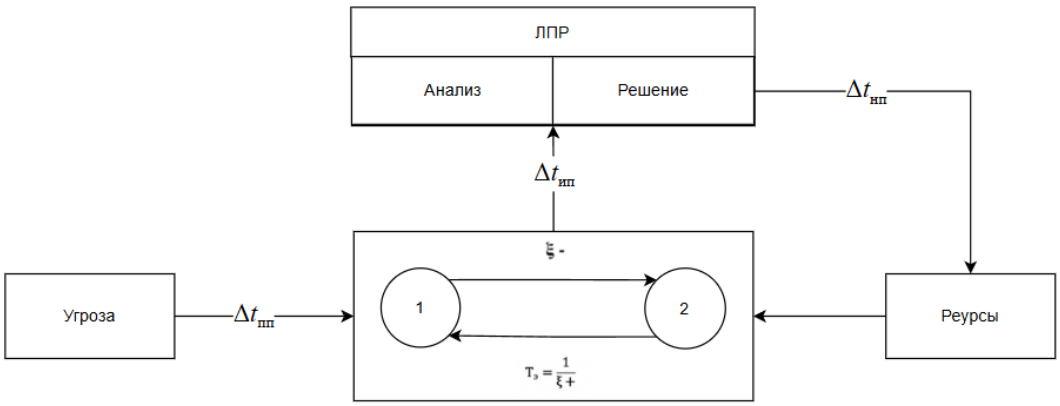

Рисунок 3 - Структурная схема функционирования ЛВС, на основе обратной связи

В результате синтеза модель управления безопасностью ЛВС была преобразована в математическую модель управленческого решения. Рассмотрим логику этого процесса: когда ЛВС находится в состоянии 3 под воздействием интенсивности угроз λ, система должна обнаружить и идентифицировать угрозу. На этапе первичного выявления и осознания проблемы специалист затрачивает время $\Delta t_{\text{пи}}$. Этот этап включает подготовку к привлечению ресурсов для устранения угрозы. После этого система переходит в состояние 4, где специалист распознаёт конкретную угрозу и определяет механизмы её нейтрализации. Затем происходит устранение угрозы, и система возвращается в состояние 2 - «проблема решена». После этого на вход может поступить новая угроза, и процесс повторяется.

Возврат системы в исходное работоспособное состояние характеризует её устойчивость к множеству угроз. Частота перехода системы из состояния 1 в состояние 2, обозначенная как ξ^{+}, равна величине, обратной среднему времени выполнения целевой задачи. Этот показатель отражает эффективность восстановления системы после устранения угрозы. В свою очередь, ξ^{-}- частота перехода из состояния 2 в состояние 1, которая характеризует среднюю частоту невыполнения задач из-за атак; на практике этот показатель должен быть очень низким, например, не превышать 0,1%.

Частота перехода из состояния 4 в состояние 2, равная $v_2 = 1/\Delta t_{\text{пн}}$, зависит от среднего времени нейтрализации угрозы $\Delta t_{\text{пн}}$. Уровень компетентности системы в решении неизвестных задач определяется соотношением этих частот v_2.

Данная логика позволяет построить граф состояний процесса формирования управленческого решения, изображённый на Рисунке 4, где отражены все переходы между состояниями.

.

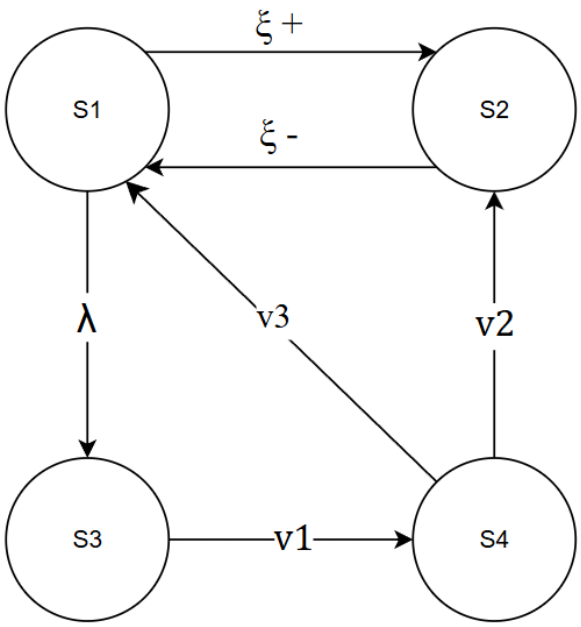

Рисунок 4 - Граф состояний, процесса формирования управленческого решения

На основе этого может быть применена система дифференциальных уравнений Колмогорова:

$$\frac{dP_i(t)}{dt} = \sum_{j=1}^{n} \lambda_{ji}(t) * P_j(t) - P_i(t) * \sum_{j=1}^{n} \lambda_{ji}(t)$$

где i=0,1, 2,…,n.

Конечные вероятности состояний можно определить, решая систему линейных алгебраических уравнений, полученных из дифференциальных уравнений Колмогорова при условии, что их производные равны нулю. В этом случае вероятностные функции состояний $P_1(t), \dots, P_n(t)$в правых частях

уравнений заменяются на искомые конечные вероятности $\mathrm{P}_1, \ldots, \mathrm{P}_n$. Для получения точных значений вероятностей необходимо дополнить систему уравнений нормализующим условием:

$$\mathrm{P}_0 + \mathrm{P}_1 + \cdots + \mathrm{P}_n = 1,$$

где P_0- вероятность начального состояния.

Для графа состояний, изображенного на рисунке 5, система уравнений Колмогорова записывается в виде:

$$\begin{cases} \frac{dP_1(t)}{dt} = -(\xi^+ + \lambda) * P_1(t) + \xi^- * P_2(t) \\ \frac{dP_2(t)}{dt} = \xi^+ * P_1(t) - \xi^- * P_2(t) + v_2 * P_4(t) \\ \frac{dP_3(t)}{dt} = \lambda * P_1(t) - v_1 * P_3(t) \\ \frac{dP_4(t)}{dt} = v_1 * P_3(t) - v_2 * P_4(t) \end{cases}$$

Рисунок 5 - Система уравнений Колмогорова

Тогда конечные вероятности могут быть вычислены путем решения системы линейных алгебраических уравнений.

$$\begin{cases} 0 = -(\xi^+ + \lambda) * P_1 + \xi^- * P_2 \\ 0 = \xi^+ * P_1 - \xi^- * P_2 + v_2 * P_4 \\ 0 = \lambda * P_1 - v_1 * P_3 \\ 1 = P_1 + P_2 + P_3 + P_4 \end{cases}$$

Системное решение выглядит следующим образом:

$$P_1 = \frac{v_1 * v_2 * \xi^-}{\lambda * v_1 * v_2 + \lambda * v_1 * \xi^- + \lambda * v_2 * \xi^- + v_1 * v_2 * \xi^+ + v_1 * v_2 * \xi^-}$$

$$P_2 = \frac{\lambda * v_1 * v_2 + v_1 * v_2 * \xi^+}{\lambda * v_1 * v_2 + \lambda * v_1 * \xi^- + \lambda * v_2 * \xi^- + v_1 * v_2 * \xi^+ + v_1 * v_2 * \xi^-}$$

$$P_3 = \frac{\lambda * v_2 * \xi^-}{\lambda * v_1 * v_2 + \lambda * v_1 * \xi^- + \lambda * v_2 * \xi^- + v_1 * v_2 * \xi^+ + v_1 * v_2 * \xi^-}$$

$$P_4 = \frac{\lambda * v_1 * \xi^-}{\lambda * v_1 * v_2 + \lambda * v_1 * \xi^- + \lambda * v_2 * \xi^- + v_1 * v_2 * \xi^+ + v_1 * v_2 * \xi^-}$$

Вероятность обнаружения и нейтрализации угрозы определяется следующей корреляцией:

$$P_2 = \frac{\lambda * v_1 * v_2 + v_1 * v_2 * \xi^+}{\lambda * v_1 * v_2 + \lambda * v_1 * \xi^- + \lambda * v_2 * \xi^- + v_1 * v_2 * \xi^+ + v_1 * v_2 * \xi^-}$$

Это соотношение связывает три ключевых параметра. Таким образом, получена аналитическая зависимость между обобщёнными характеристиками: временем появления проблемы (Δtпп), её идентификации (Δtип) и нейтрализации (Δtнп) в контексте обеспечения безопасности ЛВС.

Для анализа модели применяются сетевые модели, являющиеся разновидностью ориентированных графов. Вершины графа соответствуют событиям обнаружения проблем, обозначающим начало и окончание отдельных работ, а дуги - самим работам. В данном исследовании используется сетевая модель, поскольку она наглядно демонстрирует взаимодействие между узлами системы.

2.2 Проектирование сетевой модели возникновения угроз НСД в локальной сети

Основная задача проектирования заключается в установлении взаимосвязи между программно-аппаратными компонентами локальной сети и параметрами её безопасности. Возникновение угрозы несанкционированного доступа (НСД), как правило, сопровождается отклонением штатных параметров функционирования информационной системы.

2.2.1 Сетевая модель угроз

Для анализа уязвимостей строится сетевая модель, представляющая локальную сеть как комплекс взаимодействующих компонентов. Модель, представленная на рисунке 6, разделяет среду возникновения угроз НСД на три ключевые подсистемы:

1. Программное обеспечение.
2. Аппаратное обеспечение.
3. Информационные ресурсы.

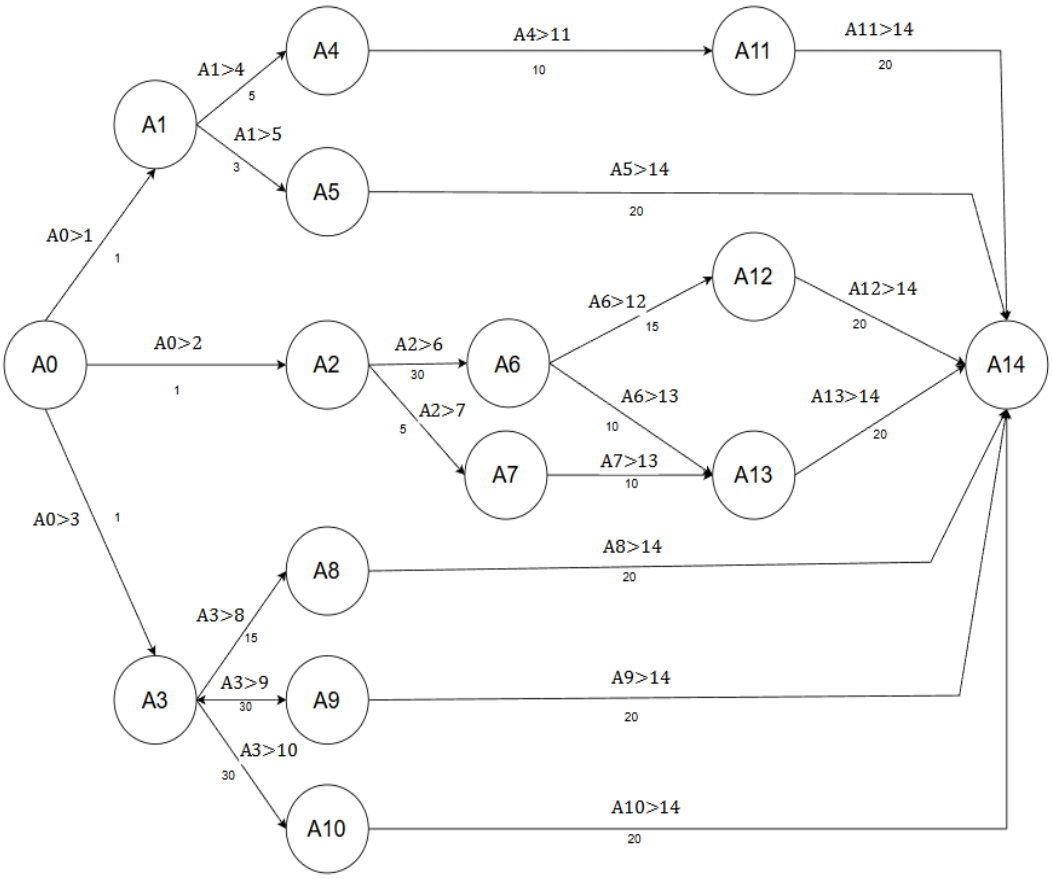

Рисунок 6 - Сетевой граф угроз

В таблице 1 представлен перечень инцидентов безопасности, которые могут привести к реализации несанкционированного доступа в рамках локальной вычислительной сети (ЛВС).

Таблица 1 - Перечень событий

Обозначение (ai)	**Наименование *i*-го события процесса**
1	2
a_0	Угроза
a_1	Угроза аппаратному обеспечению
a_2	Угроза программному обеспечению
a_3	Угроза информационным ресурсам
a_4	Подключение неавторизованных сетевых устройств к коммутаторам
a_5	Подключение к кабельной инфраструктуре

Обозначение (ai)	**Наименование *i*-го события процесса**
1	2
a_6	Внедрение и активация вредоносной программы
a_7	Внедрение эксплойта для взлома системы аутентификации
a_8	Несанкционированное изменение данных
a_9	Перехват пакетов
a_{10}	Несанкционированный доступ к конфиденциальным данным
a_{11}	Дестабилизация работы сети
a_{12}	Нарушение целостности информационных ресурсов
a_{13}	Прекращение работы ПО
a_{14}	Нарушение работы ЛВС

Таблица 2 - Перечень угроз

Обозначение работ	**Наименование работ**	**Предшествующие работы**	**Последующие работы**
1	2	4	5
A_{0-1}	Атака на аппаратную часть	-	A_{1-4}, A_{1-5}
A_{1-4}	Выход из строя оборудования	A_{0-1}	A_{4-11}
A_{1-5}	Подключение к кабельной инфраструктуре	A_{0-1}	A_{5-14}
A_{4-11}	Перегрузка трафика сети	A_{1-4}	A_{11-14}
A_{5-14}	Анализ информации и нарушение работы системы	A_{1-5}	-
A_{11-14}	Анализ информации и нарушение работы системы	A_{4-11}	-
A_{0-2}	Атака на программную часть	-	A_{2-6}, A_{2-7}
A_{2-6}	Внедрение и активация ВПО	A_{0-2}	A_{6-12}, A_{6-13}
A_{2-7}	Внедрение эксплойта для взлома системы аутентификации	A_{0-2}	A_{7-13}
A_{6-12}	Несанкционированный доступ к	A_{2-6}	A_{12-14}

Обозначение работ	Наименование работ	Предшествующие работы	Последующие работы
1	2	4	5
	данным		
A_{6-13}	Нарушение сетевого соединения	A_{2-6}	A_{13-14}
A_{7-13}	Нарушение сетевого соединения	A_{2-7}	A_{13-14}
A_{12-14}	Анализ информации и нарушение работы системы	A_{6-12}	-
A_{13-14}	Анализ информации и нарушение работы системы	A_{6-13} A_{7-13}	-
A_{0-3}	Нарушение работы информационных ресурсов	-	A_{3-8}, A_{3-9}, A_{3-10}
A_{3-8}	Несанкционированный доступ к конфиденциальным данным	A_{0-3}	A_{8-14}
A_{3-9}	Получение контроля над работой системы	A_{0-3}	A_{9-14}
A_{3-10}	Внедрение ВПО для дальнейших атак (плацдарм)	A_{0-3}	A_{10-14}
A_{8-14}	Анализ информации и нарушение работы системы	A_{3-8}	-
A_{9-14}	Анализ информации и нарушение работы системы	A_{3-9}	-
A_{10-14}	Анализ информации и нарушение работы системы	A_{3-10}	-

2.2.2. Сетевая модель мониторинга

Стадия анализа, входящие в который процедуры обрабатывают полученную на стадии мониторинга информацию, сравнения с ранее полученными результатами и на основании результатов, предоставление информации о вероятных причинах сбоев или ненадежной работе ЛВС. Для составления сетевой модели мониторинга системы нужно определить список работ, которые следует сделать для полного мониторинга и время, потраченное на эти работы (Таблица 3).

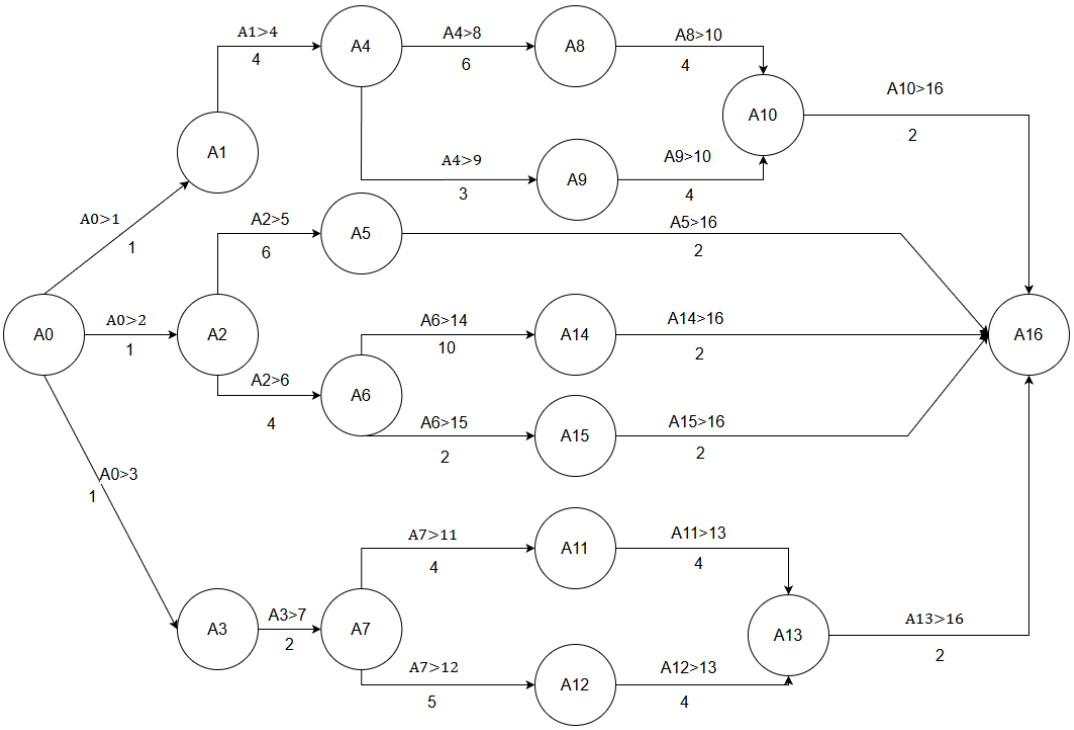

Рисунок 7 - Сетевой график мониторинга ЛВС

В таблице 3 представлен перечень событий, мониторинга ЛВС .

Таблица 3 - Перечень событий

Обозначение	**Наименование *i*-го события процесса**	**Тр, с**	**Тп, с**	**Ri, с**
a_0	Мониторинг ЛВС	0	0	0
a_1	Мониторинг аппаратной части	1	1	0
a_2	Мониторинг программной части	1	1	0
a_3	Мониторинг информационных ресурсов	1	4	3
a_4	Проверка состояния работоспособности оборудования	5	5	0
a_5	Успешное выполнение проверки ПО и формирование отчета	7	15	8
a_6	Обнаружение ВПО	5	5	0
a_7	Проверка стабильности соединения	3	6	3
a_8	Выполнение проверки оборудования на наличие проблем	11	11	0

a_9	Обнаружено несанкционированное подключение к оборудованию	8	11	3
a_{10}	Формирование результата проверок	15	15	0
a_{11}	Анализ события	7	11	4
a_{12}	Система в стабильном состоянии	8	11	3
a_{13}	Формирование отчета	12	15	3
a_{14}	Попытка автоматического реагирование и формирование отчета	15	15	0
a_{15}	Формирование отчета по событию	7	15	8
a_{16}	Запись в журнал событий	17	17	0

Таблица 4 - Перечень работ

Обозначение работ	**Наименование работ**	t_{ij}, с	**Предшествующие работы**	**Последующие работы**	$r_n(i,j)$, с
A(0,1)	Запуск мониторинга аппаратной части	1	-	A(1,4)	0
A(0,2)	Запуск мониторинга программной части	1	-	A(2,5), A(2,6)	0
A(0,3)	Запуск мониторинга информационных ресурсов	1	-	A(3,7)	3
A(1,4)	Начало проверки оборудования	4	A(0,1)	A(4,8), A(4,9)	0
A(2,5)	Проверка состояния ПО	6	A(0,2)	A(5,16)	8

Обозначение работ	**Наименование работ**	t_{ij}, с	**Предшествующие работы**	**Последующие работы**	$r_n(i,j)$, с
A(2,6)	Сбой при проверка состояния ПО	4	A(0,2)	A(6,14), A(6,15)	0
A(3,7)	Подготовка к проверка системы	2	A(0,3)	A(7,11), A(7,12)	3
A(4,8)	Работоспособность не нарушена, запуск следующей проверки	6	A(1,4)	A(8,10)	0
A(4,9)	Зафиксировано аномальное поведение оборудования	3	A(1,4)	A(9,10)	3
A(5,16)	Отправка отчета по событию	2	A(2,5)	-	8
A(6,14)	Сбор данных об угрозе	10	A(2,6)	A(14,16)	0
A(6,15)	Отправка сигнала об обнаружении угрозы	2	A(2,6)	A(15,16)	8
A(7,11)	Зафиксирован аномальный трафик	4	A(3,7)	A(11,13)	4
A(7,12)	Сканирование сети на наличие угроз	5	A(3,7)	A(12,13)	3
A(8,10)	Анализ полученных данных	4	A(4,8)	A(10,16)	0
A(9,10)	Анализ полученных данных	4	A(4,9)	A(10,16)	3
A(10,16)	Отправка отчета по событию	2	A(8,10), A(9,10)	-	0

Обозначение работ	**Наименование работ**	t_{ij}, с	**Предшествующие работы**	**Последующие работы**	$r_n(i,j)$, с
A(11,13)	Анализ полученных данных	4	A(7,11)	A(13,16)	4
A(12,13)	Анализ полученных данных	4	A(7,12)	A(13,16)	3
A(13,16)	Отправка отчета по событию	2	A(11,13)	-	3
A(14,16)	Отправка отчета по событию	2	A(6,14)	-	0
A(15,16)	Отправка отчета по событию	2	A(6,15)	-	8

Анализ сетевого графика мониторинга

Основными параметрами сетевого графика являются:

1. Наиболее раннее возможное время наступления j-го события T_p (j), вычисляемое по формуле:

$$T_P(j) = \frac{\max}{i \subset j}(T_P(i) - t_{ij}),$$

где - i и j обозначаются номера предшествующего и последующего событий соответственно;

- t_{ij} - продолжительность (i, j) -й работы.

Из обозначения $i \subset j$ следует, что событие i предшествует событию j.

Расчеты:

Tp(0) = 0
Tp(1) = 1
Tp(2) = 1
Tp(3) = 1
Tp(4) = 5
Tp(5) = 7
Tp(10) = 15
Tp(11) = 7
Tp(12) = 8
Tp(13) = 12
Tp(14) = 15
Tp(15) = 7

Tр(6) = 5
Tр(7) = 3
Tр(8) = 11
Tр(9) = 8
Tр(16) = 17

2. Самое позднее допустимое время наступления i-го события Tп (i), вычисляемое по формуле

$$T_{П}(j) = \frac{\min}{i \supset j}(T_{П}(i) - t_{ij}),$$

где из обозначения $i \supset j$ следует, что событие j предшествует событию i

Расчеты:

Tп(0) = 0
Tп(1) = 1
Tп(2) = 1
Tп(3) = 4
Tп(4) = 5
Tп(5) = 15
Tп(6) = 5
Tп(7) = 6
Tп(8) = 11
Tп(9) = 11
Tп(10) = 15
Tп(11) = 11
Tп(12) = 11
Tп(13) = 15
Tп(14) = 15
Tп(15) = 15
Tп(16) = 17

3. Резерв времени данного события Ri вычисляемый по формуле

$$R_i = (T_{П}(i) - T_{P}(i))$$

Расчеты:

R(0) = 0
R(1) = 1-1 = 0
R(2) = 1-1 = 0
R(3) = 4-1 = 3
R(4) = 5-5 = 0
R(5) = 15-7 = 8
R(6) = 5-5 = 0
R(7) = 6-3 = 3
R(8) = 11-11 = 0
R(9) = 11-8 = 3
R(10) = 15-15 = 0
R(11) = 11-7 = 4
R(12) = 11-8 = 3
R(13) = 15-12 = 3
R(14) = 15-15 = 0
R(15) = 15-7 = 8
R(16) = 17-17 = 0

4. Полный резерв времени работы rп (i,j), вычисляемый по формуле

$$r_{П}(\mathrm{i},\mathrm{j}) = (T_{П}(j) - T_{P}(i) - t_{ij})$$

Расчеты:

rп(1,2) = 0
rп(1,3) = 0
rп(1,4) = 3
rп(2,5) = 0
rп(3,6) = 8
rп(3,7) = 0
rп(4,8) = 3
rп(5,9) = 0
rп(5,10) = 3
rп(6,17) = 8
rп(7,15) = 0
rп(7,16) = 8
rп(8,12) = 4
rп(8,13) = 3
rп(9,11) = 0
rп(10,11) = 3
rп(11,17) = 0
rп(12,14) = 4
rп(13,14) = 3
rп(14,17) = 3
rп(15,17) = 0
rп(16,17) = 8

2.2.3 Сетевая модель устранения проблемы в ЛВС

Определяем работы, которые следует сделать для полного устранения выявленных проблем:

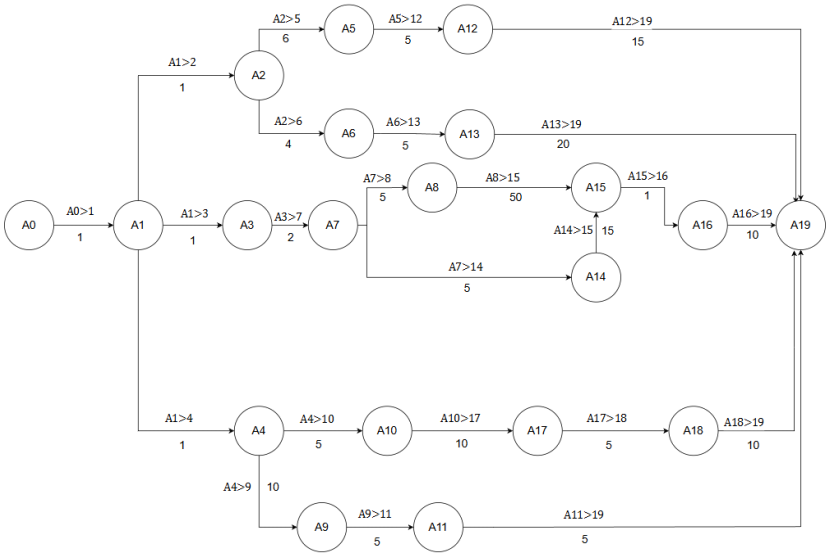

Рисунок 8 - Сетевой график нейтрализации угрозы в ЛВС

В таблице 5 представлен перечень событий, нейтрализации угрозы ЛВС .

Таблица 5 - Перечень событий

Обозначение	**Наименование *i*-го события процесса**	**Тр, с**	**Тп, с**	**Ri, с**
a_0	Получение отчетов об угрозе	0	0	0
a_1	Анализ и распределение отчетов	1	1	0
a_2	Изолирование подключенного устройства	1	20	19
a_3	Анализ инцидента ВПО	1	1	0
a_4	Принятие необходимых мер	1	20	19
a_5	Повторное сканирование на наличие проблемы	4	25	21
a_6	Временная блокировка доступа к ресурсам	3	22	19
a_7	Изолирование зараженного АРМ	2	2	0
a_8	Полная переустановка системы	5	5	0
a_9	Переход на резервные каналы	6	30	24
a_{10}	Фильтрация и блокировка трафика	4	22	19
a_{11}	Продолжение работы системы	8	32	24
a_{12}	Устранение проблемы и обновление прошивки	6	27	21
a_{13}	Физическое отключение оборудования и проверка системы	0	25	19
a_{14}	Переустановка зараженного ПО	5	22	17
a_{15}	Повторное сканирование	28	28	0
a_{16}	Проблем не обнаружено	30	30	0

	восстановление работы системы			
a_{17}	Повторное сканирование трафика	8	27	19
a_{18}	Проблем не обнаружено восстановление работы системы	11	29	19
a_{19}	Проблема устранена	40	40	0

Таблица 6 - Перечень работ

Обозначение работ	**Наименование работ**	t_{ij}, с	**Предшествующие работы**	**Последующие работы**	$r_n(i,j)$, с
A(0,1)	Получение отчетов	1	-	A(1,2), A(1,3), A(1,4)	0
A(1,2)	Анализ аппаратных проблем	1	A(0,1)	A(2,5), A(2,6)	19
A(1,3)	Анализ программных проблем	1	A(0,1)	A(3,7)	0
A(1,4)	Анализ угрозы информационным ресурсам	1	A(0,1)	A(4,9), A(4,10)	19
A(2,5)	Успешное изолирование и блокировка устройства	3	A(1,2)	A(5,12)	21
A(2,6)	Сбой при изолировании	2	A(1,2)	A(6,13)	19
A(3,7)	Анализ ситуации	1	A(1,3)	A(7,8), A(7,14)	0
A(4,9)	Изолирование критически важной инфраструктуры	5	A(1,4)	A(9,11)	24
A(4,10)	Активизация протоколов фильтрации и блокировки	2	A(1,4)	A(10,17)	18
A(5,12)	Сканирование оборудования	2	A(2,5)	A(12,19)	21
A(6,13)	Блокировка доступа	2	A(2,6)	A(13,19)	19

Обозначение работ	**Наименование работ**	t_{ij}, с	**Предшествующие работы**	**Последующие работы**	$r_n(i,j)$, с
A(7,8)	Подготовка к переустановки системы	2	A(3,7)	A(8,15)	0
A(7,14)	Подготовка к переустановка зараженного ПО	2	A(3,7)	(14,15)	17
A(8,15)	Переустановка системы	25	A(7,8)	A(15,16)	0
A(9,11)	Переход на резервные каналы для непрерывной работы бизнес-процессов	2	A(4,9)	A(11,19)	24
A(10,17)	Фильтрация и блокировка трафика	5	A(4,10)	A(17,18)	19
A(11,19)	Продолжение работы системы в штатном режиме	2	A(9,11)	-	19
A(12,19)	Устранение проблем	8	A(5,12)	-	21
A(13,19)	Устранение проблем	10	A(6,13)	-	19
A(14,15)	Переустановка зараженного ПО	8	A(7,14)	A(15,16)	17
A(15,16)	Повторное сканирование	1	A(14,15)	A(16,19)	0
A(16,19)	Восстановление работы системы	5	A(15,16)	-	0
A(17,18)	Запуск повторного сканирования	2	A(10,17)	A(18,19)	17
A(18,19)	Восстановление работы системы	5	A(17,18)	-	17

Анализ сетевого графика нейтрализации угрозы

Основными параметрами сетевого графика являются:

1. Наиболее раннее возможное время наступления j-го события T_p (j), вычисляемое по формуле:

$$T_P(j) = \frac{\max}{i \subset j}(T_P(i) - t_{ij})$$,

где - i и j обозначаются номера предшествующего и последующего событий соответственно;

- t_{ij} - продолжительность (i, j) -й работы.

Из обозначения $i \subset j$ следует, что событие i предшествует событию j.

Расчеты:

Тр(0) = 0
Тр(1) = 1
Тр(2) = 1
Тр(3) = 1
Тр(4) = 1
Тр(5) = 4
Тр(6) = 3
Тр(7) = 2
Тр(8) = 5
Тр(9) = 6
Тр(10) = 4
Тр(11) = 8
Тр(12) = 6
Тр(13) = 5
Тр(14) = 4
Тр(15) = 29
Тр(16) = 30
Тр(17) = 8
Тр(18) = 11
Тр(19) = 40

2. Самое позднее допустимое время наступления i-го события Тп (i), вычисляемое по формуле

$$T_П(j) = \frac{\min}{i \supset j}(T_П(i) - t_{ij})$$,

где из обозначения $i \supset j$ следует, что событие j предшествует событию i

Расчеты:

Тп(0) = 0
Тп(1) = 1
Тп(2) = 20
Тп(3) = 1
Тп(4) = 20
Тп(5) = 25
Тп(10) = 22
Тп(11) = 32
Тп(12) = 27
Тп(13) = 25
Тп(14) = 22
Тп(15) = 29

Тп(6) = 22
Тп(7) = 2
Тп(8) = 4
Тп(9) = 30
Тп(16) = 30
Тп(17) = 27
Тп(18) = 30
Тп(19) = 40

3. Резерв времени данного события Ri вычисляемый по формуле

$$R_i = (T_П(i) - T_P(i))$$

Расчеты:

R(0) = 0
R(1) = 0
R(2) = 19
R(3) = 0
R(4) = 19
R(5) = 21
R(6) = 19
R(7) = 0
R(8) = 0
R(9) = 24
R(10) = 19
R(11) = 24
R(12) = 21
R(13) = 19
R(14) = 17
R(15) = 0
R(16) = 0
R(17) = 17
R(18) = 17
R(19) = 0

4. Полный резерв времени работы rп (i,j), вычисляемый по формуле

$$r_П(\mathrm{i,j}) = (T_П(j) - T_P(i) - t_{ij})$$

Расчеты:

rп(0,1) = 0
rп(1,2) = 19
rп(1,3) = 0
rп(1,4) = 19
rп(2,5) = 21
rп(2,6) = 19
rп(3,7) = 0
rп(4,9) = 24
rп(4,10) = 19
rп(5,12) = 21
rп(6,13) = 19
rп(7,14) = 17
rп(8,15) = 0
rп(9,11) = 24
rп(10,17) = 19
rп(11,19) = 24
rп(12,19) = 21
rп(13,19) = 19
rп(14,15) = 17
rп(15,16) = 0
rп(16,19) = 0
rп(17,18) = 19

rп(7,8) = 0 rп(18,19) = 19

2.2.4 Сетевая модель функционирования ЛВС

На рисунке 9 представлен граф функционирования ЛВС

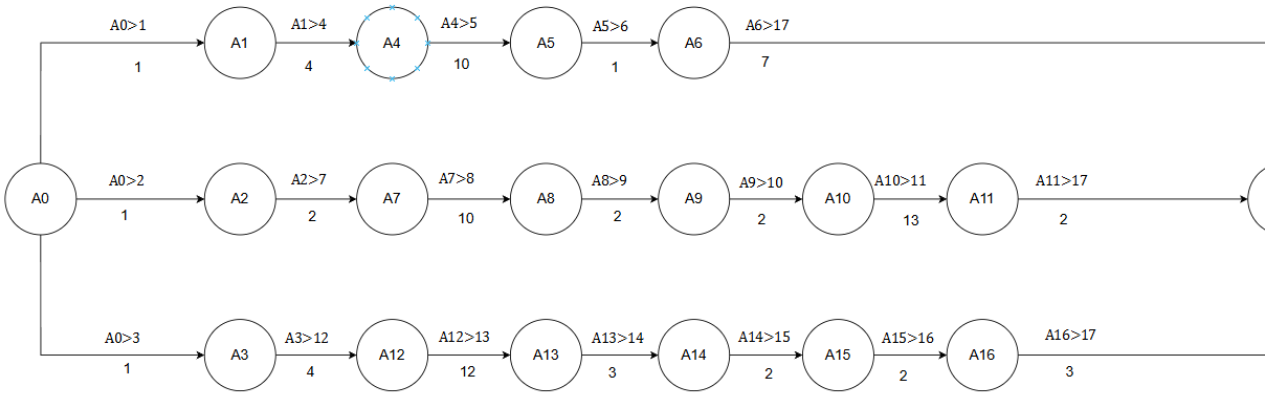

Рисунок 9 - Сетевой график функционирования ЛВС

Таблица 7 - Перечень событий

Обозначение	**Наименование *i*-го события процесса**	**Tp, м**	**Tп, м**	**Ri, м**
a_0	Начало функционирования ЛВС	0	0	0
a_1	Запуск аппаратной части	1	10	9
a_2	Запуск программной части	1	1	0
a_3	Проверка доступности информационных ресурсов	1	6	5
a_4	Проверка состояния оборудования	5	14	9
a_5	Тестирование соединения	15	24	9
a_6	Создание и отправка отчета для мониторинга	16	25	9
a_7	Ввод данных для входа в учетную запись пользователя	3	3	0
a_8	Проверка авторизации пользователя	13	13	0
a_9	Открытие доступа к ПО	15	15	0
a_{10}	Сканирование АРМ на наличие нарушений	17	17	0
a_{11}	Создание и отправка отчета для мониторинга	30	30	0
a_{12}	Обработка трафика сервером	5	10	5
a_{13}	Сформирован запрос к базе данных	17	22	5

Обозначение	Наименование *i*-го события процесса	Тр, м	Тп, м	Ri, м
a_{14}	Проверка прав пользователя	20	25	5
a_{15}	Проверка целостности данных	22	27	5
a_{16}	Проверка прошла успешно, разрешено использование ресурсов	24	29	5
a_{17}	Успешное функционирование ЛВС	32	32	0

Таблица 8 - Перечень работ

Обозначение работ	Наименование работ	t_{ij}, м	Предшествующие работы	Последующие работы	$r_n(i,j)$, м
A(0,1)	Инициирование запуска аппаратной части ЛВС	1	-	A(1,4)	9
A(0,2)	Инициирование запуска программной части ЛВС	1	-	A(2,7)	0
A(0,3)	Инициирование подключения к информационным ресурсов	1	-	A(3,12)	5
A(1,4)	Начало работы аппаратной части	4	A(0,1)	A(4,5)	9
A(2,7)	Запуск программной части	2	A(0,2)	A(7,8)	0
A(3,12)	Тестирование соединения с серверами	4	A(0,3)	A(12,13)	5
A(4,5)	Проверка настройки работы оборудования	10	A(1,4)	A(5,6)	9
A(5,6)	Проверка соединения между устройствами	1	A(4,5)	A(6,17)	9
A(6,17)	Отправка отчета для журналирования	7	A(5,6)	-	9

Обозначение работ	**Наименование работ**	t_{ij}, м	**Предшествующие работы**	**Последующие работы**	$r_n(i,j)$, м
	события				
A(7,8)	Ввод логина и пароля пользователя	10	A(2,7)	A(8,9)	0
A(8,9)	Отправка запроса к серверу безопасности	2	A(7,8)	A(9,10)	0
A(9,10)	Разрешен доступ к ПО на АРМ	2	A(8,9)	A(10,11)	0
A(10,11)	Начато первичное сканирование системы	13	A(9,10)	A(11,17)	0
A(11,17)	Отправка отчета	2	A(10,11)	-	0
A(12,13)	Анализ входящего/исходящего трафика	12	A(3,12)	A(13,14)	5
A(13,14)	Проверка запроса к БД	3	A(12,13)	A(14,15)	5
A(14,15)	Проверка уровня доступа пользователя к желаемым данным	2	A(13,14)	A(15,16)	5
A(15,16)	Сравнение хэш-сумм необходимых данных для проверки целостности	2	A(14,15)	A(16,17)	5
A(16,17)	Отправка отчета для журналирования события	3	A(15,16)	-	5

Анализ сетевого графика функционирования ЛВС

Основными параметрами сетевого графика являются:

1. Наиболее раннее возможное время наступления j-го события T_p (j), вычисляемое по формуле:

$$T_P(j) = \frac{\max}{i \subset j}(T_P(i) - t_{ij})$$,

где - i и j обозначаются номера предшествующего и последующего событий соответственно;

- t_{ij} - продолжительность (i, j) -й работы.

Из обозначения $i \subset j$ следует, что событие i предшествует событию j.

Расчеты:

Tp(0) = 0
Tp(1) = 1
Tp(2) = 1
Tp(3) = 1
Tp(4) = 5
Tp(5) = 15
Tp(6) = 16
Tp(7) = 3
Tp(8) = 13
Tp(9) = 15
Tp(10) = 17
Tp(11) = 30
Tp(12) = 5
Tp(13) = 17
Tp(14) = 20
Tp(15) = 22
Tp(16) = 24
Tp(17) = 32

2. Самое позднее допустимое время наступления i-го события Tп (i), вычисляемое по формуле

$$T_П(j) = \frac{\min}{i \supset j}(T_П(i) - t_{ij})$$,

где из обозначения $i \supset j$ следует, что событие j предшествует событию i

Расчеты:

Tп(0) = 0
Tп(1) = 10
Tп(2) = 1
Tп(3) = 6
Tп(10) = 17
Tп(11) = 30
Tп(12) = 10
Tп(13) = 22

Тп(4) = 14
Тп(5) = 24
Тп(6) = 25
Тп(7) = 3
Тп(8) = 13
Тп(9) = 15
Тп(14) = 25
Тп(15) = 27
Тп(16) = 29
Тп(17) = 32

3. Резерв времени данного события Ri вычисляемый по формуле

$$R_i = (T_П(i) - T_P(i))$$

Расчеты:

R(0) = 0
R(1) = 9
R(2) = 0
R(3) = 5
R(4) = 9
R(5) = 9
R(6) = 9
R(7) = 0
R(8) = 0
R(9) = 0
R(10) = 0
R(11) = 0
R(12) = 5
R(13) = 5
R(14) = 5
R(15) = 5
R(16) = 5
R(17) = 0

4. Полный резерв времени работы rп (i,j), вычисляемый по формуле

$$r_П(\mathrm{i,j}) = (T_П(j) - T_P(i) - t_{ij})$$

Расчеты:

rп(0,1) = 9
rп(0,2) = 0
rп(0,3) = 5
rп(1,4) = 9
rп(2,7) = 0
rп(3,12) = 5
rп(4,5) = 9
rп(5,6) = 9
rп(6,17) = 9
rп(10,11) = 0
rп(11,17) = 0
rп(12,13) = 5
rп(13,14) = 5
rп(14,15) = 5
rп(15,16) = 5
rп(16,17) = 5

rп(7,8) = 0

rп(8,9) = 0

rп(9,10) = 0

5. Свободный резерв времени работы rc(i,j), вычисляется по формуле

$$R^{C}_{i-j} = Tп_i - t_{i-j} - Tp_i$$

Расчеты:

rc(0,1) = 0

rc(0,2) = 0

rc(0,3) = 0

rc(1,4) = 0

rc(2,7) = 0

rc(3,12) = 0

rc(4,5) = 0

rc(5,6) = 0

rc(6,17) = 9

rc(7,8) = 0

rc(8,9) = 0

rc(9,10) = 0

rc(10,11) = 0

rc(11,17) = 0

rc(12,13) = 0

rc(13,14) = 0

rc(14,15) = 0

rc(15,16) = 0

rc(16,17) = 5

Оценивание характеристик работы системы:

За единицу времени выполнения одной задачи в рамках сетевого графика принимается 12 минут (Тр). Также учитывается общий объём работ за день, например, 50 задач (Nп) за сутки.

При формировании перечня работ, которые могут привести к возникновению угрозы, сделаем допущение, что система мониторинга (АПК) спроектирована и установлена корректно. За критическое время проявления проблемы при штатной эксплуатации оборудования мониторинга примем 66 секунд (Тпп).

Для этапа мониторинга, согласно расчётам, существует один критический путь. Он определяется через операции, у которых полный резерв времени равен

нулю. Длина этого критического пути для мониторинга составила 17 секунд (Тип).

Для этапа нейтрализации угрозы, на основе полученных данных, также выявлен один критический путь. Он рассчитывается как максимальная продолжительность цепочки работ с нулевым полным резервом времени. Длина критического пути для устранения возникшей проблемы в заданных условиях равна 40 секунд (Тнп).

2.3 Оценивание показателя эффективности реализации управленческих решений ЛВС.

Эффективность - это свойство системы, отражающее меру достижения поставленной цели или реализации её проектного потенциала при заданных ограничениях. Данная характеристика количественно выражается с помощью специального показателя.

Поскольку основная задача управленческого решения заключается в анализе текущей ситуации и формировании команд для мобилизации ресурсов, логично выбрать в качестве показателя эффективности вероятность своевременного выявления и нейтрализации каждой проблемы, возникающей перед лицом, принимающим решение (ЛПР).

Показателем эффективности нашей модели управления процессом управления ЛПР УЗВО, будет служить следующая аналитическая зависимость:

P2=f($\lambda, v1, v2, v3, \xi^{+}, \xi^{-}$)

где λ - есть величина $\lambda = (\frac{1}{\Delta t\text{пп}})$, где Δtпп - среднее время проявление проблемы;

v1 - есть величина v1=($\frac{1}{\Delta t\text{ип}}$), где Δtип - среднее время идентификации проблемы;

v2 - есть величина v2=($\frac{1}{\Delta t\text{нп}}$), где Δtнп - среднее время нейтрализации проблемы;

v3 - частота срыва нейтрализации проблемы ЛПР, по причине невозможности распознать ситуацию (показатель квалификации ЛПР);

ξ^+ - есть величина $\xi^+ = \frac{1}{Тэ}$, где Тэ - длительность решения задачи;

ξ^- - частота срыва работы (невыполнение задачи);

Р2 - показатель эффективности реализации управленческих решений ЛПР.

По условию задачи имеем:

Тэ =10 - время решения задач;

N1 = 1 - количество срывов за 1 сутки;

N2 = 50 - количество работ;

Δtпп = 66; $\lambda = (\frac{1}{66}) = 0.015$

Δtип = 17; v1=$(\frac{1}{17}) = 0.058$

Δtнп = 40; v2=$(\frac{1}{40}) = 0.025$

$$\xi^- = \frac{N1}{N2} = \frac{1}{50} = 0.02$$

$$\xi^+ = \frac{1}{10} = 0.1$$

Задаем условие:

$$\frac{\Delta t\text{ип}+\Delta t\text{нп}}{\Delta t\text{пп}} < 1; \frac{17+40}{66} < 1$$

Найдем необходимую нам вероятность того, что задача будет выполнена Р2, с учетом частоты ξ^- - которая характеризует среднее количество срыва выполнения управленческих решений, что показывает успешность выполнения функционирования БДЗ.

Вероятности нахождения системы в состояниях *S1, S2, S3, S4*:

Если $v3 = \frac{v1}{1000}$, $\xi^- = \frac{N1}{N2} = \frac{1}{50} = 0.02$, то Р2=0.7558

$$P_1 = \frac{\nu_1 \cdot \nu_2 \cdot \zeta^- + \nu_1 \cdot \nu_3 \cdot \zeta^-}{\lambda \cdot \nu_1 \cdot \nu_2 + \lambda \cdot \nu_1 \cdot \zeta^- + \lambda \cdot \nu_3 \cdot \zeta^- + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \nu_2 \cdot \zeta^- + \nu_1 \cdot \zeta^+ \cdot \nu_3 + \nu_1 \cdot \nu_2 \cdot \zeta^-}$$

$$P_2 = \frac{\lambda \cdot \nu_1 \cdot \nu_2 + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \zeta^+ \cdot \nu_3}{\lambda \cdot \nu_1 \cdot \nu_2 + \lambda \cdot \nu_1 \cdot \zeta^- + \lambda \cdot \nu_3 \cdot \zeta^- + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \nu_2 \cdot \zeta^- + \nu_1 \cdot \zeta^+ \cdot \nu_3 + \nu_1 \cdot \nu_2 \cdot \zeta^-}$$

$$P_3 = \frac{\lambda \cdot \nu_1 \cdot \zeta^- + \lambda \cdot \nu_3 \cdot \zeta^-}{\lambda \cdot \nu_1 \cdot \nu_2 + \lambda \cdot \nu_1 \cdot \zeta^- + \lambda \cdot \nu_3 \cdot \zeta^- + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \nu_2 \cdot \zeta^- + \nu_1 \cdot \zeta^+ \cdot \nu_3 + \nu_1 \cdot \nu_2 \cdot \zeta^-}$$

$$P_4 = \frac{\lambda \cdot \nu_1 \cdot \zeta^-}{\lambda \cdot \nu_1 \cdot \nu_2 + \lambda \cdot \nu_1 \cdot \zeta^- + \lambda \cdot \nu_3 \cdot \zeta^- + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \nu_2 \cdot \zeta^- + \nu_1 \cdot \zeta^+ \cdot \nu_3 + \nu_1 \cdot \nu_2 \cdot \zeta^-}$$

Проверяем условие P1+P2+P3+P4=0.1221+0.7558+0,0611+0,0610=1.

Если $v3 = \frac{v1}{100}$, $\xi^- = \frac{N1}{N2} = \frac{1}{50} = 0.02$ то P2= 0.7565.

$$P_1 = \frac{\nu_1 \cdot \nu_2 \cdot \zeta^- + \nu_1 \cdot \nu_3 \cdot \zeta^-}{\lambda \cdot \nu_1 \cdot \nu_2 + \lambda \cdot \nu_1 \cdot \zeta^- + \lambda \cdot \nu_3 \cdot \zeta^- + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \nu_2 \cdot \zeta^- + \nu_1 \cdot \zeta^+ \cdot \nu_3 + \nu_1 \cdot \nu_2 \cdot \zeta^-}$$

$$P_2 = \frac{\lambda \cdot \nu_1 \cdot \nu_2 + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \zeta^+ \cdot \nu_3}{\lambda \cdot \nu_1 \cdot \nu_2 + \lambda \cdot \nu_1 \cdot \zeta^- + \lambda \cdot \nu_3 \cdot \zeta^- + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \nu_2 \cdot \zeta^- + \nu_1 \cdot \zeta^+ \cdot \nu_3 + \nu_1 \cdot \nu_2 \cdot \zeta^-}$$

$$P_3 = \frac{\lambda \cdot \nu_1 \cdot \zeta^- + \lambda \cdot \nu_3 \cdot \zeta^-}{\lambda \cdot \nu_1 \cdot \nu_2 + \lambda \cdot \nu_1 \cdot \zeta^- + \lambda \cdot \nu_3 \cdot \zeta^- + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \nu_2 \cdot \zeta^- + \nu_1 \cdot \zeta^+ \cdot \nu_3 + \nu_1 \cdot \nu_2 \cdot \zeta^-}$$

$$P_4 = \frac{\lambda \cdot \nu_1 \cdot \zeta^-}{\lambda \cdot \nu_1 \cdot \nu_2 + \lambda \cdot \nu_1 \cdot \zeta^- + \lambda \cdot \nu_3 \cdot \zeta^- + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \nu_2 \cdot \zeta^- + \nu_1 \cdot \zeta^+ \cdot \nu_3 + \nu_1 \cdot \nu_2 \cdot \zeta^-}$$

Проверяем условие P1+P2+P3+P4=0,1217+0.7565+0,0609+0,0607=1.

Если $v3 = \frac{v1}{1000}$, $\xi^- = \frac{N1}{N2} = \frac{2}{50} = 0.04$ то P2= 0.6075

$$P_1 = \frac{\nu_1 \cdot \nu_2 \cdot \zeta^- + \nu_1 \cdot \nu_3 \cdot \zeta^-}{\lambda \cdot \nu_1 \cdot \nu_2 + \lambda \cdot \nu_1 \cdot \zeta^- + \lambda \cdot \nu_3 \cdot \zeta^- + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \nu_2 \cdot \zeta^- + \nu_1 \cdot \zeta^+ \cdot \nu_3 + \nu_1 \cdot \nu_2 \cdot \zeta^-}$$

$$P_2 = \frac{\lambda \cdot \nu_1 \cdot \nu_2 + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \zeta^+ \cdot \nu_3}{\lambda \cdot \nu_1 \cdot \nu_2 + \lambda \cdot \nu_1 \cdot \zeta^- + \lambda \cdot \nu_3 \cdot \zeta^- + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \nu_2 \cdot \zeta^- + \nu_1 \cdot \zeta^+ \cdot \nu_3 + \nu_1 \cdot \nu_2 \cdot \zeta^-}$$

$$P_3 = \frac{\lambda \cdot \nu_1 \cdot \zeta^- + \lambda \cdot \nu_3 \cdot \zeta^-}{\lambda \cdot \nu_1 \cdot \nu_2 + \lambda \cdot \nu_1 \cdot \zeta^- + \lambda \cdot \nu_3 \cdot \zeta^- + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \nu_2 \cdot \zeta^- + \nu_1 \cdot \zeta^+ \cdot \nu_3 + \nu_1 \cdot \nu_2 \cdot \zeta^-}$$

$$P_4 = \frac{\lambda \cdot \nu_1 \cdot \zeta^-}{\lambda \cdot \nu_1 \cdot \nu_2 + \lambda \cdot \nu_1 \cdot \zeta^- + \lambda \cdot \nu_3 \cdot \zeta^- + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \nu_2 \cdot \zeta^- + \nu_1 \cdot \zeta^+ \cdot \nu_3 + \nu_1 \cdot \nu_2 \cdot \zeta^-}$$

Проверяем условие P1+P2+P3+P4=0,1958+0.6075+0,0979+0,0979=1.

Если $v3 = \frac{v1}{100}$, $\xi^- = \frac{N1}{N2} = \frac{2}{50} = 0.04$ то P2 = 0.6084.

$$P_1 = \frac{\nu_1 \cdot \nu_2 \cdot \zeta^- + \nu_1 \cdot \nu_3 \cdot \zeta^-}{\lambda \cdot \nu_1 \cdot \nu_2 + \lambda \cdot \nu_1 \cdot \zeta^- + \lambda \cdot \nu_3 \cdot \zeta^- + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \nu_2 \cdot \zeta^- + \nu_1 \cdot \zeta^+ \cdot \nu_3 + \nu_1 \cdot \nu_2 \cdot \zeta^-}$$

$$P_2 = \frac{\lambda \cdot \nu_1 \cdot \nu_2 + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \zeta^+ \cdot \nu_3}{\lambda \cdot \nu_1 \cdot \nu_2 + \lambda \cdot \nu_1 \cdot \zeta^- + \lambda \cdot \nu_3 \cdot \zeta^- + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \nu_2 \cdot \zeta^- + \nu_1 \cdot \zeta^+ \cdot \nu_3 + \nu_1 \cdot \nu_2 \cdot \zeta^-}$$

$$P_3 = \frac{\lambda \cdot \nu_1 \cdot \zeta^- + \lambda \cdot \nu_3 \cdot \zeta^-}{\lambda \cdot \nu_1 \cdot \nu_2 + \lambda \cdot \nu_1 \cdot \zeta^- + \lambda \cdot \nu_3 \cdot \zeta^- + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \nu_2 \cdot \zeta^- + \nu_1 \cdot \zeta^+ \cdot \nu_3 + \nu_1 \cdot \nu_2 \cdot \zeta^-}$$

$$P_4 = \frac{\lambda \cdot \nu_1 \cdot \zeta^-}{\lambda \cdot \nu_1 \cdot \nu_2 + \lambda \cdot \nu_1 \cdot \zeta^- + \lambda \cdot \nu_3 \cdot \zeta^- + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \nu_2 \cdot \zeta^- + \nu_1 \cdot \zeta^+ \cdot \nu_3 + \nu_1 \cdot \nu_2 \cdot \zeta^-}$$

Проверяем условие P1+P2+P3+P4=0,1958+0.6084+0,0978+0,0978=1.

Требуется рассчитать среднее время мониторинга и устранения проблемы при заданном времени проявления проблемы в системе для заданных вероятностей идентификации и нейтрализации проблемы: 0.8.

Для $P_2 = 0.8$

$$P_2 = \frac{\lambda \cdot \nu_1 \cdot \nu_2 + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \zeta^+ \cdot \nu_3}{\lambda \cdot \nu_1 \cdot \nu_2 + \lambda \cdot \nu_1 \cdot \zeta^- + \lambda \cdot \nu_3 \cdot \zeta^- + \nu_1 \cdot \nu_2 \cdot \zeta^+ + \nu_1 \cdot \nu_2 \cdot \zeta^- + \nu_1 \cdot \zeta^+ \cdot \nu_3 + \nu_1 \cdot \nu_2 \cdot \zeta^-}$$

Получаем:

$\Delta t_{ип}$= 9 сек.

$\Delta t_{нп}$=20 сек.

Чтобы обеспечить вероятность P, = 0.8, расчеты для идентификации и нейтрализации на каждом переходе события к событию должны быть:

1. При расчете графиков идентификации все данные необходимо откорректировать, примерно уменьшить в два раза.

2. При расчете графиков нейтрализации все данные надо, также откорректировать, например уменьшить в два раза.

Чтобы обеспечить бесперебойную работу локальной сети с помощью сетевого планирования можно увеличить время между процессами появления проблемы или уменьшить время идентификации или нейтрализации проблемы, а также нейтрализовать срыв в работе или срыв устранения проблемы ЛПР, по причине невозможности распознать ситуацию, при этом увеличится вероятность

реализации управленческих решений при управлении локальной сетью, что значительно улучшит процесс принятия управленческих решений в целом.

После анализа всех полученных результатов, получены результаты, к которым должна стремиться система обеспечения безопасности для выполнения условия P2 = 0.8. Для того чтобы их достичь необходимо разработать технологию, которая это обеспечит.

Вывод по второму разделу:

Для построения системы защиты информации от несанкционированного доступа (НСД) в условиях современных угроз использован системный подход. Его практическая реализация потребовала создания аналитической динамической модели процесса обеспечения безопасности ЛВС. В результате данная модель позволила определить ключевое условие устойчивого существования и функционирования системы защиты.

Достижение требуемого уровня защищенности локальной сети требует комплексной (системной) интеграции следующих взаимосвязанных процессов:

- проявление угрозы (инцидента НСД);
- выявление (детектирование) угрозы;
- нейтрализация (парирование) угрозы;
- профилактика угроз (упреждающие меры).

Такая структура позволяет оценивать уровень безопасности на основе способности системы к своевременному обнаружению и ликвидации попыток несанкционированного доступа.

Повышение уровня защищенности может быть достигнуто несколькими путями:

- сокращение времени реагирования и нейтрализации инцидентов и/или ускорение процессов мониторинга и обнаружения угроз.

-внедрение современных технических средств защиты, включая специализированное программное обеспечение (например, SIEM-системы, средства анализа трафика), доступные на рынке.

- регулярное проведение инструктажей и обучение персонала правилам информационной безопасности и корректной эксплуатации ресурсов локальной сети.

На основе системного подхода была составлена система алгебраических уравнений, описывающая взаимодействие процессов. Решение этой системы дало формализованное условие существования стабильного процесса защиты информации. Из данного условия было выведено уравнение, которое напрямую связывает итоговый показатель безопасности ЛВС с характеристиками четырёх ключевых процессов.

Показано, что сформулированное условие существования процесса защиты формируется на основе временных параметров выполнения работ по обеспечению ИБ. Эти временные характеристики, в свою очередь, зависят от текущих состояний каждого из четырёх процессов.

Для установления этой зависимости была решена задача связывания:

- временных параметров процессов,

- состояний этих процессов,

- конкретных работ, необходимых для перехода между состояниями,

и временных ресурсов на выполнение данных работ.

В качестве математического аппарата использованы сетевые модели. Расчёт критического пути в такой модели позволил количественно определить элементы ключевого условия существования системы защиты. Соблюдение этого условия, заданного критическим путём, обеспечивает достижение и поддержание требуемого уровня безопасности информации в локальной сети.

Разработанная аналитическая математическая модель послужила основой для перехода к этапу проектирования и разработки конкретной технологии защиты информации от НСД в рамках локальной вычислительной сети.

ГЛАВА 3. ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ТЕХНОЛОГИИ ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ ФУНКЦИОНИРОВАНИЯ ЛОКАЛЬНОЙ ВЫЧИСЛИТЕЛЬНОЙ СЕТИ

3.1. Проектирование системы обнаружения вторжений (IDS)

В результате разработанной аналитической-математической модели в условиях деструктивной среды и полученных расчетов следующий этап ВКР разработка технологии обеспечения безопасности.

Для начала нужно понимать что мы разрабатываем и при помощи каких средств.

Наиболее подходящим решением для реализации системы обнаружения вторжений (IDS) на Python является создание комплексной системы, объединяющей несколько мощных инструментов и библиотек для обеспечения всесторонней защиты локальной вычислительной сети (ЛВС). Такой подход позволит эффективно перехватывать, анализировать и реагировать на потенциальные угрозы.

Основные компоненты IDS на Python:

1. Scapy - используется для перехвата, формирования и анализа сетевого трафика. Благодаря его гибкости можно реализовать мониторинг пакетов, обнаружение аномалий и создание собственных правил обработки трафика.

2. Cryptography - обеспечивает шифрование передаваемых и хранящихся данных, что важно для защиты конфиденциальной информации и предотвращения ее перехвата злоумышленниками.

3. Python-nmap - служит для сканирования сети, выявления активных устройств, открытых портов и сервисов. Это помогает обнаруживать несанкционированные или подозрительные узлы в сети.

4. Yara-python - предназначена для обнаружения вредоносного программного обеспечения по сигнатурам. Можно создавать правила для поиска известных вредоносных образцов и автоматизировать их обнаружение.

Объединение этих компонентов позволяет создать мощную систему IDS, которая сможет не только обнаруживать атаки и вредоносное ПО, но и обеспечивать защиту данных посредством шифрования, а также регулярно отслеживать состояние сети. Такой комплексный подход значительно повысит уровень безопасности вашей ЛВС.

Внутренняя составляющая системы обнаружения вторжений

Внутренняя составляющая системы обнаружения вторжений (IDS) может значительно повысить эффективность за счет интеграции методов искусственного интеллекта (ИИ) и машинного обучения (МО). Эти технологии позволяют системе не только обнаруживать известные угрозы по сигнатурам, но и выявлять новые, ранее неизвестные атаки и аномалии с высокой точностью.

1. Модуль перехвата трафика (Packet Capture Module)

Использует библиотеки вроде Scapy для сбора пакетов в реальном времени.

2. Аналитический модуль (Traffic Analysis & Detection)

В дополнение к сигнатурному и аномальному анализу, внедряются модели машинного обучения:

- Обучение на исторических данных: используются алгоритмы классификации (например, случайный лес, градиентный бустинг, нейронные сети) для определения подозрительных паттернов.

- Обнаружение новых угроз: Модели способны выявлять аномалии, которые не попадают под заранее заданные сигнатуры, повышая уровень обнаружения новых видов атак.

- Обучение в режиме онлайн: система может адаптироваться на основе новых данных, улучшая точность со временем.

3. Модуль шифрования и защиты данных (Encryption & Data Security)

Обеспечивает безопасное хранение и передачу данных, используемых для обучения и анализа.

4. Модуль сканирования сети (Network Scanning Module)

Выполняет регулярные сканирования сети, собирает метаданные и признаки для обучения моделей.

5. Модуль обучения и обновления моделей (Model Training & Updating)

- Собирает и предварительно обрабатывает данные для обучения.

- Обновляет модели на основе новых данных, чтобы повысить их эффективность.

- Использует библиотеки ML (например, scikit-learn, TensorFlow, PyTorch).

6. Модуль оповещения и реагирования (Alerting & Response Module)**

- Генерирует оповещения на основе результатов анализа ML.

- Может запускать автоматические меры, например, блокировку IP или отключение сервиса.

7. Хранилище данных (Logging & Database)

- Хранит все перехваченные пакеты, метки, результаты анализа и обученные модели.

8. Интерфейс управления (Management Interface)

- Предоставляет инструменты для настройки системы, мониторинга работы моделей, просмотра логов и управления обучением.

Преимущества внедрения ИИ/МО:

- Повышенная точность обнаружения новых и сложных атак.

- Автоматическая адаптация к изменяющимся атакам и условиям сети.

- Снижение количества ложных срабатываний.

- Возможность анализа больших объемов данных и выявления скрытых закономерностей.

Этот подход позволяет сделать систему IDS более интеллектуальной, гибкой и способной противостоять современным угрозам, постоянно обучаясь и совершенствуясь.

Основные инструменты ИИ для кибербезопасности на Python включают популярные библиотеки, которые позволяют реализовать различные методы машинного обучения и анализа данных для повышения эффективности систем

обнаружения и предотвращения угроз. Ниже представлены ключевые библиотеки и их роли:

Популярные библиотеки:

1. Scikit-learn - Для классического машинного обучения: классификации, кластеризации, регрессии и оценки метрик. Хорошо подходит для построения моделей обнаружения аномалий, анализа поведения и предсказания угроз.

2. TensorFlow - Для создания и обучения нейронных сетей, особенно глубокого обучения. Используются для выявления сложных паттернов, анализа больших объемов данных, распознавания аномалий и обработки последовательностей.

3. PyTorch - Альтернативная библиотека для глубокого обучения, предоставляющая гибкий и динамический подход к построению нейросетевых моделей. Часто применяется для исследований и разработки новых архитектур.

4. Pandas - Для обработки и анализа структурированных данных. Используется для подготовки данных, очистки, фильтрации и преобразования перед обучением модели.

5. NumPy - Для математических и численных вычислений, работы с массивами и матрицами. Обеспечивает основу для большинства операций внутри других библиотек.

Эти инструменты позволяют разрабатывать системы, способные автоматически обнаруживать угрозы, анализировать поведение сети и пользователей, а также обучать модели для предсказания возможных атак и уязвимостей, повышая уровень кибербезопасности.

На рисунке 10 представлена схема работы системы обнаружения угроз с интегрированным ИИ.

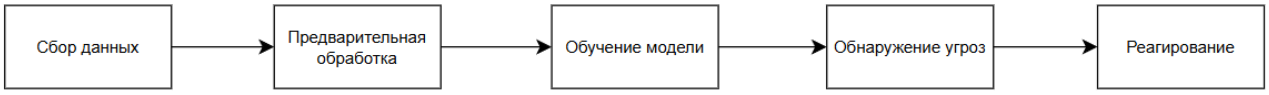

Рисунок 10 - схема работы системы обнаружения угроз с интегрированным ИИ

Подведение итогов по пункту 3.1

Рассматривается проектирование системы обнаружения вторжений (IDS) с использованием современных методов и инструментов. Основной акцент сделан на создании комплексной системы, объединяющей мощные библиотеки Python, такие как Scapy, Cryptography, Python-nmap и Yara-python, что позволяет реализовать всестороннюю защиту локальной сети. Такой подход обеспечивает не только обнаружение и анализ угроз, но и защиту данных посредством шифрования, а также регулярное мониторинг состояния сети.

Особое внимание уделяется внутренней архитектуре IDS с интеграцией методов искусственного интеллекта и машинного обучения. Это позволяет не только выявлять известные атаки по сигнатурам, но и обнаруживать новые, ранее неизвестные угрозы и аномалии, повышая эффективность системы. Внутренние модули включают перехват трафика, аналитическую обработку, обучение и обновление моделей, а также автоматические реакции на угрозы.

Использование библиотек машинного обучения, таких как scikit-learn, TensorFlow, PyTorch, Pandas и NumPy, позволяет создавать интеллектуальные модели, которые автоматически обучаются на собранных данных, адаптируются к изменяющейся обстановке и снижают число ложных срабатываний.

В целом, предложенная архитектура системы IDS с интеграцией ИИ обеспечивает высокий уровень автоматизации, точности и гибкости в обнаружении современных киберугроз.

3.2 Разработка системы обнаружения вторжений (IDS)

Для понимания всех процессов работы системы обнаружение необходимо структурировать весь код. Это позволит увидеть все процессы происходящие в системе.

На данном этапе осуществляется реализация структурированной и модульной системы обнаружения вторжений, основанной на интеграции современных инструментов и технологий. В качестве основы используется централизованный подход к сбору зависимостей, что реализуется через файл imports.py, включающий все необходимые библиотеки и модули для сетевого анализа, криптографии, машинного обучения и интерфейса пользователя.

В начале работы необходимо импортировать необходимы библиотеки, код представлен на рисунке 11.

```
" Базовые библиотеки
import os
import json
import logging
import threading
import time
from datetime import datetime

" Сетевой анализ
from scapy.all import sniff, IP, TCP, UDP, ICMP

" Сканирование сети
import nmap

" Шифрование
from cryptography.fernet import Fernet

" Анализ угроз
import yara

" Машинное обучение
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import IsolationForest, RandomForestClassifier
from sklearn.preprocessing import StandardScaler
import tensorflow as tf
from tensorflow import keras

" GUI (опционально)
import tkinter as tk
from tkinter import ttk, messagebox
```

Рисунок 11 - Импорт библиотек

Конфигурационные параметры системы задаются в файле config.py, что позволяет гибко адаптировать систему под конкретные требования сети без изменения исходного кода. В нем хранятся настройки интерфейса захвата трафика, пороги срабатываний, пути к логам, моделям и ключам шифрования (рис.12).

main.py

```python
CONFIG = {
    "interface": "eth0",  # Интерфейс для перехвата
    "scan_interval": 3600,  # Проверка сети раз в час
    "model_update_interval": 86400,  # Обновление модели раз в сутки
    "alert_threshold": 0.85,  # Порог срабатывания
    "log_file": "ids.log",
    "yara_rules": "rules.yar",
    "model_path": "ml_model.pkl",
    "encryption_key": b'...'  # Ключ шифрования
}
```

Рисунок 12 - Файл конфигурации

Модуль packet_capture.py отвечает за перехват сетевого трафика с помощью Scapy, извлекая из пакетов ключевые признаки, такие как IP-адреса, протокол, длина и временные метки. Эти данные служат сырьем для последующего анализа (рис.13)

```python
class PacketCapture:
    def __init__(self, interface):
        self.interface = interface
        self.packets = []

    def capture(self, count=100):
        """Перехват пакетов"""
        packets = sniff(iface=self.interface, count=count)
        self.packets.extend(packets)
        return packets

    def extract_features(self, packet):
        """Извлечение признаков из пакета"""
        return {
            'src_ip': packet[IP].src,
            'dst_ip': packet[IP].dst,
            'protocol': packet.proto,
            'length': len(packet),
            'timestamp': packet.time
        }
```

Рисунок 13 - Модуль перехвата трафика

Модуль network_scan.py реализует активное сканирование сети с помощью nmap для выявления скрытых устройств, открытых портов и потенциальных уязвимостей, дополняя пассивный мониторинг (рис.14)

```python
class NetworkScanner:
    def __init__(self):
        self.nm = nmap.PortScanner()

    def scan_hosts(self, subnet):
        """Сканирование подсети"""
        self.nm.scan(hosts=subnet, arguments='-sV -O')
        return self.nm.scaninfo()

    def get_open_ports(self, host):
        """Получение открытых портов"""
        return self.nm[host].get('tcp', {})
```

Рисунок 14 - Модуль сканирования сети

Для защиты данных применяется encryption.py, обеспечивающий шифрование логов и отчетов с использованием алгоритмов Fernet, что гарантирует их конфиденциальность и соответствие нормативным требованиям(рис.15).

```python
class DataEncryption:
    def __init__(self, key):
        self.cipher = Fernet(key)

    def encrypt(self, data):
        return self.cipher.encrypt(data.encode())

    def decrypt(self, token):
        return self.cipher.decrypt(token).decode()
```

Рисунок 15 - Модуль шифрования

Анализ угроз осуществляется через threat_analysis.py, где сигнатурные правила YARA комбинируются с моделями машинного обучения (ML), использующими библиотеки scikit-learn, TensorFlow и PyTorch. Это позволяет системе обнаруживать как известные угрозы, так и новые, ранее не фиксированные атаки, за счет обнаружения аномалий(рис.16).

```python
class ThreatAnalyzer:
    def __init__(self, yara_rules_path):
        self.rules = yara.compile(filepath=yara_rules_path)

    def check_yara(self, data):
        """Проверка по YARA-правилам"""
        matches = self.rules.match(data=data)
        return [match.rule for match in matches]

    def detect_anomalies(self, features):
        """Обнаружение аномалий через ML"""
        # Загрузка модели
        model = IsolationForest(contamination=0.1)
        # Предсказание
        prediction = model.predict([features])
        return prediction[0] == -1  # -1 = аномалия
```

Рисунок 16 - Модуль анализа угроз

Модуль ml_engine.py занимается обучением и применением моделей классификации, таких как RandomForest, для оценки вероятности угрозы и автоматической настройки порогов срабатывания, что снижает количество ложных тревог(рис.17).

```python
class MLModel:
    def __init__(self):
        self.model = None
        self.scaler = StandardScaler()

    def train(self, X, y):
        """Обучение модели"""
        X_scaled = self.scaler.fit_transform(X)
        self.model = RandomForestClassifier(n_estimators=100)
        self.model.fit(X_scaled, y)

    def predict(self, features):
        """Предсказание угрозы"""
        features_scaled = self.scaler.transform([features])
        prob = self.model.predict_proba(features_scaled)[0]
        return prob.max() > CONFIG["alert_threshold"]

    def save_model(self, path):
        """Сохранение модели"""
        import pickle
        with open(path, 'wb') as f:
            pickle.dump((self.model, self.scaler), f)

    def load_model(self, path):
        """Загрузка модели"""
        with open(path, 'rb') as f:
            self.model, self.scaler = pickle.load(f)
```

Рисунок 17 - Модуль машинного обучения

response_engine.py реализует механизмы автоматического реагирования: блокировку IP, изоляцию хоста и отправку оповещений, минимизирующих последствия атаки(рис.18).

```python
class ResponseEngine:
    def block_ip(self, ip):
        """Блокировка IP через iptables"""
        os.system(f"iptables -A INPUT -s {ip} -j DROP")

    def send_alert(self, threat_info):
        """Отправка уведомления"""
        logging.warning(f"УГРОЗА: {threat_info}")
        # Дополнительно: отправка email/SMS

    def isolate_host(self, host):
        """Изоляция хоста"""
        self.block_ip(host)
        self.send_alert(f"Хост {host} изолирован")
```

Рисунок 18 - Модуль реагирования

Основной движок системы - ids_engine.py - объединяет все компоненты в непрерывный цикл мониторинга. Он включает перехват пакетов, анализ признаков, проверку по сигнатурам и ML-моделям, а также запуск реактивных мер при обнаружении угроз. Этот модуль обеспечивает автоматическую, устойчивую работу системы в режиме реального времени.

```python
class IDSEngine:
    def __init__(self):
        self.capture = PacketCapture(CONFIG["interface"])
        self.scanner = NetworkScanner()
        self.analyzer = ThreatAnalyzer(CONFIG["yara_rules"])
        self.ml_model = MLModel()
        self.response = ResponseEngine()
        self.load_model()

    def load_model(self):
        if os.path.exists(CONFIG["model_path"]):
            self.ml_model.load_model(CONFIG["model_path"])

    def run_monitoring(self):
        """Основной цикл мониторинга"""
        while True:
            # 1. Перехват пакетов
            packets = self.capture.capture(count=50)

            for packet in packets:
                # 2. Извлечение признаков
                features = self.capture.extract_features(packet)

                # 3. Анализ угроз
                yara_matches = self.analyzer.check_yara(str(packet))
                is_anomaly = self.analyzer.detect_anomalies(features)
                ml_alert = self.ml_model.predict(features)

                # 4. Реагирование
                if yara_matches or is_anomaly or ml_alert:
                    threat_info = {
                        "timestamp": datetime.now(),
                        "src_ip": features["src_ip"],
                        "dst_ip": features["dst_ip"],
                        "details": yara_matches or "Аномалия/ML-срабатывание"
                    }
                    self.response.send_alert(threat_info)
                    self.response.isolate_host(features["src_ip"])

            time.sleep(10)  # Цикл каждые 10 сек
```

Рисунок 19 - Основной движок IDS

Точка входа - main.py - настраивает логирование и запускает цикл мониторинга, обеспечивая централизованный контроль и обработку ошибок(рис.20).

```python
from ids_engine import IDSEngine
import logging

if __name__ == "__main__":
    # Настройка логирования
    logging.basicConfig(
        filename=CONFIG["log_file"],
        level=logging.INFO,
        format='%(asctime)s %(levelname)s %(message)s'
    )

    # Запуск IDS
    ids = IDSEngine()
    ids.run_monitoring()
```

Рисунок 20 - Файл запуска

Также для большего удобства управления системой, разработана GUI. Которая поможет структурировать и визуализировать данные.

На рисунке 21 представлена рабочая визуальная оболочка системы обнаружения вторжений.

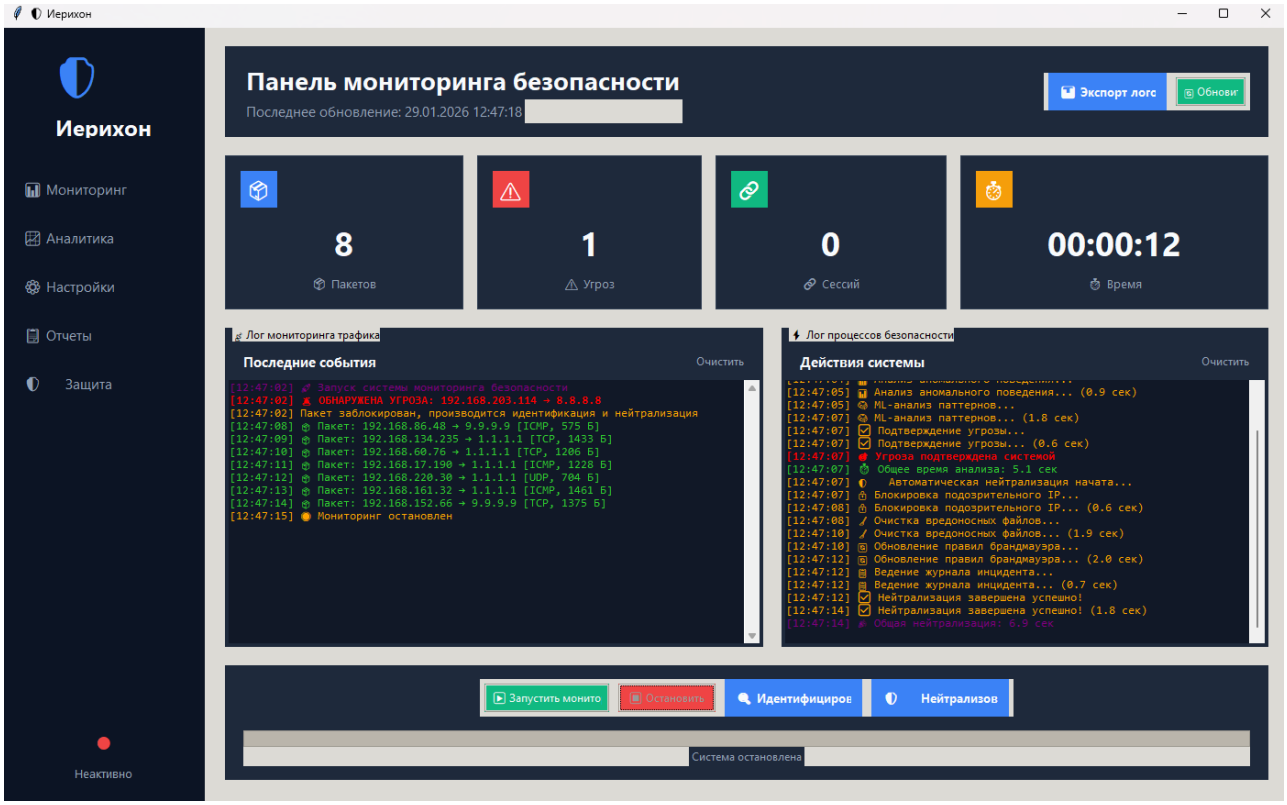

Рисунок 21 – GUI программы

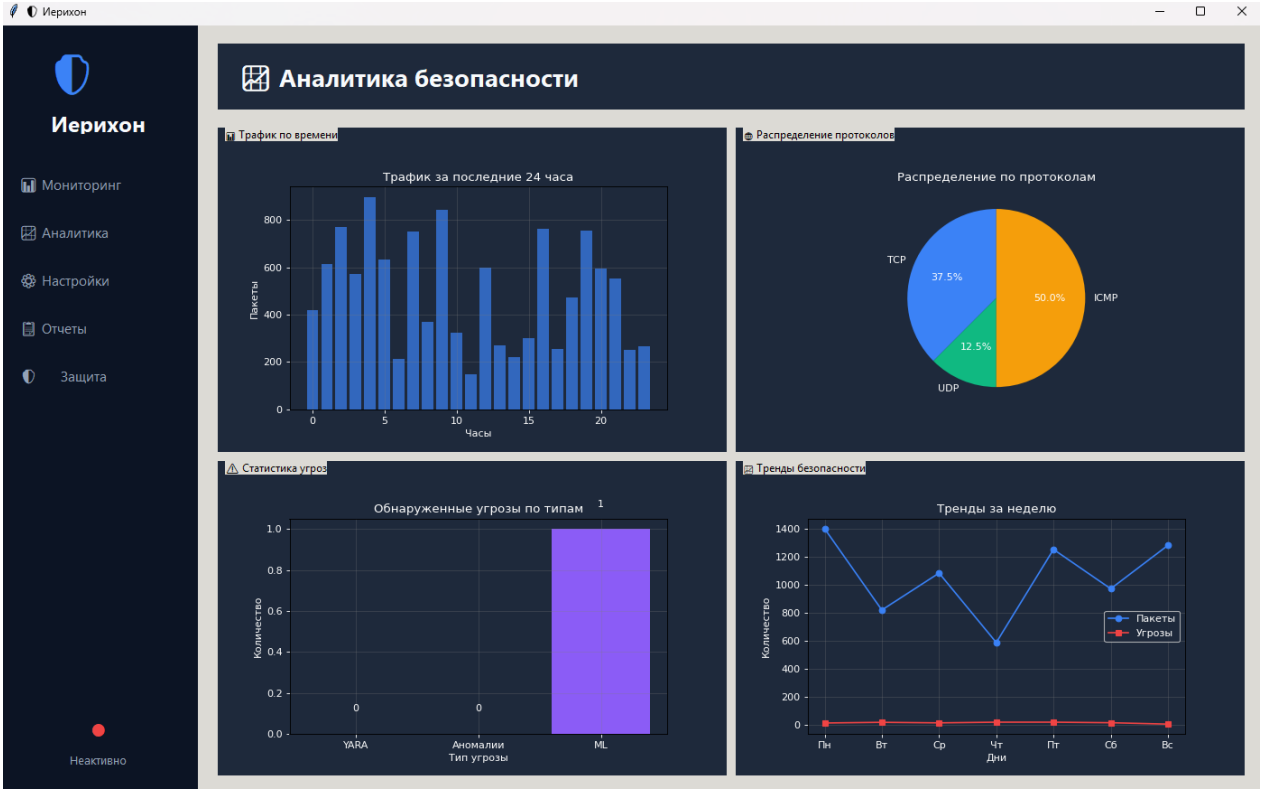

Рисунок 22 - Аналитика состояния сети

Схема работы системы

1. Захват пакетов:

Использует Scapy для захвата сетевых пакетов.

Захват пакетов периодически повторяется через заданный интервал.

2. Извлечение признаков:

Извлекает следующие признаки из захваченных пакетов:

- IP-адрес отправителя и получателя.

- порт отправителя и получателя.

- протокол (TCP, UDP, ICMP и т. д.).

- длина пакета.

Признаки сохраняются в базу данных для последующего анализа.

3. Анализ:

Проверка по YARA-правилам: система проверяет захваченные пакеты на соответствие YARA-правилам, чтобы выявить потенциальные опасные признаки.

ML-классификация (RandomForest): система использует Random Forest для классификации пакетов как безопасных или потенциально опасных на основе извлеченных признаков.

Обнаружение аномалий (IsolationForest): система использует Isolation Forest для выявления аномальных пакетов, которые могут указывать на потенциальную угрозу.

4. Решение:

Если угроза превышает заданный порог, система срабатывает и выполняет следующие действия:

-блокировка IP: система блокирует IP-адрес отправителя пакета.

- изоляция хоста: система изолирует хост, который отправил потенциально опасный пакет.

- оповещение: система отправляет оповещение о потенциальной угрозе.

5. Логирование:

Система логирования всех событий, включая захват пакетов, извлечение признаков, анализ, решение и реакцию.

6. Повтор цикла:

Система повторяет цикл через заданный интервал, чтобы обеспечить постоянный мониторинг и анализ сетевого трафика.

Преимущества архитектуры

Модульность: каждый компонент можно дорабатывать отдельно без влияния на другие части системы.

Масштабируемость: можно добавлять новые модели и правила без изменения существующей архитектуры.

Гибкость: конфигурацию системы можно менять через конфигурационный файл «config.py».

Комбинированный подход: система использует комбинацию сигнатурного анализа, ML-классификации и активного сканирования для обеспечения наиболее эффективной защиты от угроз.

Выводы по третьему разделу:

Получена четкая и структурированная концепция разработки системы обнаружения вторжений (IDS) на базе Python, включающую интеграцию современных инструментов и технологий. Также я ознакомился с перечнем основных компонентов системы - таких как Scapy, Cryptography, Python-nmap и Yara-python - и их ролью в обеспечении всесторонней защиты сети.

В процессе проектирования были учтены актуальные требования к безопасности, а также возможности современных библиотек и инструментов для автоматизации и повышения эффективности IDS. Использование модульного подхода, интеграция методов машинного обучения и сигнатурного анализа позволяют создавать гибкую, масштабируемую и эффективную систему, способную адаптироваться к новым угрозам.

Я буду использовать полученную концепцию и архитектуру для разработки собственной системы IDS, которая сможет перехватывать, анализировать и реагировать на сетевые угрозы в реальном времени. Модульность и гибкость позволят мне расширять и дорабатывать систему, внедрять новые модели машинного обучения, создавать новые правила и алгоритмы защиты. Также я буду использовать разработанную модель для обучения и тестирования методов обнаружения угроз, а полученные знания - для внедрения автоматических реакций, таких как блокировка IP и оповещение пользователей.

Этот подход позволит повысить уровень защиты сети, снизить количество ложных срабатываний и обеспечить постоянное обновление системы в соответствии с меняющимися условиями киберугроз

ГЛАВА 4. РАЗРАБОТАТЬ ПРЕДЛОЖЕНИЯ ПО СОВЕРШЕНСТВОВАНИЮ СИСТЕМЫ ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ ЛОКАЛЬНОЙ СЕТИ