

Основы построения защищенных компьютерных сетей

Лекция 5 Metasploit. Введение

Семён Новосёлов

2024



БФУ
ИМЕНИ И. КАНТА

Что нам потребуется?

- VirtualBox

<https://www.virtualbox.org/>

- Metasploitable 2

<https://sourceforge.net/projects/metasploitable/>

- Metasploit

<https://www.metasploit.com/>

- Nmap

<https://nmap.org/>

есть в составе Kali Linux


<https://www.kali.org/get-kali/>

Metasploit Framework



www.metasploit.com

Фреймворк для пентеста и разработки эксплойтов

- Разработчик: **Rapid7**
- Есть две версии:
 - консольная
 - Web-интерфейс (Pro)
- Написан на  **Ruby**

Запуск:
`msfconsole`

A screenshot of the Metasploit console interface. At the top, it says 'msfconsole'. Below that is a large ASCII art of a duck. To the right of the duck, there is a 'HONK >' prompt. In the background, the word 'KALI' is visible in large letters, and below it, 'BY OFFENSIVE SECURITY'. At the bottom, there is a list of statistics for the Metasploit framework, followed by a tip and the prompt 'msf6 >'.

```
msfconsole  
  
      =[ metasploit v6.0.15-dev ]  
+ -- --=[ 2071 exploits - 1123 auxiliary - 352 post ]  
+ -- --=[ 592 payloads - 45 encoders - 10 nops ]  
+ -- --=[ 7 evasion ]  
  
Metasploit tip: Display the Framework log using the log command, learn more with help log  
msf6 > |
```

Модули Metasploit

exploits — модули для эксплуатации уязвимостей

auxiliary — модули для сбора информации, сканеры, ...

payload — полезная нагрузка эксплойтов

post — модули для постэксплуатации (сбор логинов, паролей, хешей, ...)

Поиск/использование/настройка:

search [ключевое_слово]

use [модуль]

show options

set [опция] [значение]

run

Расположение модулей

- Главное дерево модулей:
`/usr/share/metasploit-framework/modules/`
- Пользовательское:
`~/.msf4/modules/`
- Дополнительные пути можно указать ключом `-m`:
`msfconsole -m ./modules`

Порядок работы

1. Разведка: Сканирование целевой сети или хоста
 - модули из раздела `auxiliary`
 - сканер Nmap (команда `db_nmap`)
2. Выбор и настройка эксплойта:
 - поиск и выбор подходящих модулей: `search / use`
 - установка настроек: `show options / set`
3. Запуск эксплойта
4. Постэксплуатация
 - выполнение команд на цели
 - вытаскивание данных из цели (модули `post`)

База данных MSF

- Используется для хранения результатов работы модулей (обычно postgresql)
 - хосты, информация о версиях ПО на них
 - подобранные пароли и т.д.

Первый запуск

```
service postgresql start  
msfdb init  
msfconsole  
db_status
```

Последующие запуски

```
service postgresql start  
msfconsole  
db_status
```

Работа с базой данных

db_nmap сканирование сети с помощью nmap и запись результатов в базу

creds вывод ключевой информации (логины, пароли, хэши, ...)

loot собранная информация (дампы баз, пользовательская история и т.п.)

hosts список хостов

services список сервисов их версий

vulns найденные уязвимости

notes заметки (остальная полезная информация)

Аудит виртуальной машины с Metasploitable

Metasploitable 2

<https://sourceforge.net/projects/metasploitable/>

Специализированная виртуальная машина для тренировки проведения тестовых вторжений.

Справка: Команды msfconsole

Общие команды

`help, quit, exit`

`set/setg` установка параметров

`show` вывод списка модулей и опций

`show exploits, show payloads, ...`

`reload_all` обновить список модулей

> нераспознанные команды передаются в ОС

Использование модулей

`search` поиск модулей по ключевым словам

`use` загрузка модуля

`use exploit/unix/ftp/vsftpd_234_backdoor`

`show options` вывод опций модуля

`show payloads` вывод поддерживаемой целевой нагрузки

`show targets` вывод поддерживаемых целей

`edit` редактирование исходного кода модуля в редакторе

`info` вывод информации о модуле

`check` проверка цели на уязвимость без использования

`run` запуск модуля на выполнение

Разработка простого модуля

Задача: Написать сканер, который сканирует сеть на наличие уязвимых версий сервера Nginx.

- Факт наличия уязвимости проверяется по версии в http-заголовках.
- За основу возьмем модуль: `auxiliary/scanner/http/http_version`

Скелет модуля

```
require 'rex/proto/http'

class MetasploitModule < Msf::Auxiliary
  include Msf::Exploit::Remote::HttpClient
  include Msf::Auxiliary::Scanner
  include Msf::Auxiliary::Report

  def initialize
    super('Name' => 'Nginx Vulnerability scanner',
          # ...
        )

    register_options([
      OptString.new('TARGETURI', [ true, 'The URI to use', '/'])
    ])
  end
end
```

Инициализация

```
def run_host(ip)
  begin
    connect
    res = send_request_raw({
      'uri' => datastore['TARGETURI'],
      'method' => 'GET'
    })
    # ...
  end
  rescue ::Timeout::Error, ::Errno::EPIPE
  ensure
    disconnect
  end
end
end
```

Метод, выполняемый для
каждого хоста (в RHOSTS)

Функция для проверки на уязвимость

```
def run_host(ip)
  begin
    connect
    res = send_request_raw({ 'uri' => datastore['TARGETURI'], 'method' => 'GET' })
    fp = http_fingerprint(:response => res)
    print_status("#{ip}: \"#{fp}\"") if fp
    if fp && fp =~ /nginx\/([0-9]+)\.([0-9]+)\.?(?:[0-9]+)?/i
      major = $1.to_i; minor = $2.to_i; patch = $3.to_i
      version = "#{major}.#{minor}.#{patch.to_i}"
      nginx_vulns = []
      if major == 1 && (minor == 5 && patch.between?(0,11) || minor == 4 && patch.between?(0,3) ||
        minor == 3 && patch.between?(15,16))
        nginx_vulns.push "CVE-2014-0133"
      end
      if nginx_vulns.size > 0
        print_good("#{ip}: nginx #{version} - " + nginx_vulns.join(", "))
        report_vuln(:host => ip, :port => rport, :name => "Vulnerable nginx version: #{version}",
          :refs => nginx_vulns)
      else
        print_status("#{ip}: nginx #{version} - not vulnerable")
      end
    end
  end
  # ...
end
```

Литература и ссылки

- Документация к MSF:
<https://docs.rapid7.com/metasploit/>
- Курс Metasploit Unleashed, включающий в себя разработку модулей:
<https://www.offensive-security.com/metasploit-unleashed/>
- Интерактивные курсы по Ruby:
 - Try Ruby (<https://try.ruby-lang.org/>)
 - Ruby Monk (<https://rubymonk.com/>)
 - Codecademy (<https://www.codecademy.com/learn/learn-ruby>)