

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ЯДЕРНЫЙ
УНИВЕРСИТЕТ «МИФИ»**

**ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ
«СКИЛФЭКТОРИ»**

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

**На тему: МЕТОД И СИСТЕМА СБОРА ПОВЕРХНОСТИ АТАКИ ДЛЯ
ВНЕШНЕГО ПЕРИМЕТРА ОРГАНИЗАЦИИ**

Выполнили:

Студент группы МИФИИБ-3

Студент группы МИФИИБ-3

Студент группы МИФИИБ-3

Студент группы МИФИИБ-3

Студент группы МИФИИБ-3

Студент группы МИФИИБ-3

Студент группы МИФИИБ-3

Ежов С. А.

Белоусов Д. С.

Баев В. А.

Картамышев М. С.

Иванов П. Ю.

Загородний А. В.

Выпирахин А. В.

Научный руководитель

Цуканов А. С.

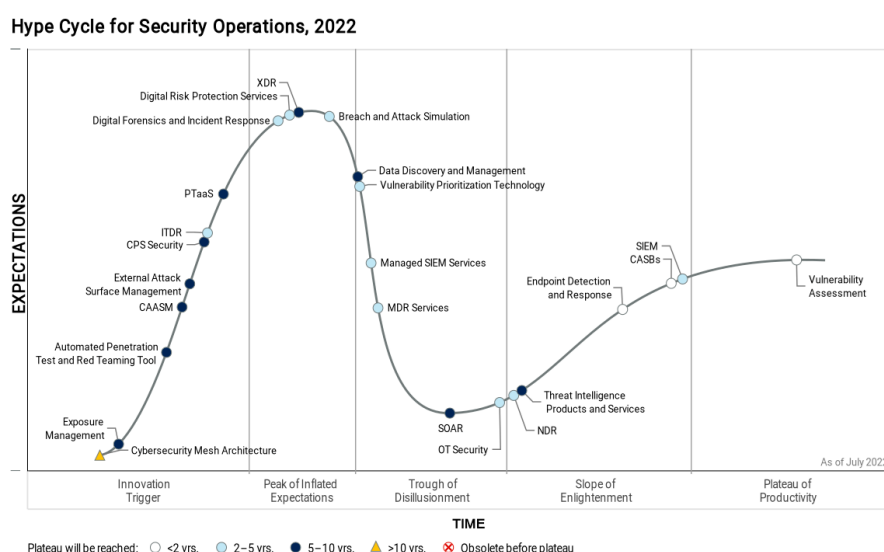
г. Москва, 2024

Оглавление

ВВЕДЕНИЕ.....	3
1 ИССЛЕДОВАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ.....	5
1.1 ПОСТАНОВКА ЗАДАЧИ	5
1.2 МЕТОДОЛОГИЯ	7
1.3 АНАЛИЗ И СРАВНЕНИЕ АНАЛОГОВ.....	11
2 ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ.....	16
2.1 ОПИСАНИЕ ИНФРАСТРУКТУРЫ	16
2.2 ЛОГИКА РАБОТЫ ПРОГРАММЫ-СКАНЕРА.....	24
2.3 РАБОТА ПРОГРАММЫ	29
3 РИСК-МЕНЕДЖМЕНТ.....	34
3.1 НОРМАТИВНО-ПРАВОВАЯ БАЗА ПО ЗАЩИТЕ КОНФИДЕНЦИАЛЬНЫХ ДАННЫХ.....	34
3.2 РИСКИ В СВЯЗИ С УТЕЧКОЙ КОНФИДЕНЦИАЛЬНЫХ ДАННЫХ	42
3.3 МИТИГАЦИЯ РИСКОВ	46
ЗАКЛЮЧЕНИЕ	50
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	51
ПРИЛОЖЕНИЕ.....	55

ВВЕДЕНИЕ

Современная модель экономики, которая постоянно развивается в цифровом поле, заставляет компании сталкиваться с проблемами в обеспечении собственной безопасности. Согласно обнародованному докладу организации Positive Technologies (www.ptsecurity.com, 2022), из 50 проектов 30 российских организаций, принимавшие участие в пентестах в 2021-2022 годах 96% оказались не защищены от проникновения в локальную сеть и взятия организации под полный контроль. Осознавая плачевность последствий для бизнеса, сотрудники организации сами стали тестировать и писать программы для изучения поверхности возможной атаки на свою организацию, а затем и нечто большее. Так возникло несколько успешных коммерческих продуктов. Например, Scanfactory. В это же время образуется большой пул программ класса EASM, призванных непрерывно анализировать ресурсы организации для предотвращения возможных проблем с утечками информации и взломом систем. Verizon в отчёте за 2022 год указала, что 70 % атак на компании совершаются через внешний периметр (www.verizon.com, 2022), а агентство Gartner назвало решения типа EASM трендом № 1 в кибербезопасности на ближайшие 5–10 лет (www.gartner.com, 2022) (рисунок 1).



Gartner

Рисунок 1. Hype Cycle of Security Operations, 2022 by Gartner.

Всего в мире существует 36 крупных вендоров EASM, сообщает Forrester в своём обзоре (reprints2.forrester.com, 2023). Процесс работы EASM-платформы состоит из четырёх этапов:

- Обнаружение и мониторинг активов. EASM собирает перечень доменов и IP-адресов компании. Инструменты фиксируют каждый найденный ресурс и ключевые данные о запущенных там сервисах. В результате мониторинга компания видит все свои активы, которые могут быть атакованы злоумышленниками.

- Поиск рисков. Все найденные активы проверяются на наличие уязвимостей и рисков, которыми может воспользоваться злоумышленник: возможности эксплуатации известных CVE, уязвимости из OWASP Top 10, ошибки конфигураций приложений, слабые пароли.

- Приоритезация. Не все найденные уязвимости будут опасны для компании, поэтому EASM ранжирует их от наиболее опасных к наименее опасным. Это позволяет понять, что нужно исправлять в первую очередь.

- Предложение исправлений. Для каждой найденной уязвимости инструмент EASM предлагает рекомендации.

Идея нашей ВКР представляет собой решение базового уровня для фирм и организаций широкого профиля, чтобы значительно повысить уровень безопасности собственных сервисов. Учитывая массовость и простоту применения, мы можем сделать вклад и значительно повысить общую безопасность бизнеса в России. Это является целью и задачей для широкого внедрения в практики инфраструктуры при старте и уже существующем бизнесе.

1 ИССЛЕДОВАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ

1.1 ПОСТАНОВКА ЗАДАЧИ

Сбор поверхности атаки - это процесс сбора информации об атаках на информационные системы. Метод и система сбора поверхности атаки могут включать в себя следующие шаги:

1. Определение целей: Необходимо определить, какую информацию вы планируете собирать о поверхности атаки. Цели могут включать в себя идентификацию потенциальных атакующих, обнаружение необычной активности или анализ уязвимостей в системе.

2. Сбор информации: Для сбора информации об атаках на информационные системы можно использовать различные методы, включая сетевое отслеживание, регистрацию событий, анализ журналов безопасности, мониторинг сетевого трафика и другие техники.

3. Анализ данных: После сбора информации необходимо проанализировать полученные данные, чтобы выявить потенциальные атаки или аномалии в работе системы. Это может включать в себя поиск необычных событий, анализ поведения пользователей или обнаружение уязвимостей в системе.

4. Реагирование на атаки: В случае обнаружения атаки или угрозы безопасности необходимо принять соответствующие меры по защите информационной системы. Это может включать в себя изоляцию уязвимостей, блокирование доступа для подозрительных пользователей или восстановление данных после атаки.

5. Улучшение безопасности: После анализа и реагирования на атаки важно проанализировать причины инцидентов и принять меры по улучшению системы безопасности. Это может включать в себя обновление политик безопасности, установку дополнительных мер защиты или обучение персонала по безопасности информационной системы.

Метод и система сбора поверхности атаки могут быть частью общей стратегии безопасности информационной системы и помочь предотвратить атаки, обеспечить защиту данных и обеспечить безопасность в целом.

1.2 МЕТОДОЛОГИЯ

Поверхность атаки в кибербезопасности включает в себя все слабые места и уязвимости, которые могут быть использованы злоумышленниками для атаки на информационные системы. Это могут быть недостатки в программном обеспечении, неактуальные обновления, недостаточная защита паролей, недостаточная защита сети, недостаточные меры безопасности при работе с API и т. д.

Понимание и анализ поверхности атаки позволяет организациям эффективно мониторить и защищать свои системы от потенциальных угроз. Меры по уменьшению поверхности атаки включают в себя улучшение защиты сети, установку обновлений безопасности, многофакторную аутентификацию, обучение сотрудников по безопасности информации, мониторинг активности сети и многое другое.

Эффективное управление поверхностью атаки помогает организациям предотвращать утечку конфиденциальных данных, вредоносные атаки и другие киберугрозы, обеспечивая более надежную защиту информационных ресурсов.

Ниже приведен список стандартных компонентов, которые можно считать частью "поверхности атаки":

- Программное и аппаратное обеспечение включает в себя все части системы, начиная от операционных систем и приложений, заканчивая серверами и сетевым оборудованием. Любая из этих частей может содержать уязвимости, которые могут быть использованы злоумышленниками для вторжения в систему.

- Сети также являются важной частью поверхности атаки. Незащищенные Wi-Fi сети, неправильно сконфигурированные брандмауэры и открытые порты могут предоставить злоумышленникам доступ к системе.

- Пользователи также составляют часть поверхности атаки, так как они могут быть обмануты при помощи социальной инженерии, такой как фишинг, для получения доступа к системам.

– Данные также могут быть уязвимыми частями поверхности атаки. Несанкционированный доступ к хранимым данным или перехват данных в пути может привести к утечкам информации и другим проблемам.

В целом, для обеспечения безопасности системы необходимо учитывать все эти компоненты и обеспечить защиту каждой из них от потенциальных угроз.

Поверхность атаки также определяется внешним и внутренним периметром:

– Внешний периметр организации - это ресурсы, доступные из интернета: веб-сайты, серверы, конечные устройства.

– Внутренний периметр - это локальная сеть и подключенные к ней стационарные устройства.

Возможные кейсы использования системы сбора поверхности атаки:

Для внешнего периметра

– Непрерывная инвентаризация облачных активов и контроль всех внешних точек входа

– Обнаружение уязвимых страниц входа, которые могут использоваться для кражи учетных данных

– Оценка уязвимости дочерних компаний и сторонних сервисов, связанных с основной ИТ-инфраструктурой

Для внутреннего периметра

– Анализ защищенности внутренней сети и выявление уязвимостей

– Демонстрация того, как злоумышленники могут использовать уязвимости информационных систем

– Предоставление рекомендаций по устранению выявленных уязвимостей

При этичном взломе поверхности атаки ищутся с помощью программного обеспечения, специально разработанного для этой цели; обычно проверяются различные данные, такие как права доступа к файлам, сетевые порты, запущенные процессы и многое другое.

Существует несколько методов и систем сбора информации о целевой сети для последующего проведения сетевой атаки:

- Сканирование сети: атакующий сканирует сеть цели с целью определения активных узлов и сервисов. Для этого могут использоваться инструменты, такие как Nmap, Masscan, Zenmap и другие. При сканировании можно получить информацию о портах, протоколах, открытых сервисах и т.д.

- Сбор информации о цели: атакующий может использовать различные открытые источники информации, такие как WHOIS базы данных, соцсети, форумы и т.д., для получения дополнительной информации о целевой организации, её сотрудниках, структуре сети и т.д.

- Фишинг: атакующий может использовать спам-письма, веб-сайты и другие методы для получения логинов и паролей от сотрудников целевой организации или для заражения их компьютеров вредоносным ПО.

- Сбор данных из открытых источников: атакующий может использовать открытые базы данных, поисковики и другие источники информации для получения дополнительных данных о цели, её сотрудниках, структуре сети и т.д.

- Техники социальной инженерии: атакующий может использовать методы манипуляции людьми для получения доступа к защищенной информации, например, путем обмана, угроз, вымогательства и других методов.

- Использование уязвимостей: атакующий может использовать известные уязвимости в программах и сервисах, установленных на серверах цели, для проведения атаки. Для этого могут использоваться специализированные инструменты, такие как Metasploit, Nessus и другие.

Общими для всех методов являются необходимость постоянного мониторинга целевой сети, поиск уязвимостей и слабых мест, а также анализ собранной информации для планирования атаки.

Сканирование сети для атакующего является важным этапом при сетевой атаке по следующим причинам:

- Определение уязвимостей: сканирование позволяет определить уязвимые узлы в сети, которые можно использовать для проникновения или атаки.
- Определение архитектуры сети: сканирование помогает атакующему понять структуру сети, настройку оборудования и связи между устройствами, что облегчает планирование атаки.
- Поиск целей: сканирование помогает определить цели атаки, такие как серверы, маршрутизаторы, межсетевые экраны и другие устройства, содержащие ценную информацию или уязвимые для атаки.
- Проведение разведки: сканирование помогает собрать информацию о целевой сети, такую как IP-адреса, открытые порты, используемые сервисы, версии программного обеспечения и другие данные, необходимые для успешного выполнения атаки.

В рамках ВКР будет применен метод сканирования сети. Выбор метода исследования обуславливается важностью данного этапа при подготовке перед проведением сетевой атаки и помогает атакующему понять среду, в которой он собирается действовать, а также увеличить вероятность успешного завершения атаки.

1.3 АНАЛИЗ И СРАВНЕНИЕ АНАЛОГОВ

Для оценки поверхности атаки часто используют специализированные инструменты, такие как сканеры уязвимостей, сетевые сканеры и поисковики в интернете. Каждый из этих инструментов имеет свою цель, поэтому необходимо понимать, для чего конкретно он применяется. Иногда для получения более полной и точной картины уязвимостей следует использовать несколько сканеров одновременно.

В рамках данной работы перечислим лишь некоторые сканеры и сервисы, которые используются для сканирования сетевых периметров и интернета. Подробнее рассмотрим сетевые сканеры, так как именно их функционал будем использовать в процессе написания ВКР.

Сетевые сканеры: Masscan , Zmap , nmap. В действительности утилит для сканирования сети намного больше, однако, для сканирования периметра вряд ли существует необходимость использовать другие. Данный набор утилит позволяет решить большинство задач, связанных со сканированием портов и служб.

Nmap – утилита для сканирования сетей, позволяет составить подробную карту, получить максимум информации о запущенных сервисах на хостах в сети, а также превентивно проверить некоторые уязвимости (nmap.org, 2024) (рисунок 2.1). Также имеет гибкие настройки сканирования:

- настройка скорости сканирования;
- количества потоков;
- количества групп для сканирования.

Удобен для сканирования небольших сетей и незаменим для точечного сканирования отдельных хостов.

```

root@kali:~# nmap -p22-200 -o 192.168.5.102

Starting Nmap 7.01 ( https://nmap.org ) at 2016-03-06 15:38 CET
Nmap scan report for 192.168.5.102
Host is up (0.31s latency).
Not shown: 173 closed ports
PORT      STATE SERVICE
53/tcp    open  domain
80/tcp    open  http
88/tcp    open  kerberos-sec
111/tcp   open  rpcbind
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
Device type: general purpose
Running: Microsoft Windows 7|2012|XP
OS CPE: cpe:/o:microsoft:windows_7 cpe:/o:microsoft:windows_server_2012 cpe:/o:microsoft:windows_xp::sp3
OS details: Microsoft Windows 7 or Windows Server 2012, Microsoft Windows XP SP3

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.10 seconds

```

Рисунок 2.1 – Применение утилиты Nmap

Zmap – сканер с открытым исходным кодом, создавался как более быстрая альтернатива Nmap (zmap.io, 2022) (рисунок 2.2). При сканировании не ждет, пока вернется ответ, а продолжает сканирование.

```

replicante:/home/jose # zmap -p 443 -o results.txt
Aug 20 14:04:49.354 [INFO] zmap: started
0:01 0%; send: 139718 140 Kp/s (138 Kp/s avg); recv: 20 19 p/s (19 p/s avg); dr
ops: 0 p/s (0 p/s avg); hits: 0.01%
0:02 0%; send: 282771 143 Kp/s (140 Kp/s avg); recv: 47 26 p/s (23 p/s avg); dr
ops: 0 p/s (0 p/s avg); hits: 0.02%
0:03 0%; send: 424889 142 Kp/s (141 Kp/s avg); recv: 71 23 p/s (23 p/s avg); dr
ops: 0 p/s (0 p/s avg); hits: 0.02%
0:04 0%; send: 552626 128 Kp/s (137 Kp/s avg); recv: 97 25 p/s (24 p/s avg); dr
ops: 0 p/s (0 p/s avg); hits: 0.02%
0:05 0% (7h25m left); send: 696299 144 Kp/s (139 Kp/s avg); recv: 130 32 p/s (2
5 p/s avg); drops: 0 p/s (0 p/s avg); hits: 0.02%
0:06 0% (7h22m left); send: 839796 143 Kp/s (139 Kp/s avg); recv: 155 24 p/s (2
5 p/s avg); drops: 0 p/s (0 p/s avg); hits: 0.02%
0:07 0% (7h20m left); send: 984162 144 Kp/s (140 Kp/s avg); recv: 184 28 p/s (2
6 p/s avg); drops: 0 p/s (0 p/s avg); hits: 0.02%
0:08 0% (7h19m left); send: 1127676 144 Kp/s (141 Kp/s avg); recv: 217 32 p/s (
27 p/s avg); drops: 0 p/s (0 p/s avg); hits: 0.02%

```

Рисунок 2.2 – Применение утилиты Zmap

Masscan – инструмент для быстрого сканирования сетей. Он может просканировать весь Интернет со скоростью передачи данных 25 миллионов запросов в секунду, менее чем за 6 минут (Github.com/robertdavidgraham/masscan, 2023).

```
kali@kali:~$ sudo masscan 172.217.167.46 -p443,80,4444
[sudo] password for kali:

Starting masscan 1.0.6 (http://bit.ly/14GZzcT) at 2020-09-11 06:01:04 GMT
-- forced options: -sS -Pn -n --randomize-hosts -v --send-eth
Initiating SYN Stealth Scan
Scanning 1 hosts [3 ports/host]
Discovered open port 80/tcp on 172.217.167.46
Discovered open port 443/tcp on 172.217.167.46
kali@kali:~$ █
```

Рисунок 2.3 – Применение утилиты Masscan

Ниже в таблице 1.1 представлена сравнительная характеристика сетевых сканеров.

Таблица 1.1 – Сравнительная характеристика сетевых сканеров

Инструмент	Плюсы	Минусы
Masscan	<ul style="list-style-type: none">– Один из самых быстрых асинхронных сканеров;– Возобновление прерванного сканирования, распределение нагрузки по нескольким устройствам;	<ul style="list-style-type: none">– Крайне высокая нагрузка на сеть;– По умолчанию нет возможности сканировать на прикладном уровне L7.
Zmap	<ul style="list-style-type: none">– Генерирует Ethernet-фреймы минуя системный стек TCP/IP;– Возможность использования PF_RING для быстрого сканирования больших сетей;– Равномерно сканирует сеть для распределения нагрузки.	<ul style="list-style-type: none">– Возможность появления отказа в обслуживании промежуточных маршрутизаторов.
Nmap	<ul style="list-style-type: none">– Быстро работает с небольшим диапазоном	<ul style="list-style-type: none">– Информация о каком-либо хосте недоступна, пока

	<p>хостов;</p> <ul style="list-style-type: none"> – Возможность комбинировать опции; – Возможность параллельного сканирования; – Предопределенные наборы скриптов для разных задач; – Вывод результатов в пяти различных форматах. 	<p>не закончится сканирование всей группы;</p> <ul style="list-style-type: none"> – При сканировании больших сетей с использованием флагов для ускорения сканирования может давать false-negative результаты, пропуская открытые порты на хосте; – Низкая производительность. При сканировании Nmap отправляет SYN-пакеты на целевой порт и ожидает любого ответного пакета или наступления таймаута, в случае когда ответа нет.
--	--	--

Поисковики по интернету вещей, или онлайн-сканеры — важные инструменты для сбора информации об интернете в целом. Они предоставляют следующую сводку:

- принадлежности узлов к организации;
- сведения о сертификатах;
- сведения об активных службах.

С разработчиками этого типа сканеров можно договориться об исключении ваших ресурсов из списка сканирования или о сохранении информации о ресурсах только для корпоративного пользования. Наиболее известные поисковики: Shodan , Censys , Fofa .

Для решения задачи не обязательно применять сложный коммерческий инструмент с большим числом проверок: это излишне для сканирования

пары «легких» приложений и сервисов. В таких случаях будет достаточно бесплатных сканеров, но так как их много, и тяжело выделить наиболее эффективные, то здесь выбор, скорее, дело вкуса. Наиболее известные: Skipfish , Nikto , ZAP , Acunetix , SQLmap .

При тщательном ручном анализе будут полезны инструменты Burp Suite, Metasploit и OpenVAS.

Следует также упомянуть о онлайн-поисковике уязвимостей Vulners. Это обширная база данных с информацией об уязвимостях с множеством источников, включая вендорские бюллетени безопасности, программы bug bounty и другие специализированные ресурсы. Ресурс предоставляет API для получения результатов, что позволяет проводить проверки без сканирования в реальном времени. Также доступен Vulners vulnerability scanner, который анализирует операционную систему, установленные пакеты и проверяет уязвимости через API. Некоторые функции ресурса являются платными.

2 ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ

2.1 ОПИСАНИЕ ИНФРАСТРУКТУРЫ

В целях имитации сбора поверхности атаки внешнего периметра нами была развёрнута виртуальная инфраструктура на базе системы виртуализации Proxmox VE.

Proxmox Virtual Environment (Proxmox VE) — система виртуализации с открытым исходным кодом, основанная на Debian GNU/Linux. Разрабатывается австрийской фирмой Proxmox Server Solutions GmbH, спонсируемой Internet Foundation Austria (www.proxmox.com, 2024).

Proxmox позволяет контролировать виртуальные и физические сервера без необходимости в ручной настройке. Proxmox работает как гипервизор 2-ого типа, что означает, что между сервером и Proxmox есть прослойка в виде хостовой операционной системы Debian, основанной на модульном ядре Linux (рисунок 3.1).

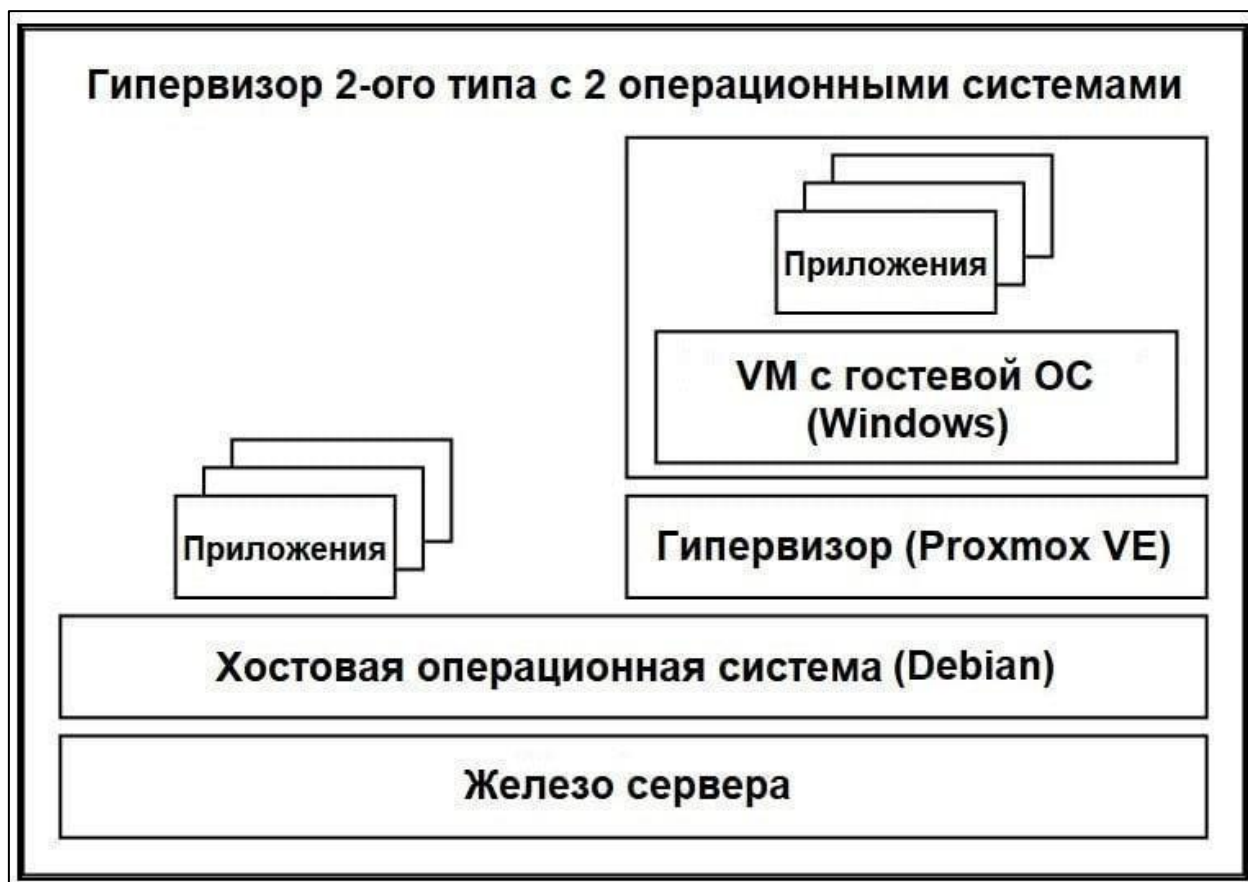


Рисунок 3.1 – Схема работы гипервизора 2-го типа

Основная цель Proxmox — упростить быстрое масштабирование ИТ-инфраструктуры без дорогостоящей модернизации оборудования и приобретения лицензий на программное обеспечение. Пользователи легко настраивают виртуальные машины, пулы хранения общих файлов, дисков и сетевых дисков. Платформа предлагает полный набор инструментов для создания образов для локального и удаленного использования и поддерживает различные дистрибутивы Linux.

Proxmox упрощает задачу мониторинга, позволяя легко отслеживать уровень загрузки ресурсов, включая скорость использования процессора всеми виртуальными машинами, а также потребление полосы пропускания всей сетью. Это помогает сразу обнаружить неполадки с производительностью, пока они не переросли в серьёзные проблемы.

Интегрированная в платформу система резервного копирования обеспечивает сохранность важных данных от случайного удаления или аппаратных сбоев с быстрым восстановлением в случае их непредвиденного возникновения. Это гарантирует непрерывность бизнеса вне зависимости от внешних событий, то есть тех, на которые пользователь не может повлиять.

Таблица 2.1 – Плюсы и минусы Proxmox VE

Гипервизор:	Плюсы гипервизора:	Минусы гипервизора:
Proxmox VE	Бесплатность и открытый исходный код	Сложность установки на программный RAID (mdadm)
	Встроенная система бэкапов и снапшотов	Необходимость использования сторонних репозиториев для обновлений
	Удобная веб-панель управления	Отсутствие официальной поддержки и документации на

		русском языке
	Возможность использования разных типов хранилищ (локальные, NFS, ZFS и т.д.)	Низкая популярность по сравнению с другими системами виртуализации
	Поддержка USB-проброса и других расширенных функций	Неудобство работы с сетевыми настройками
	Поддержка разных типов виртуальных машин (KVM и LXC)	Отсутствие готовых шаблонов для LXC-контейнеров
	Возможность создавать кластеры и высокодоступные системы	Необходимость использования сторонних скриптов для некоторых функций
	Поддержка новых версий Ceph и Btrfs	Низкая совместимость с другими системами виртуализации
	Интеграция с различными облачными сервисами	

В связи с выше обозначенными возможностями система виртуализации Proxmox VE полностью удовлетворяет требования для организации виртуальной инфраструктуры в целях учебного проекта. С её помощью мы сможем развернуть необходимые компоненты для имитации сбора поверхности атаки периметра организации.

После выбора системы виртуализации и настройки гипервизора для работы с виртуальной средой (рисунок 3.2) мы можем переходить к добавлению уязвимых компонентов в нашу инфраструктуру.

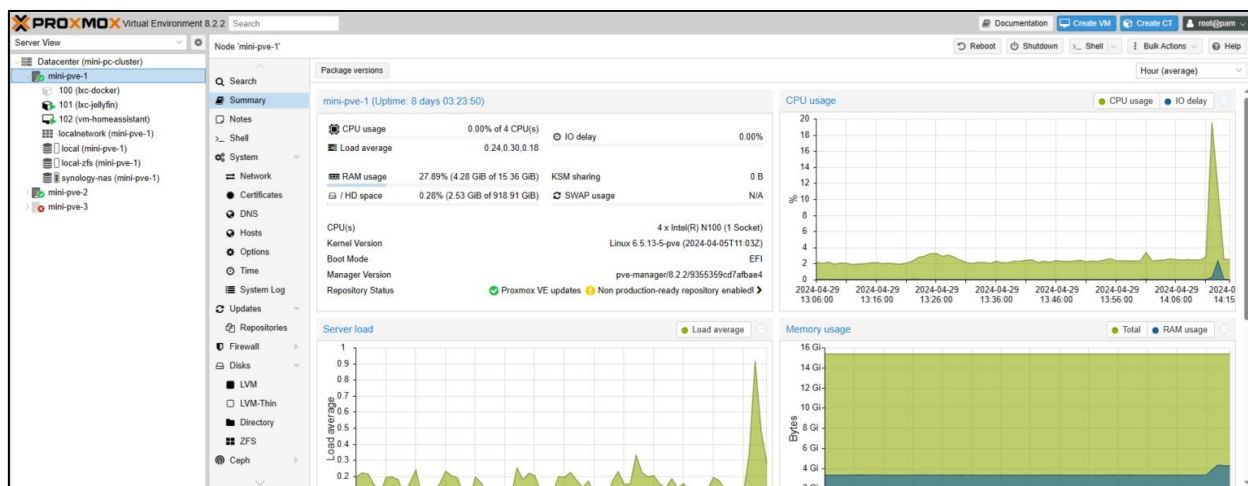


Рисунок 3.2 – Главное окно гипервизора Proxmox VE

Одним из компонентов виртуальной инфраструктуры возьмём готовый образ VM Metasploitable 2 (рисунок 3.3).

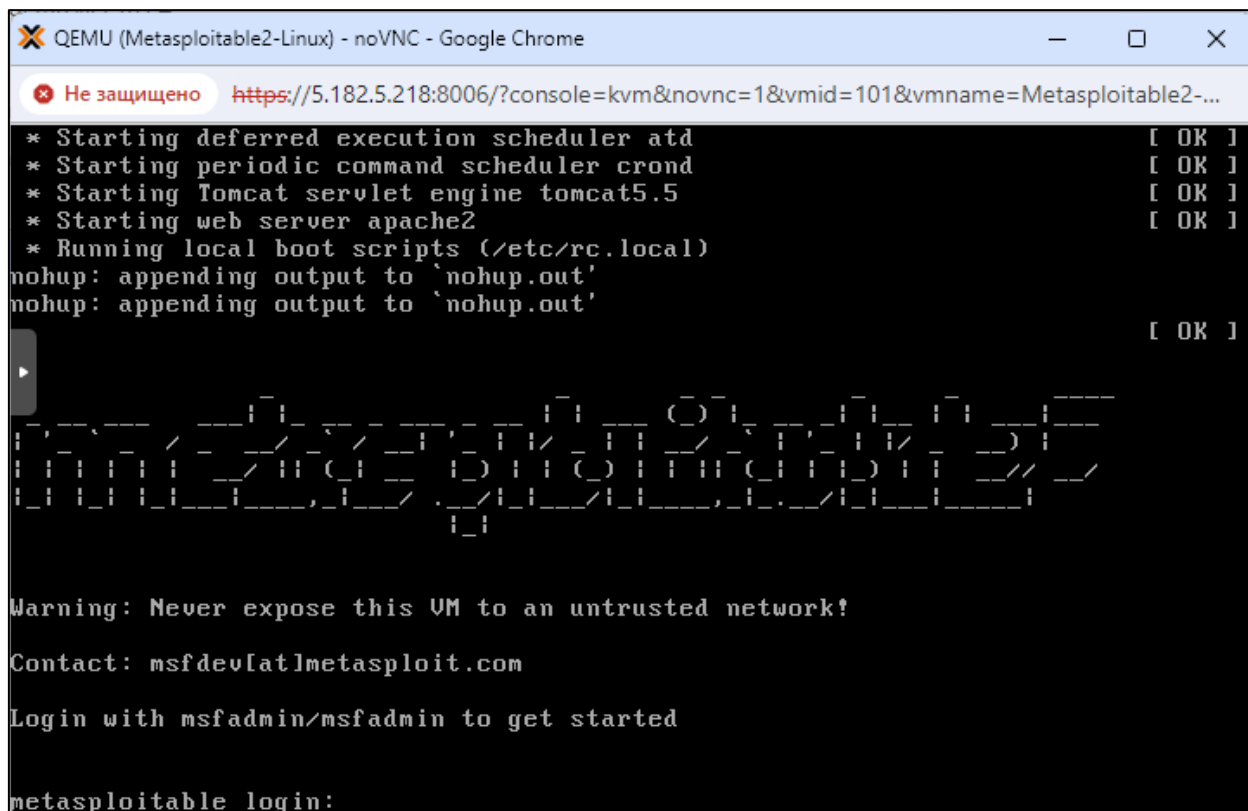


Рисунок 3.3 – Загрузочный экран Metasploitable 2

Metasploitable 2 – виртуальная машина на базе Ubuntu GNU/Linux, специально спроектированная на максимальную уязвимость для тестирования инструментов безопасности и демонстрации общих уязвимостей. Эта виртуальная машина совместима с VMWare, VirtualBox, и с другими общими виртуализированными платформами. По умолчанию, интерфейсы сетей Metasploitable привязаны к NAT и Host-only сетевым адаптерам, и образ не должен никогда подвергаться воздействию враждебной сети (docs.rapid7.com, 2024).

Вторым компонентом возьмём также готовый образ VM Breakout с портала VulnHub (рисунок 3.4).

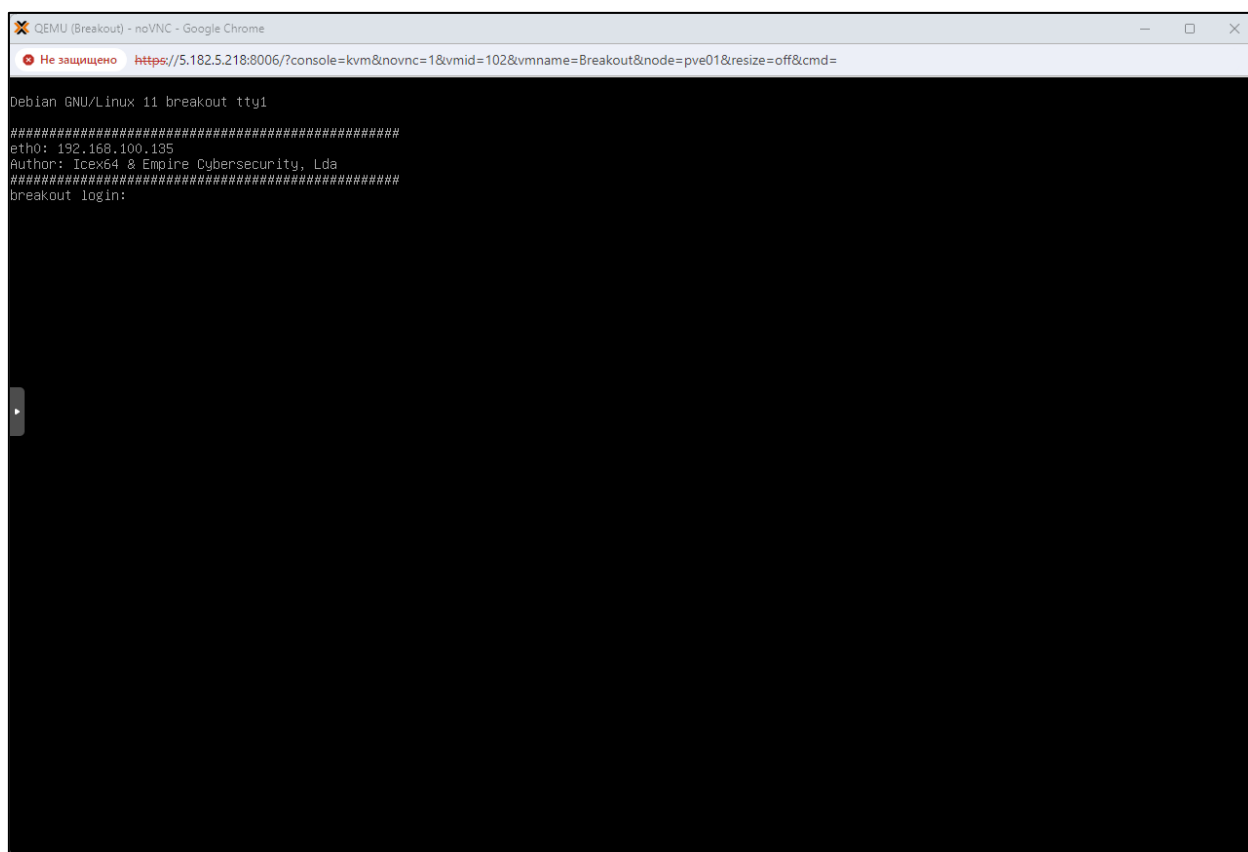


Рисунок 3.4 – Загрузочное экран Breakout

VulnHub — ресурс, предоставляющий образы операционных систем с сервисами, в которых «защиты» уязвимости. Скачав такой образ, любой желающий может получить опыт взлома или системного администрирования (www.vulnhub.com, 2024).

Третьим уязвимым компонентом сконфигурируем собственную виртуальную машину с уязвимым компонентом CMS WordPress (рисунок 3.5).

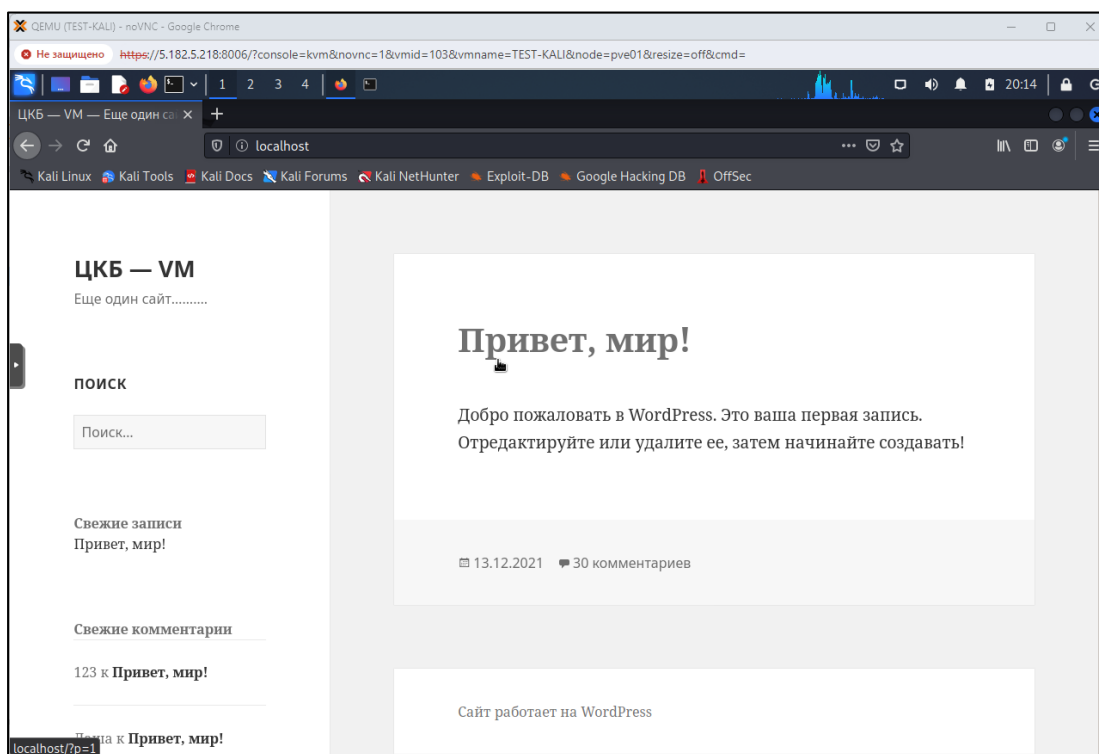


Рисунок 3.5 – Загрузочное окно VM с CMS WordPress

Последним компонентом поставим виртуальную машину, с которой мы будем осуществлять сбор поверхности атаки в имитации внешнего периметра. Для этих целей вполне подойдёт готовый образ VM Kali Linux (рисунок 3.6).

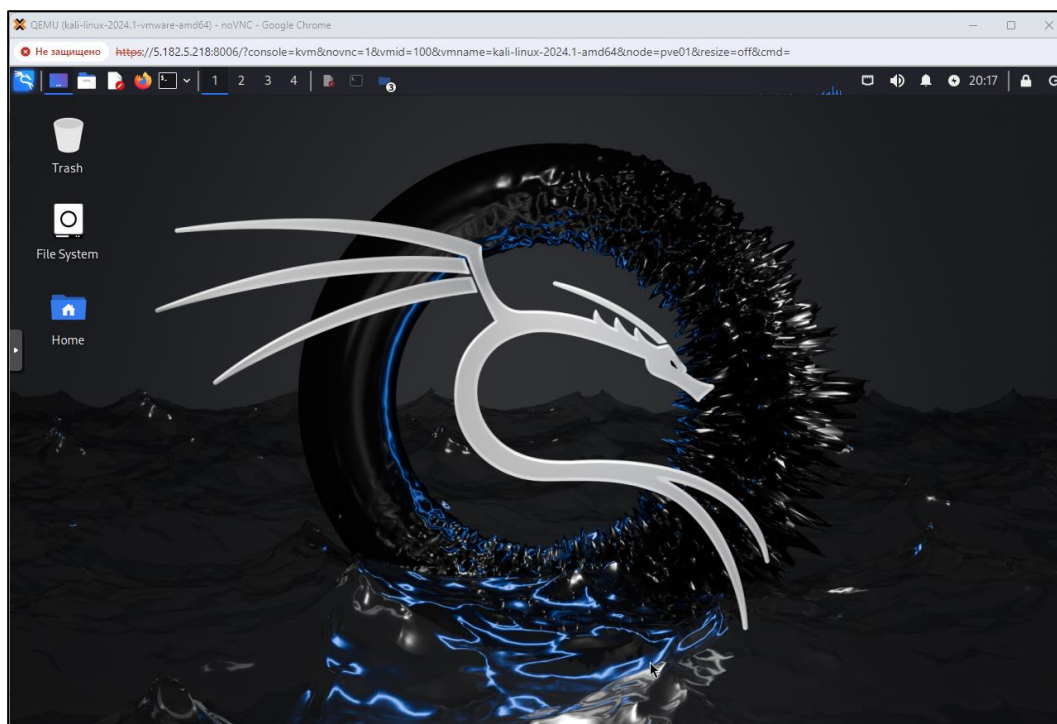


Рисунок 3.6 – Загрузочный экран Kali Linux

Kali Linux — это дистрибутив Linux на основе Debian с открытым исходным кодом, предназначенный для расширенного тестирования на проникновение, проверки уязвимостей, аудита безопасности систем и сетей. Дистрибутив используют многие специалисты информационной безопасности: сетевые архитекторы и администраторы, инженеры-криминалисты, директора по информационной безопасности, пентестеры (www.kali.org, 2024).

Kali Linux предлагает широкий набор предустановленных инструментов для сканирования сети, для проведения тестов на безопасность, анализа уязвимостей, взлома паролей, реверс-инжиниринга, а также для выполнения других операций, связанных с безопасностью информационных систем. Популярные утилиты — Nmap, Metasploit Framework, Wireshark, Aircrack-ng, John the Ripper, Burp Suite, Maltego и другие.

На этом подготовку виртуальной инфраструктуры можно считать законченной. Так как данные виртуальные машины имеют критические уязвимости, то было решено разместить их за NAT Proxmox VE, это

обеспечит сокрытие наших учебных машин от злоумышленников. Кроме того, чтобы упростить дальнейшую настройку и разворачивания дополнительных компонентов на Proxmox VE был настроен сервер DHCP для раздачи IP-адресов нашим ВМ. Это позволит довольно гибко менять состав и сетевую конфигурацию нашей виртуальной инфраструктуры.

2.2 ЛОГИКА РАБОТЫ ПРОГРАММЫ-СКАНЕРА

Нам необходимо написать программу, которая будет собирать поверхность атаки на заданном периметре. Программу будет писать на языке программирования Python.

Python – это высокоуровневый язык программирования, который был разработан в конце 1980-х годов. Его разработчик, Гвидо ван Россум, вложил в основу языка простоту и читабельность кода, что позволяет использовать Python для быстрой и эффективной разработки. Много популярных веб-сайтов, компьютерных игр и программ, написанных на Python, вы используете ежедневно: Dropbox, Uber, Sims, Google, GIMP и другие (www.python.org, 2024).

Язык отличается понятным синтаксисом, поэтому Python подходит для начинающих программистов. Он широко используется во многих областях: веб-разработка, научные исследования, анализ данных, искусственный интеллект, машинное обучение, разработка игр.

У Python большая библиотека сторонних модулей и инструментов, что делает его мощным инструментом. Наличие активного сообщества разработчиков позволяет постоянно поддерживать и обновлять язык, предоставлять достаточный объем обучающих материалов, документацию и форумы для программистов с любым уровнем знаний.

Таблица 2.2 – Плюсы и минусы языка программирования Python

Язык программирования:	Плюсы языка:	Минусы языка:
Python	Простота и воспринимаемость	Низкая производительность
	Обширная библиотека	Глобальная блокировка интерпретатора (GIL)
	Совместимость	Синтаксис
	Мультиплатформенность	
	Мультипарадигмальность	

Как видим, данный язык программирования прекрасно подходит для требуемых задач в рамках учебного проекта. Для организации работы с сетевыми протоколами мы будем использовать библиотеку Scapy.

Scapy – интерактивная оболочка и программная библиотека для манипулирования сетевыми пакетами на языке программирования Python. Scapy написана Филиппом Бионди в 2003 году и распространяется под лицензией GPLv2 (scapy.net, 2024).

Scapy — самая популярная библиотека для пентестинга и разработки инструментов безопасности. Она предоставляет потрясающие возможности для отправки, прослушивания (сниффинга) или подделки пакетов данных. Пользователи могут использовать ее в интерактивном режиме или импортировать напрямую. Scapy предоставляет функциональные возможности nmap, arpspoof, wireshark и многих других инструментов для сканирования сетей, которые используются на начальном этапе пентестинга.

Данная библиотека обладает богатым потенциалом, но в нашем проекте мы ограничимся проверкой доступности указанных портов сетевых узлов. Для этого будем использовать трёхстороннее рукопожатие (TCP handshake) (afteracademy.com, 2020) (рисунок 3.7).

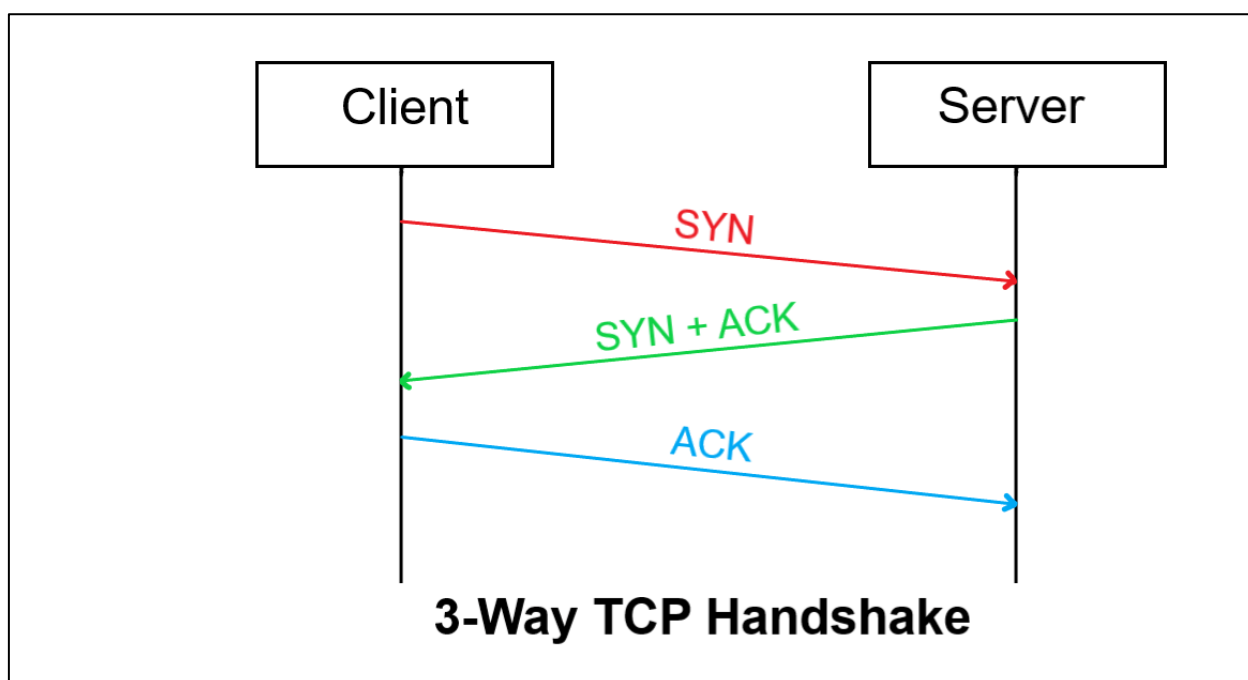


Рисунок 3.7 – Трёхстороннее рукопожатие

Обмен при трёхстороннем рукопожатии выполняется следующим образом:

1) Клиент отправляет сегмент с установленным флагом SYN. При этом сегменту присваивается произвольный порядковый номер (sequence number) в интервале от 1 до 232 (т.н. initial sequence number), относительно которого будет вестись дальнейший отсчет последовательности сегментов в соединении.

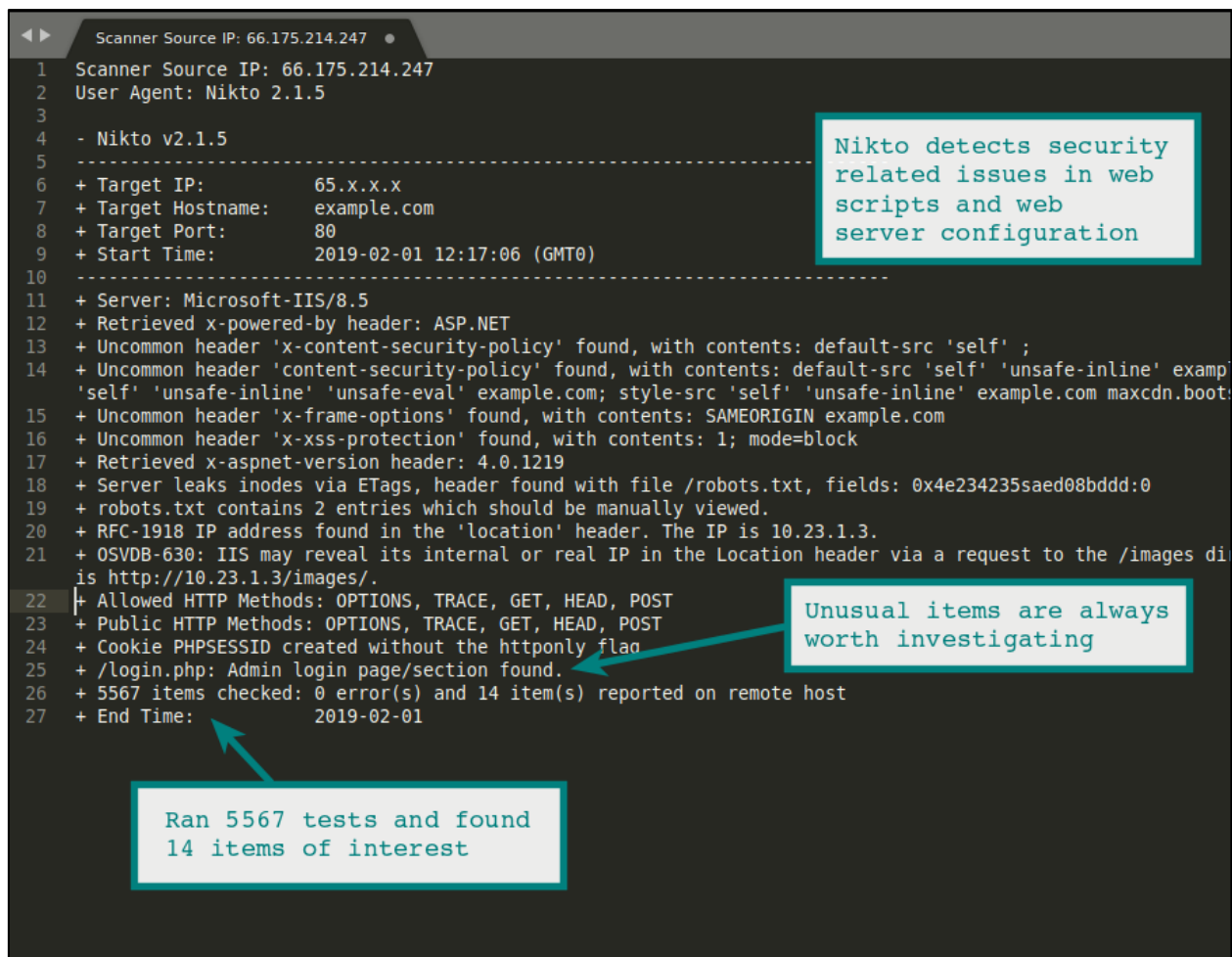
2) Сервер получает запрос и отправляет ответный сегмент с одновременно установленными флагами SYN+ACK, при этом записывает в поле «номер подтверждения» (acknowledgement number), полученный порядковый номер, увеличенный на 1 (что подтверждает получение первого сегмента), а также устанавливает свой порядковый номер, который, как и в SYN-сегменте, выбирается произвольно.

3) После получения клиентом сегмента с флагами SYN+ACK соединение считается установленным, клиент, в свою очередь, отправляет в ответ сегмент с флагом ACK с обновленными номерами последовательности, не содержащий полезной нагрузки.

В результате мы сможем с минимальным сетевым трафиком проверить доступность указанных сетевых ресурсов. После этого этапа доступные сетевые ресурсы должны быть проверены на наличие уязвимостей. Для этого мы будем использовать сканер nikto.

Nikto – это сканер с открытым исходным кодом (GPL) для поиска уязвимостей в веб-серверах. Утилита относится к классу blackbox-сканеров, т. е. сканеров, использующих стратегию сканирования методом черного ящика (cirt.net, 2024). Это значит, что заранее неизвестно о внутреннем устройстве программы/сайта (доступ к исходному коду отсутствует) и упор сделан на функциональность. Программа может обнаруживать более 6700 потенциально опасных файлов и уязвимостей. Новые уязвимости добавляются в базу данных программы по мере их возникновения. Помимо

поиска уязвимостей, сканер производит поиск на наличие устаревших версий, используемых библиотек и фреймворков (рисунок 3.8).



```
Scanner Source IP: 66.175.214.247
1 Scanner Source IP: 66.175.214.247
2 User Agent: Nikto 2.1.5
3
4 - Nikto v2.1.5
5 -----
6 + Target IP: 65.x.x.x
7 + Target Hostname: example.com
8 + Target Port: 80
9 + Start Time: 2019-02-01 12:17:06 (GMT0)
10 -----
11 + Server: Microsoft-IIS/8.5
12 + Retrieved x-powered-by header: ASP.NET
13 + Uncommon header 'x-content-security-policy' found, with contents: default-src 'self' ;
14 + Uncommon header 'content-security-policy' found, with contents: default-src 'self' 'unsafe-inline' examp
'self' 'unsafe-inline' 'unsafe-eval' example.com; style-src 'self' 'unsafe-inline' example.com maxcdn.boot
15 + Uncommon header 'x-frame-options' found, with contents: SAMEORIGIN example.com
16 + Uncommon header 'x-xss-protection' found, with contents: 1; mode=block
17 + Retrieved x-aspnet-version header: 4.0.1219
18 + Server leaks inodes via ETags, header found with file /robots.txt, fields: 0x4e234235aed08bdd:0
19 + robots.txt contains 2 entries which should be manually viewed.
20 + RFC-1918 IP address found in the 'location' header. The IP is 10.23.1.3.
21 + OSVDB-630: IIS may reveal its internal or real IP in the Location header via a request to the /images di
is http://10.23.1.3/images/.
22 + Allowed HTTP Methods: OPTIONS, TRACE, GET, HEAD, POST
23 + Public HTTP Methods: OPTIONS, TRACE, GET, HEAD, POST
24 + Cookie PHPSESSID created without the httponly flag
25 + /login.php: Admin login page/section found.
26 + 5567 items checked: 0 error(s) and 14 item(s) reported on remote host
27 + End Time: 2019-02-01
```

Nikto detects security related issues in web scripts and web server configuration

Unusual items are always worth investigating

Ran 5567 tests and found 14 items of interest

Рисунок 3.8 – Пример вывода сканера nikto

Nikto не позиционируется как стелс сканер (стелс сканеры никогда не устанавливают TCP-соединения до конца, тем самым сканирование происходит скрытно) – при сканировании сайта в логах сайта или в любой другой системе обнаружения вторжений, если она используется, будет отображена информация о том, что сайт подвергается сканированию.

На этом этапе мы уже можем сказать, какие есть уязвимости на открытых сетевых ресурсах, но в целях оптимизации сбора поверхности атаки и минимизации участия оператора для генерации итогового отчёта, мы подключим к нашей программе один из популярных GPT-чатов.

GPT (Generative Pre-trained Transformer) – это алгоритм обработки естественного языка, выпущенный американской компанией OpenAI. Главная особенность нейросети заключается в ее способности запоминать и

анализировать информацию, создавая на ее основе связный и логичный текст. Мощная языковая модель имеет архитектуру типа «трансформер», которая позволяет ей находить связи между отдельными словами и просчитывать наиболее релевантную последовательность слов и предложений.

Для работы с GPT-чатом мы будем использовать популярную библиотеку g4f.

g4f (GPT4Free) – библиотека для Python с бесплатным доступом к ChatGPT. Данная библиотека имеет свои ограничения на доступ к GPT-чатам, так как проект позиционируется «для учебных целей» (рисунок 3.9).

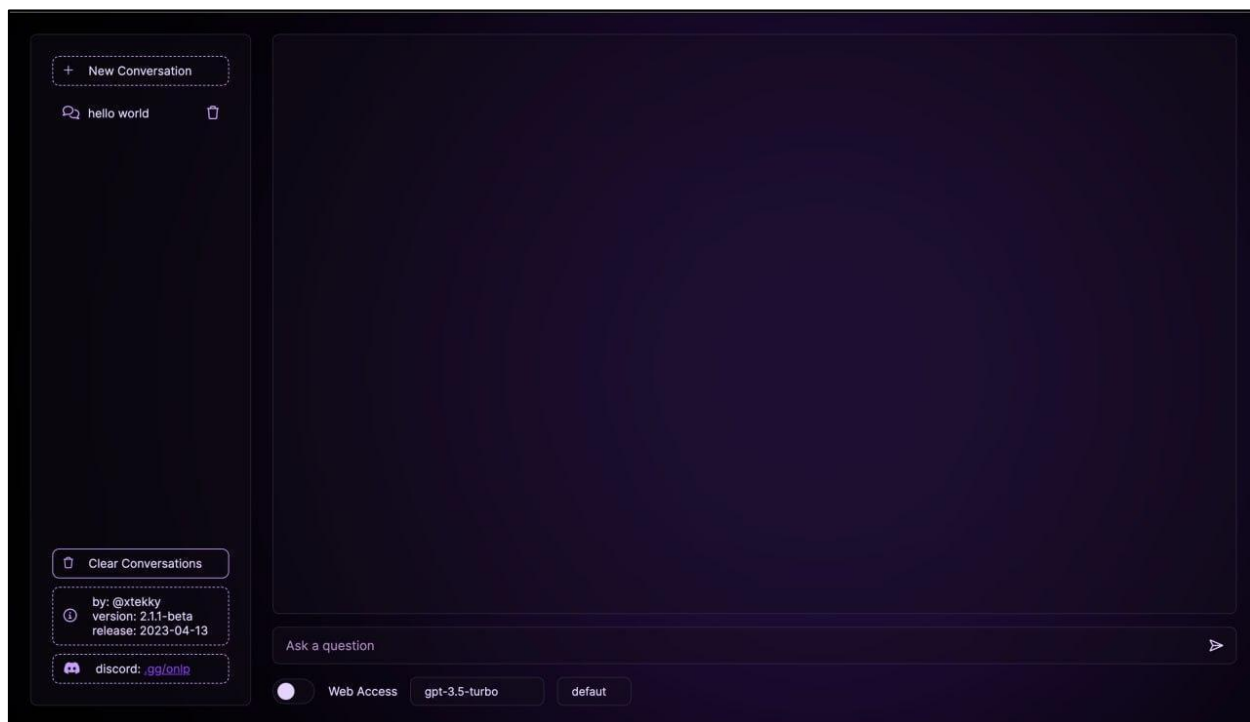


Рисунок 3.9 – Стартовое окно GUI docker-версии проекта GPT4Free

Мы не сможем передавать большие объёмы запросов или историю сообщений, но для генерации итогового отчёта по результатам сбора поверхности атаки данных возможностей достаточно. В результате мы будем получать человекочитаемый отчёт, который сразу можно использовать для дальнейших действия, либо по устранению найденных уязвимостей, либо по проникновению во внешний периметр.

2.3 РАБОТА ПРОГРАММЫ

Вначале реализуем первый этап работы программы, а именно проверку доступности сетевых ресурсов при помощи библиотеки **scapy**.

На вход программы на Python мы будем подавать файл **scan_ports.csv**, содержащий перечень сетевых ресурсов для сканирования в следующем формате:

192.168.100.135;80,443,8080

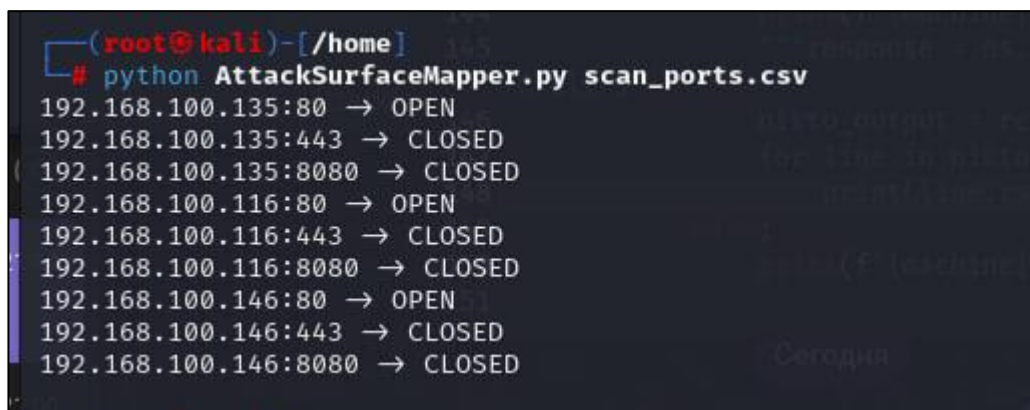
192.168.100.116;80,443,8080

192.168.100.146;80,443,8080

, где вначале строки идёт IP-адрес сетевого ресурса, а после точки с запятой список портов для сканирования.

Запустив следующую команду, получаем её вывод (рисунок 2.10):

python AttackSurfaceMapper.py scan_ports.csv



```
(root@kali)-[/home]
# python AttackSurfaceMapper.py scan_ports.csv
192.168.100.135:80 → OPEN
192.168.100.135:443 → CLOSED
192.168.100.135:8080 → CLOSED
192.168.100.116:80 → OPEN
192.168.100.116:443 → CLOSED
192.168.100.116:8080 → CLOSED
192.168.100.146:80 → OPEN
192.168.100.146:443 → CLOSED
192.168.100.146:8080 → CLOSED
```

Рисунок 2.10 – Вывод проверки доступности портов с помощью scapy

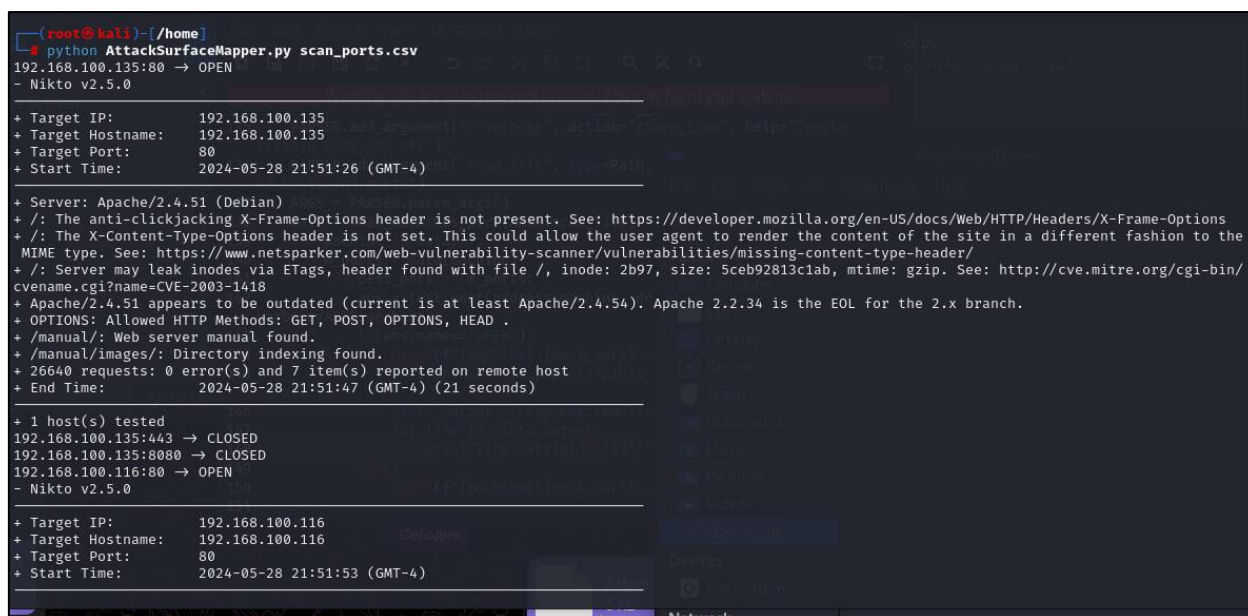
CSV (Comma-Separated Values) — это текстовый формат для представления табличных данных. Строка таблицы соответствует строке текста, которая содержит поля, разделенные запятыми. Тип файлов предназначен для передачи объемных текстовых данных между различными программами и сервисами.

В последнее время разделителем может быть не только запятая, но и другие символы (пробел, точка с запятой, табуляция, другое).

Данная особенность позволит использовать нашу программу не только как отдельный инструмент, но и встраивать в конвейер разработки. То есть входной файл **scan_ports.csv** может быть выходным результатом другой программы.

На втором этапе разработки программы нам необходимо для доступных сетевых ресурсов (статус «OPEN») проверить наличие известных уязвимостей с помощью сканера **nikto**.

Для этого мы дополним наш код, вызовом сканера **nikto**, и посмотрим на получившийся вывод программы (рисунок 2.11).



```
(root@kali) ~ [~/home]
python AttackSurfaceMapper.py scan_ports.csv
192.168.100.135:80 → OPEN
- Nikto v2.5.0

+ Target IP: 192.168.100.135
+ Target Hostname: 192.168.100.135
+ Target Port: 80
+ Start Time: 2024-05-28 21:51:26 (GMT-4)

+ Server: Apache/2.4.51 (Debian)
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the
MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ /: Server may leak inodes via ETags, header found with file /, inode: 2b97, size: 5ceb92813c1ab, mtime: gzip. See: http://cve.mitre.org/cgi-bin/
cvename.cgi?name=CVE-2003-1418
+ Apache/2.4.51 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ OPTIONS: Allowed HTTP Methods: GET, POST, OPTIONS, HEAD .
+ /manual/: Web server manual found.
+ /manual/images/: Directory indexing found.
+ 26640 requests: 0 error(s) and 7 item(s) reported on remote host
+ End Time: 2024-05-28 21:51:47 (GMT-4) (21 seconds)

+ 1 host(s) tested
192.168.100.135:443 → CLOSED
192.168.100.135:8080 → CLOSED
192.168.100.116:80 → OPEN
- Nikto v2.5.0

+ Target IP: 192.168.100.116
+ Target Hostname: 192.168.100.116
+ Target Port: 80
+ Start Time: 2024-05-28 21:51:53 (GMT-4)
```

Рисунок 2.11 – Вывод с запуском сканера **nikto** для доступных портов

Теперь нам доступна информация о найденных уязвимостях (CVE) для сетевых ресурсов.

CVE (Common Vulnerabilities and Exposures) — база данных общеизвестных уязвимостей информационной безопасности. Каждой уязвимости присваивается идентификационный номер вида CVE-год-номер, описание и ряд общедоступных ссылок с описанием. Проект финансируется американской группой быстрого реагирования на инциденты (US CERT). Непосредственная поддержка возложена на некоммерческую организацию MITRE Corp.

Уже на данном этапе мы имеем достаточную информацию о поверхности атаки периметра организации.

На последнем этапе в целях оптимизации сбора поверхности атаки и для генерации итогового отчёта о найденных уязвимостях, мы подключим GPT-чат с помощью библиотеки **g4f**.

Модернизируем код нашей программы, передав вывод сканера nikto чату GPT (Рисунок 2.12).

```

**Цель сканирования:**
Анализ безопасности сетевых узлов с IP-адресами 192.168.100.135, 192.168.100.116 и 192.168.100.146 на порту 80.

--

**Результаты сканирования:**

**1. Узел 192.168.100.135:**
- **Сервер:** Apache/2.4.51 (Debian)
- **Выявленные уязвимости:**
  - Отсутствует заголовок X-Frame-Options, что может привести к атакам clickjacking.
  - Отсутствует заголовок X-Content-Type-Options, что может привести к неправильному отображению контента.
  - Возможная утечка инодов через ETags (CVE-2003-1418).
- **Оценка уязвимости (CVSS):**
  - CVE-2003-1418: 5.0 (Medium)

**2. Узел 192.168.100.116:**
- **Сервер:** Apache/2.2.8 (Ubuntu) DAV/2
- **Выявленные уязвимости:**
  - Отсутствует заголовок X-Frame-Options, что может привести к атакам clickjacking.
  - Отсутствует заголовок X-Content-Type-Options, что может привести к неправильному отображению контента.
  - Включён Apache mod_negotiation с MultiViews, что может способствовать перебору имен файлов.
  - Активен HTTP TRACE method, что указывает на уязвимость к XST.
  - Различные директории и файлы, такие как phpMyAdmin и phpinfo.php, могут предоставить чувствительную информацию.
- **Оценка уязвимости (CVSS):**
  - CVE-2003-1418: 5.0 (Medium)

**3. Узел 192.168.100.146:**
- **Сервер:** Apache/2.4.51 (Debian)
- **Выявленные уязвимости:**
  - Отсутствует заголовок X-Frame-Options, что может привести к атакам clickjacking.
  - Отсутствует заголовок X-Content-Type-Options, что может привести к неправильному отображению контента.
  - Различные директории и файлы, такие как wp-content/uploads, могут раскрывать чувствительную информацию.
- **Оценка уязвимости (CVSS):**
  - Нет выявленных CVE-уязвимостей.

--

**Общий вывод:**
Узлы имеют несколько общих уязвимостей, таких как отсутствие заголовков безопасности и доступность директорий с чувствительными данными. Рекомендуется принять меры для обновления серверного ПО и реализации рекомендаций по безопасности HTTP-заголовков для устранения обнаруженных уязвимостей

```

Рисунок 2.12 – Вывод с использованием GPT-чата

В результате мы получаем готовый отчёт о найденных уязвимостях, где удобно сгруппированы сетевые ресурсы, а также даны рекомендации по устранению уязвимостей.

В дополнении мы попросили GPT-чат выполнить расчёт критичности найденных уязвимостей (CVSS) (рисунок 2.13) и, если возможно, определить CMS (рисунок 2.14).

CVSS (Common Vulnerability Scoring System) — открытый стандарт для оценки степени опасности уязвимостей.

CVSS разработал Национальный совет по инфраструктуре (National Infrastructure Advisory Council, NIAC) США. Также в создании и обновлении стандарта участвовали коммерческие компании, такие как Microsoft, Cisco и другие. Поддержкой системы занимается Форум групп безопасности и реагирования на инциденты (Forum of Incident Response and Security Teams, FIRST).

В настоящее время для оценки уязвимостей используется версия CVSS 3.1, вышедшая в июне 2019 года.

```
### Расчёт CVSS

#### 192.168.100.135
- CVE-2003-1418: [CVSS 3.0 Score](https://www.first.org/cvss/calculator/3.0): 5.3 (Medium)

#### 192.168.100.116
- CVE-2003-1418: [CVSS 3.0 Score](https://www.first.org/cvss/calculator/3.0): 5.3 (Medium)
- OSVDB-12184: [CVSS 3.0 Score](https://www.first.org/cvss/calculator/3.0): 7.5 (High)

#### 192.168.100.146
- CWE-530: [CVSS 3.0 Score](https://www.first.org/cvss/calculator/3.0): 5.3 (Medium)
- CWE-552: [CVSS 3.0 Score](https://www.first.org/cvss/calculator/3.0): 5.3 (Medium)
- RSnake's RFI: [CVSS 3.0 Score](https://www.first.org/cvss/calculator/3.0): 7.5 (High)
```

Рисунок 2.13 – Расчёт критичности найденных уязвимостей GPT-чатом

CMS (Content Management System) — это система создания и управления сайтом. Это визуально удобный интерфейс, с помощью которого можно добавлять и редактировать содержимое сайта.


```
#### Сканирование цели: 192.168.100.146
- **Сервер:** Apache/2.4.51 (Debian)
- **Порт:** 80
- **Начало сканирования:** 17:44:08 (GMT-4)
- **Завершение сканирования:** 17:44:24 (GMT-4)

**Найденные уязвимости и проблемы конфигурации:**
1. Отсутствует заголовок X-Frame-Options (анти-кликджекинг).
2. Отсутствует заголовок X-Content-Type-Options.
3. Найден заголовок Drupal Link.
4. Необычный заголовок 'x-redirect-by' со значением 'WordPress'.
5. Индексация каталогов `/wp-content/plugins/`, `/wp-content/`, `/wp-content/uploads/`.
6. Найдены файлы `backup.zip` (CWE-530) и `db.sql`.
7. Найден скрипт `info.php` (CWE-552).
8. RFI через файл `info.php` (RSnake's RFI list).
9. Найден файл `wp-links-opml.php` (раскрывает версию WordPress).
10. Найден файл `license.txt`.
11. Найден файл `wp-login.php?action=register` (cookie без httponly флага).

**CMS:** WordPress и Drupal.
```

Рисунок 2.14 – Определение системы управления контентом GPT-чатом

Например, в данном случае, GPT-чат правильно определил наличие CMS WordPress на нашей самостоятельно подготовленной машине.

На данном этапе можем считать, что сбор поверхности атаки внешнего периметра завершён.

3 РИСК-МЕНЕДЖМЕНТ

3.1 НОРМАТИВНО-ПРАВОВАЯ БАЗА ПО ЗАЩИТЕ КОНФИДЕНЦИАЛЬНЫХ ДАННЫХ

Защита персональных данных в Российской Федерации регулируется комплексом нормативно-правовых актов, направленных на обеспечение прав граждан на конфиденциальность и безопасность их личной информации. Основными документами в этой области являются федеральные законы, подзаконные акты и ведомственные инструкции. В этом разделе рассмотрим ключевые нормативно-правовые документы, регулирующие защиту персональных данных и конфиденциальных данных в РФ.

1. Федеральный закон № 152-ФЗ "О персональных данных"

Федеральный закон № 152-ФЗ "О персональных данных" от 27 июля 2006 года (с последними изменениями от 24 февраля 2023 года) является основополагающим документом в сфере защиты персональных данных. Он определяет основные понятия и принципы обработки персональных данных, права субъектов персональных данных и обязанности операторов.

Ключевые моменты:

- **Понятия и принципы:** Закон определяет персональные данные как любую информацию, относящуюся к определенному или определяемому физическому лицу. Устанавливаются принципы обработки данных: законность, справедливость, прозрачность, ограничение обработки достижением конкретных целей.
- **Права субъектов:** Субъекты персональных данных имеют право на доступ к своим данным, их исправление, блокирование или уничтожение в случае их недостоверности или неправомерной обработки.
- **Обязанности операторов:** Операторы обязаны обеспечивать конфиденциальность и безопасность персональных данных, принимать меры по защите данных от несанкционированного доступа, утечки и других угроз.

- **Передача данных:** Устанавливаются правила передачи персональных данных как внутри страны, так и за ее пределы, включая обязательное уведомление субъекта данных и получение его согласия.

2. Постановление Правительства РФ № 1239

Постановление Правительства РФ от 1 ноября 2012 года № 1239 "Об утверждении требований к защите персональных данных при их обработке в информационных системах персональных данных" с изменениями от 16 марта 2023 года устанавливает требования к мерам защиты персональных данных при их обработке в информационных системах.

Ключевые моменты:

- **Классификация систем:** Постановление вводит классификацию информационных систем по уровням защищенности, зависящую от объема и чувствительности обрабатываемых данных.

- **Криптографическая защита:** Требуется использование средств криптографической защиты информации, сертифицированных в установленном порядке.

- **Оценка эффективности:** Операторы обязаны проводить регулярную оценку эффективности принимаемых мер по защите данных.

- **Хранение и уничтожение:** Определяются условия хранения и уничтожения персональных данных после окончания их обработки.

3. Приказ ФСТЭК России № 239

Приказ ФСТЭК России от 18 февраля 2013 года № 239 "Об утверждении состава и содержания организационных и технических мер по обеспечению безопасности персональных данных при их обработке в информационных системах персональных данных" с изменениями от 18 августа 2022 года детализирует меры по защите данных.

Ключевые моменты:

- **Организация доступа:** Введение режима допуска к персональным данным, ограничивающего доступ только для уполномоченных лиц.

- **Сертификация средств защиты:** Использование сертифицированных средств защиты информации.

- **Проверки и аудит:** Регулярные проверки и аудиты системы защиты информации для выявления и устранения уязвимостей.

- **Обучение персонала:** Обучение и инструктаж сотрудников, работающих с персональными данными, по вопросам информационной безопасности.

4. Федеральный закон № 149-ФЗ "Об информации, информационных технологиях и о защите информации"

Федеральный закон № 149-ФЗ "Об информации, информационных технологиях и о защите информации" от 27 июля 2006 года (с изменениями от 1 июля 2023 года) регулирует общие принципы использования информационных технологий и защиты информации.

Ключевые моменты:

- **Доступ к информации:** Определение порядка доступа к информации, в том числе персональным данным.

- **Обязанности владельцев:** Установление обязанностей владельцев информационных систем по защите информации.

- **Ответственность:** Определение ответственности за нарушение законодательства в области защиты информации, включая штрафные санкции.

5. Постановление Правительства РФ № 687

Постановление Правительства РФ от 15 сентября 2022 года № 687 "О мерах по защите персональных данных при их обработке в государственных информационных системах" устанавливает специальные требования к защите персональных данных в государственных информационных системах.

Ключевые моменты:

- **Специфические требования:** Введение дополнительных требований для государственных информационных систем, включая более строгие меры по защите данных.

- **Мониторинг и контроль:** Усиленные меры по мониторингу и контролю за безопасностью персональных данных.

- **Совместимость систем:** Обеспечение совместимости мер защиты с другими государственными информационными системами.

6. Приказ Минцифры России № 232

Приказ Минцифры России от 14 ноября 2022 года № 232 "Об утверждении требований к обеспечению безопасности персональных данных при их обработке в информационных системах персональных данных" определяет требования к обеспечению безопасности при обработке данных в информационных системах.

Ключевые моменты:

- **Системы защиты:** Определение требований к системам защиты информации.

- **Управление доступом:** Введение правил управления доступом к информационным системам.

- **Оценка рисков:** Проведение регулярной оценки рисков и угроз информационной безопасности.

7. Приказ Роскомнадзора № 119

Приказ Роскомнадзора от 17 декабря 2022 года № 119 "Об утверждении требований к обработке и защите персональных данных оператором" устанавливает требования к обработке и защите персональных данных операторами.

Ключевые моменты:

- **Организация обработки:** Определение правил организации обработки персональных данных операторами.

- **Обеспечение безопасности:** Установление мер по обеспечению безопасности при обработке данных.

- **Контроль и отчетность:** Введение механизмов контроля и отчетности по соблюдению требований защиты данных.

8. ГОСТ Р 57580.1-2022

- ГОСТ Р 57580.4-2022 "Защита информации. Безопасность финансовых (банковских) операций. Основные положения и требования" устанавливает требования к защите информации в финансовых и банковских операциях.

Ключевые моменты:

- **Технические меры:** Требования к техническим мерам защиты информации.
- **Организационные меры:** Определение организационных мер по обеспечению безопасности.
- **Соответствие требованиям:** Обязательное соответствие требованиям стандарта для финансовых учреждений.

9. Международные нормы и соглашения

Российское законодательство также учитывает международные нормы и стандарты, такие как Общий регламент по защите данных (GDPR) Европейского Союза и рекомендации Международной организации по стандартизации (ISO) в области защиты информации.

Ключевые моменты:

- **Согласование норм:** Адаптация российских норм к международным стандартам.
- **Международное сотрудничество:** Участие России в международных соглашениях и проектах по защите информации.
- **Взаимодействие с иностранными партнерами:** Обеспечение защиты данных при их передаче за границу.

10. Федеральный закон № 242-ФЗ "О внесении изменений в отдельные законодательные акты Российской Федерации в части установления порядка хранения персональных данных"

Федеральный закон № 242-ФЗ от 21 июля 2014 года (с изменениями от 5 декабря 2022 года) устанавливает порядок хранения персональных данных граждан Российской Федерации на территории России.

Ключевые моменты:

- **Локализация данных:** Персональные данные граждан России должны храниться и обрабатываться на территории Российской Федерации.
- **Обязанности операторов:** Операторы обязаны обеспечить локализацию баз данных с персональными данными на территории РФ.
- **Ответственность:** Установление ответственности за нарушение требований по локализации данных, включая штрафные санкции и блокировку информационных ресурсов.

11. Приказ ФСТЭК России № 55

Приказ ФСТЭК России от 11 февраля 2013 года № 55 "Об утверждении требований к защите конфиденциальной информации" (с изменениями от 10 января 2023 года) определяет меры по защите конфиденциальной информации, включая коммерческую тайну и другие виды конфиденциальных данных.

Ключевые моменты:

- **Классификация информации:** Определение категорий конфиденциальной информации и требований к их защите.
- **Организационные меры:** Введение организационных мер по защите конфиденциальной информации.
- **Технические меры:** Использование сертифицированных средств защиты информации.
- **Контроль доступа:** Управление доступом к конфиденциальной информации на основе принципа "необходимости знать".

12. Федеральный закон № 98-ФЗ "О коммерческой тайне"

Федеральный закон № 98-ФЗ "О коммерческой тайне" от 29 июля 2004 года (с изменениями от 25 декабря 2023 года) регулирует отношения, связанные с установлением, использованием и защитой коммерческой тайны.

Ключевые моменты:

- **Определение коммерческой тайны:** Коммерческая тайна включает сведения, имеющие реальную или потенциальную коммерческую ценность в силу их неизвестности третьим лицам.

- **Режим коммерческой тайны:** Установление режима коммерческой тайны, включая меры по защите информации.
- **Права и обязанности:** Определение прав и обязанностей обладателей коммерческой тайны и их контрагентов.
- **Ответственность:** Установление ответственности за нарушение режима коммерческой тайны, включая административные и уголовные санкции.

13. Постановление Правительства РФ № 1119

Постановление Правительства РФ от 1 ноября 2012 года № 1119 "Об утверждении требований к защите информации в информационных системах персональных данных" с изменениями от 15 марта 2023 года устанавливает требования к защите информации в информационных системах, обрабатывающих персональные данные.

Ключевые моменты:

- **Классификация систем:** Определение уровней защищенности информационных систем в зависимости от объема и характера обрабатываемых данных.
- **Технические меры:** Введение обязательных технических мер защиты информации.
- **Оценка защищенности:** Периодическая оценка уровня защищенности информационных систем.
- **Контроль доступа:** Управление доступом к информации и её защите на всех этапах обработки.

Нормативно-правовая база по защите персональных данных и конфиденциальной информации в Российской Федерации представляет собой сложную и многослойную систему, включающую различные законы, постановления и приказы, направленные на обеспечение конфиденциальности и безопасности личной информации граждан и юридических лиц. Соблюдение этих нормативных актов является обязательным для всех операторов персональных данных и

конфиденциальной информации, что позволяет эффективно защищать права субъектов данных и предотвращать нарушения в области информационной безопасности.

3.2 РИСКИ В СВЯЗИ С УТЕЧКОЙ КОНФИДЕНЦИАЛЬНЫХ ДАННЫХ

Утечка конфиденциальной информации и персональных данных является серьезной угрозой для безопасности, как для организаций, так и для частных лиц. Рассмотрим основные риски, связанные с такими утечками, и приведем примеры для лучшего понимания.

1. Финансовые потери

Пример 1: Утечка данных банковских карт

Если киберпреступники получают доступ к данным банковских карт, они могут использовать эти данные для несанкционированных транзакций, что приводит к финансовым потерям для владельцев карт и банков. Например, в 2013 году утечка данных в компании Target привела к краже данных миллионов кредитных и дебетовых карт, что нанесло значительный ущерб клиентам и самой компании.

Пример 2: Атака на банковские счета

Фишинговые атаки или взлом учетных записей могут привести к тому, что злоумышленники получают доступ к банковским счетам пользователей и выведут с них средства. Это не только наносит ущерб пользователям, но и подрывает доверие к банковским учреждениям.

2. Репутационные потери

Пример 1: Утечка данных о клиентах

Компании, работающие с большим количеством персональных данных, такие как медицинские учреждения или социальные сети, могут потерять доверие клиентов в случае утечки. Например, утечка данных из Facebook в 2018 году, когда данные 87 миллионов пользователей были неправомерно переданы Cambridge Analytica, серьезно подорвала репутацию компании.

Пример 2: Конфиденциальная информация компаний

Утечка внутренней конфиденциальной информации, такой как стратегии развития, планы на будущее или финансовые отчеты, может привести к утрате конкурентных преимуществ и вызвать негативную реакцию со стороны инвесторов и партнеров.

3. Правовые и регуляторные последствия

Пример 1: Несоответствие требованиям GDPR

Компании, работающие с данными граждан Европейского Союза, должны соблюдать Общий регламент по защите данных (GDPR). В случае утечки данных и несоблюдения требований регламента, компании могут быть оштрафованы на большие суммы. В 2019 году British Airways была оштрафована на 183 миллиона фунтов стерлингов за утечку данных, что стало одним из крупнейших штрафов по GDPR.

Пример 2: Законодательство о защите данных в разных странах

Помимо GDPR, в разных странах существуют свои законы о защите данных, несоблюдение которых может привести к правовым последствиям. В США, например, Законы о защите конфиденциальности медицинской информации (HIPAA) строго регулируют обращение с медицинскими данными. Нарушение этих законов может привести к крупным штрафам и юридическим санкциям.

4. Нарушение приватности и личной безопасности

Пример 1: Кража личности

Утечка персональных данных, таких как номера социальных страхований, данные паспортов или водительских удостоверений, может привести к краже личности. Злоумышленники могут использовать эти данные для оформления кредитов, совершения покупок или даже для совершения преступлений под чужим именем. Это может вызвать длительные проблемы для жертвы, требующие времени и усилий для восстановления своей репутации и финансового состояния.

Пример 2: Доксинг

Публикация личной информации в интернете без согласия жертвы (доксинг) может привести к преследованиям, угрозам и даже физической опасности. В 2019 году журналисты, расследующие деятельность экстремистских групп, сталкивались с доксингом, когда их личные данные были опубликованы в интернете с целью запугивания.

5. Нарушение коммерческих тайн

Пример 1: Кража интеллектуальной собственности

Утечка данных, содержащих патенты, научные исследования, торговые секреты или технологические разработки, может нанести значительный ущерб компаниям. Примером может служить кража данных о разработке новых продуктов в технологической компании, что позволяет конкурентам использовать эти данные для создания аналогичных продуктов, снижая конкурентное преимущество первоначального разработчика.

Пример 2: Утечка данных о сделках и переговорах

Если данные о планируемых сделках или переговорах становятся известны конкурентам или общественности, это может повлиять на результаты переговоров или стоимость акций компании. В 2014 году компания Sony Pictures стала жертвой кибератаки, в результате которой были опубликованы данные о внутренних переговорах и предстоящих фильмах, что нанесло серьезный ущерб компании.

6. Социальные и психологические риски

Пример 1: Кибербуллинг и психологическое давление

Утечка персональных данных может стать причиной кибербуллинга, особенно среди молодежи. Например, утечка личной переписки или фотографий может использоваться для шантажа или публичного унижения. Это может привести к серьезным психологическим проблемам, включая депрессию и тревожность.

Пример 2: Нарушение личной жизни

Утечка данных, содержащих информацию о личной жизни, такой как переписка, фото и видео, может привести к серьезным нарушениям приватности. В 2014 году хакеры взломали учетные записи iCloud и опубликовали личные фотографии знаменитостей, что стало известным как "The Farprening". Это событие вызвало широкий общественный резонанс и поставило вопросы о безопасности облачных сервисов.

7. Угроза национальной безопасности

Пример 1: Шпионская деятельность

Утечка данных, содержащих информацию о государственных структурах или военно-промышленных комплексах, может использоваться враждебными государствами для шпионской деятельности. Примером может служить взлом базы данных Управления кадровых служб США (OPM) в 2015 году, когда были украдены данные о миллионах государственных служащих, включая данные о безопасности и проверках на допуск.

Пример 2: Критическая инфраструктура

Утечка данных о критической инфраструктуре, такой как энергетические сети, транспортные системы или водоснабжение, может поставить под угрозу национальную безопасность. Злоумышленники могут использовать эти данные для проведения атак на инфраструктуру, что приведет к серьезным последствиям для населения и экономики страны.

Заключение

Утечка конфиденциальной информации и персональных данных несет в себе множество рисков, затрагивающих финансовые, репутационные, правовые, социальные и национальные аспекты. Компании и частные лица должны принимать меры для защиты своих данных, включая использование современных технологий безопасности, обучение сотрудников и следование законодательным требованиям. Только комплексный подход к безопасности данных поможет минимизировать риски и предотвратить негативные последствия утечек.

3.3 МИТИГАЦИЯ РИСКОВ

Для эффективного управления рисками утечек данных необходимо применять комплексный подход, включающий технические, административные и организационные меры. Рассмотрим основные стратегии митигации рисков с примерами и подробным описанием.

1. Технические меры

1.1. Шифрование данных

Шифрование данных позволяет защитить информацию в случае ее утечки. Даже если злоумышленники получают доступ к зашифрованным данным, без ключа расшифровки они не смогут использовать эту информацию.

Пример: Компания применяет шифрование для всех данных, передаваемых через интернет и хранящихся на серверах, что обеспечивает защиту, как в процессе передачи, так и при хранении.

1.2. Аутентификация и авторизация

Использование многофакторной аутентификации (MFA) и ролевой модели доступа ограничивает доступ к данным только уполномоченным пользователям.

Пример: Банки внедряют многофакторную аутентификацию для доступа к онлайн-банкингу, требуя от пользователей не только пароль, но и одноразовый код, отправленный на мобильное устройство.

1.3. Мониторинг и анализ

Регулярный мониторинг систем и анализ логов помогает выявлять подозрительную активность и реагировать на нее своевременно.

Пример: Система мониторинга обнаруживает необычное количество запросов к базе данных в нерабочее время и отправляет уведомление команде безопасности для проверки.

1.4. Обновления и патчи

Регулярное обновление программного обеспечения и применение патчей устраняют уязвимости, которые могут быть использованы злоумышленниками для доступа к данным.

Пример: Организация внедряет автоматизированную систему обновлений для всех рабочих станций и серверов, обеспечивая своевременное применение критических патчей.

2. Административные меры

2.1. Политики безопасности

Разработка и внедрение четких политик безопасности данных, определяющих правила обработки, хранения и передачи информации.

Пример: Компания разрабатывает политику безопасности, которая включает правила использования персональных устройств для работы, требования к паролям и правила доступа к конфиденциальной информации.

2.2. Обучение сотрудников

Регулярное обучение сотрудников принципам информационной безопасности и действиям в случае обнаружения инцидентов.

Пример: Проводятся регулярные тренинги и семинары для сотрудников, обучающие их распознаванию фишинговых писем и правильному реагированию на инциденты безопасности.

2.3. Управление рисками

Проактивное управление рисками включает в себя идентификацию, оценку и разработку мер по минимизации рисков утечек данных.

Пример: Компания проводит регулярные аудиты безопасности и оценки рисков, на основании которых разрабатывает и внедряет меры по улучшению защиты данных.

3. Организационные меры

3.1. Планы реагирования на инциденты

Разработка планов реагирования на инциденты позволяет быстро и эффективно действовать в случае утечки данных, минимизируя последствия.

Пример: Организация разрабатывает план реагирования, включающий создание команды быстрого реагирования, процедуры уведомления пострадавших и взаимодействие с регуляторами.

3.2. Резервное копирование данных

Регулярное создание резервных копий данных и их безопасное хранение позволяет восстановить информацию в случае утраты или повреждения.

Пример: Компания проводит ежедневное резервное копирование данных и хранит копии в географически распределенных дата-центрах, обеспечивая восстановление в случае катастрофы.

3.3. Сегментация сети

Разделение корпоративной сети на сегменты ограничивает распространение угроз и минимизирует возможность утечек данных.

Пример: Внутренняя сеть компании разделена на сегменты с разными уровнями доступа, что ограничивает возможность перемещения злоумышленников по всей сети в случае взлома одной из частей.

4. Юридические меры

4.1. Соблюдение нормативных требований

Соблюдение законодательных и нормативных требований по защите данных, таких как GDPR, HIPAA и другие, снижает риск правовых последствий.

Пример: Компания внедряет процессы и процедуры, соответствующие требованиям GDPR, включая получение согласия на обработку данных и предоставление пользователям возможности управлять своими данными.

4.2. Контракты с поставщиками и партнерами

Заключение контрактов с поставщиками и партнерами, включающих требования по защите данных и ответственности за их нарушение.

Пример: В договоры с подрядчиками включаются пункты о соблюдении стандартов безопасности и ответственности за утечки данных,

что мотивирует сторонние организации также соблюдать высокие стандарты защиты.

5. Физические меры

5.1. Контроль доступа к помещениям

Ограничение физического доступа к помещениям, где хранятся или обрабатываются конфиденциальные данные.

Пример: В офисе компании установлены системы контроля доступа, видеонаблюдения и охранные системы, что позволяет ограничить доступ к серверным комнатам и другим критически важным зонам.

5.2. Безопасное уничтожение данных

Обеспечение безопасного уничтожения бумажных документов и электронных носителей, содержащих конфиденциальную информацию.

Пример: Организация использует shreddеры для уничтожения бумажных документов и сертифицированные методы удаления данных с жестких дисков перед их утилизацией.

Вывод

Митигация рисков, связанных с утечкой конфиденциальной информации и персональных данных, требует комплексного подхода, включающего технические, административные, организационные, юридические и физические меры. Только всесторонний подход к защите данных может обеспечить надежную безопасность и минимизировать последствия потенциальных утечек. Внедрение таких мер позволит организациям защищать свои данные, репутацию и доверие клиентов, а также соблюдать законодательные и нормативные требования.

ЗАКЛЮЧЕНИЕ

Мы видим, насколько широко поле для решения проблем и задач тестирования внешнего периметра атак организации. Насколько разнообразны программы, которые решают эти проблемы и задачи. В текущий момент происходит развитие и наработка разных практик и готовых продуктов для решения конкретных задач, вставших перед бизнесом и государственными структурами. Наша программа имеет задачу заполнить нишу простых и понятных решений для фирм массового сегмента, которые наименее защищены от угроз в плане ИБ, учитывая то, что решения по вопросам, касающимся ИБ в фирмах, согласно исследованию Gartner, принимаются с учетом специалистов по ИБ только от 12 до 30% случаев. Безопасность – это не риски, а дополнительный контроль. Этот базовый контроль представляет собой автоматическую оценку уязвимостей, для принятия специалистами решений по оперативному исправлению ситуации. В нашем ВКР реализована именно такая схема. Подобный базовый вариант возможно внедрить в организации с низким уровнем квалификации специалистов и успешно разворачивать необходимые сервисы, не опасаясь за внезапную остановку деятельности, финансовых, репутационных и прочих потерь. Сумбур в ИБ планировании, недостаточном понимании узких мест или отсутствия DLP систем приводят к крупным финансовым потерям и остановке работы фирм, что мы регулярно наблюдаем неоднократно за прошедший месяц. Наше простое решение – один из базовых кирпичиков, который стоит принять на вооружение, а его аналоги внедрять во всех организациях, у которых нет более серьезных систем защиты и анализа. Мы надеемся, что наше программное решение в рамках ВКР будет реализовано в продакшене. Разнообразие подобных базовых решений позволит сделать массовый шаг вперед в информационной безопасности в России и станет простым, массовым и классическим, как антивирус на каждом устройстве.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

Нормативно-правовые акты

1 Акт министерств и ведомств "Об утверждении Требований по обеспечению безопасности значимых объектов критической информационной инфраструктуры Российской Федерации" от 25.12.2017 № 239 // Официальный интернет-портал правовой информации. - 2018 г. - № 0001201803270041. - Ст. 50524 с изм. и допол. в ред. от 26.03.2018.

2 Акт правительства Российской Федерации "Об утверждении требований к защите персональных данных при их обработке в информационных системах персональных данных" от 14.11.2012 № 1239 с изм. и допол. в ред. от 16.03.2023.

3 Акт правительства Российской Федерации "Постановление Об утверждении Положения об особенностях обработки персональных данных, осуществляемой без использования средств автоматизации" от 15.09.2008 № 689 // Собрание законодательства Российской Федерации. - 2008 г. - № 38. - Ст. 4320

4 Акт правительства Российской Федерации "Постановление Об утверждении требований к защите персональных данных при их обработке в информационных системах персональных данных" от 01.11.2012 № 1119 // Российская газета. - 2012 г. - № 11

5 ГОСТ Р 57580.4-2022 "Защита информации. Безопасность финансовых (банковских) операций. Основные положения и требования" от 01.02.2023 // Официальный интернет-портал правовой информации

6 Закон Российской Федерации "О внесении изменений в отдельные законодательные акты Российской Федерации в части установления порядка хранения персональных данных" от 21.07.2014 № 242-ФЗ // Официальный интернет-портал правовой информации

7 Закон Российской Федерации "О коммерческой тайне" от 29.07.2004 № 98-ФЗ // Российская газета. - 2004 г. - № 8. - с изм. и допол. в ред. от 10.01.2023.

8 Закон Российской Федерации "Об информации, информационных технологиях и о защите информации" от 27.07.2006 № 239 // Собрание законодательства Российской Федерации. - 2006 г. - № 31. - Ст. 3448 (Часть I)

9 Закон Российской Федерации "Федеральный закон № 152-ФЗ "О персональных данных"" от 27.07.2006 № 152-ФЗ // Парламентская газета. - 2006 г. - № 8. - Ст. 126-127 с изм. и допол. в ред. от 24.02.23.

10 Приказ Минцифры "Об утверждении требований к обеспечению безопасности персональных данных при их обработке в информационных системах персональных данных" от 14.11.2022 № 232 // Официальный интернет-портал правовой информации

11 Приказ Роскомнадзора "Об утверждении требований к защите персональных данных при их обработке в информационных системах персональных данных" от 17.12.2022 № 119 // Официальный интернет-портал правовой информации

12 Приказ ФСТЭК России "Об утверждении требований к защите конфиденциальной информации" от 11.02.2013 № 55 // Официальный интернет-портал правовой информации. - с изм. и допол. в ред. от 10.01.2023.

Электронные источники

13 Главный по безопасности: проясняем роль и проходимся по болям CISO // Yandex Cloud Youtube Channel URL: <https://www.youtube.com/watch?v=HZCnIQmkyH4> (дата обращения: 5.06.2024).

14 Итоги пентестов — 2022 // Positive Technologies URL: <https://www.ptsecurity.com/ru-ru/research/analytics/results-of-pentests-2021-2022/> (дата обращения: 5.06.2024).

15 Сравнение систем класса External Attack Surface Management // CyberOK URL: <https://www.cyberok.ru/docs/CyberOK-EASM-Review-202312.pdf> (дата обращения: 5.06.2024).

16 2022 Data Breach Investigations Report // Verizon URL: <https://www.verizon.com/business/resources/reports/dbir/> (дата обращения: 5.06.2024).

17 Gartner Identifies Top Security and Risk Management Trends for 2022 // Gartner URL: <https://www.gartner.com/en/newsroom/press-releases/2022-03-07-gartner-identifies-top-security-and-risk-management-trends-for-2022> (дата обращения: 5.06.2024).

18 MASSCAN: Mass IP port scanner // Github URL: <https://github.com/robertdavidgraham/masscan/blob/master/README.md> (дата обращения: 5.06.2024).

19 Metasploitable 2 // Rapid7 URL: <https://docs.rapid7.com/metasploit/metasploitable-2/> (дата обращения: 5.06.2024).

20 Nikto 2.5 // CIRT.net URL: <https://cirt.net/Nikto2> (дата обращения: 5.06.2024).

21 Nmap Reference Guide, Chapter 15 // Nmap URL: <https://nmap.org/book/man.html#man-description> (дата обращения: 5.06.2024).

22 Proxmox Virtual Environment // Proxmox Virtual Environment URL: <https://www.proxmox.com/en/proxmox-virtual-environment/overview> (дата обращения: 5.06.2024).

23 Python: Getting Started // Welcome to Python.org URL: <https://www.python.org/about/> (дата обращения: 5.06.2024).

24 The External Attack Surface Management Landscape, Q1 2023 // Forrester URL: <https://reprints2.forrester.com/#/assets/2/2257/RES178691/report> (дата обращения: 31.05.2024).

25 The ZMap Project // The ZMap Project URL: <https://zmap.io/> (дата обращения: 5.06.2024).

26 Virtual Machines // Vulnerable By Design ~ VulnHub URL: <https://www.vulnhub.com/> (дата обращения: 5.06.2024).

27 Welcome to Scapy // Scapy URL: <https://scapy.net/> (дата обращения: 5.06.2024).

28 What is a TCP 3-way handshake process? // AfterAcademy URL: <https://afteracademy.com/blog/what-is-a-tcp-3-way-handshake-process/> (дата обращения: 5.06.2024).

29 What is Kali Linux & Kali's features // Kali Linux URL: <https://www.kali.org/docs/introduction/> (дата обращения: 5.06.2024).

ПРИЛОЖЕНИЕ

Код программы на Python:

```
#!/usr/bin/env -S sudo python3
```

```
"""
```

Проверка порта TCP с использованием Scapy

```
"""
```

```
#Подключение дополнительных библиотек
```

```
import os
```

```
import sys
```

```
import traceback
```

```
import g4f
```

```
from enum import IntEnum
```

```
from pathlib import Path
```

```
from random import randint
```

```
from typing import Dict, List
```

```
from argparse import ArgumentParser
```

```
from scapy.layers.inet import IP, TCP, ICMP
```

```
from scapy.packet import Packet
```

```
from scapy.sendrecv import sr1, sr
```

```
NON_PRIVILEGED_LOW_PORT = 1025
```

```
NON_PRIVILEGED_HIGH_PORT = 65534
```

```
ICMP_DESTINATION_UNREACHABLE = 3
```

```
#Класс для ответов на TCP-запросы
```

```
class TcpFlags(IntEnum):
```

```
    """
```

```
    CWR | ECE | URG | ACK | PSH | RST | SYN | FIN
```

```
    0  0  0  1  0  0  1  0 -> SYN + ACT
```

CWR | ECE | URG | ACK | PSH | RST | SYN | FIN

0 0 0 0 1 1 0 0 -> RST + PSH

.....

#Соединение установлено

SYNC_ACK = 0x12

#Соединение сброшено

RST_PSH = 0x14

#Класс для ответов на ICMP-запросы

class IcmpCodes(IntEnum):

.....

Коды ICMP:

0 Net is unreachable

1 Host is unreachable

2 Protocol is unreachable

3 Port is unreachable

4 Fragmentation is needed and Don't Fragment was set

5 Source route failed

6 Destination network is unknown

7 Destination host is unknown

8 Source host is isolated

9 Communication with destination network is administratively prohibited

10 Communication with destination host is administratively prohibited

11 Destination network is unreachable for type of service

12 Destination host is unreachable for type of service

13 Communication is administratively prohibited

14 Host precedence violation

15 Precedence cutoff is in effect

.....

Host_is_unreachable = 1

Protocol_is_unreachable = 2

Port_is_unreachable = 3

Communication_with_destination_network_is_administratively_prohibited = 9

Communication_with_destination_host_is_administratively_prohibited = 10

Communication_is_administratively_prohibited = 13

FILTERED_CODES = [x.value for x in IcmpCodes]

#Класс для ответов

class RESPONSES(IntEnum):

"""

Индивидуальные ответы на проверку наших портов

"""

#Фильтровано файерволом

FILTERED = 0

#Закрито

CLOSED = 1

#Открыто

OPEN = 2

#Ошибка

ERROR = 3

#Функция для загрузки списка сетевых ресурсов

def load_machines_port(the_data_file: Path) -> Dict[str, List[int]]:

"""

Парсинг csv-файла

"""

port_data = {}

with open(the_data_file, 'r', encoding="utf-8") **as** d_scan:

```

    for line in d_scan:
        host, ports = line.split(';')
        port_data[host] = [int(p) for p in ports.split(',')]
    return port_data

#Функция для попытки установки TCP-соединения с сетевым ресурсом
def test_port(
    address: str,
    dest_ports: int,
    verbose: bool = False
) -> RESPONSES:
    """
    Проверка комбинации адрес + порт
    :param address: Хост для проверки
    :param dest_ports: Порты для проверки
    :return: Ответные и неотвеченные пакеты (отфильтрованные)
    """

    src_port = randint(NON_PRIVILEGED_LOW_PORT,
NON_PRIVILEGED_HIGH_PORT)

    ip = IP(dst=address)
    ports = TCP(sport=src_port, dport=dest_ports, flags="S")
    reset_tcp = TCP(sport=src_port, dport=dest_ports, flags="S")
    packet: Packet = ip / ports
    verb_level = 0
    if verbose:
        verb_level = 99
        packet.show()
    try:
        answered = sr1(
            packet,

```

```

        verbose=verb_level,
        retry=1,
        timeout=1,
        threaded=True
    )
    if not answered:
        return RESPONSES.FILTERED
    elif answered.haslayer(TCP):
        if answered.getlayer(TCP).flags == TcpFlags.SYNC_ACK:
            rst_packet = ip / reset_tcp
            sr(rst_packet, timeout=1, verbose=verb_level)
            return RESPONSES.OPEN
        elif answered.getlayer(TCP).flags == TcpFlags.RST_PSH:
            return RESPONSES.CLOSED
        elif answered.haslayer(ICMP):
            icmp_type = answered.getlayer(ICMP).type
            icmp_code = int(answered.getlayer(ICMP).code)
            if icmp_type == ICMP_DESTINATION_UNREACHABLE and
icmp_code in FILTERED_CODES:
                return RESPONSES.FILTERED
    except TypeError:
        traceback.print_exc(file=sys.stdout)
        return RESPONSES.ERROR

#Главная функция, которая принимает на вход csv-файл
if __name__ == "__main__":
    #Проверка, что скрипт запущен с правами администратора
    if os.getuid() != 0:
        raise EnvironmentError("Sorry, you need to be root to run this program!")
    #Парсинг переданных аргументов

```

```

PARSER = ArgumentParser(description=__doc__)

PARSER.add_argument("--verbose", action="store_true", help="Toggle
verbose mode on/ off")

PARSER.add_argument("scan_file", type=Path, help="Scan file with list of
hosts and ports")

ARGS = PARSER.parse_args()

#ПРОМТ-запрос для GPT-чата
prompt="Написать подробный отчет без конфиденциальных данных об
результатах сканирования и найденных уязвимостях CVE и CWE, определить
CMS, рассчитать CVSS:\n"

#Загружаем список сетевых ресурсов из файла .csv
data = load_machines_port(ARGS.scan_file)

#Для каждого сетевого ресурса проверяем доступность
for machine in data:
    m_ports = data[machine]
    for dest_port in m_ports:
        ans = test_port(address=machine, dest_ports=dest_port,
verbose=ARGS.verbose)

        #Если ресурс доступен, то запускаем сканер nikto
        if(ans.name=="OPEN"):
            print(f"{ machine}:{ dest_port } -> {ans.name}")
            response = os.popen(f"nikto -h { machine } -p { dest_port}")
            nikto_output = response.readlines()
            for nikto_line in nikto_output:
                #Добавляем вывод сканера nikto в наш ПРОМТ-запрос для GPT-
чата

                prompt+=nikto_line.rstrip('\n')
                print(nikto_line.rstrip('\n'))
            else:
                print(f"{ machine}:{ dest_port } -> {ans.name}")

```

```

#Попробуем запросить отчёт у GPT-чата
try:
    response = g4f.ChatCompletion.create(
        model=g4f.models.gpt_4o,
        messages=[{"role": "user", "content": prompt}],
        provider=g4f.Provider.Liaobots,
        stream=True,
    )
    for message in response:
        print(message, flush=True, end="")
#Если GPT-чат недоступен, то сообщаем об этом
except Exception as e:
    print(f"{g4f.Provider.Liaobots.__name__}:", e)
    print("Извините, произошла ошибка. ЧатGPT для создания отчёта
недоступен!")

```