

利用 metasploit 进行信息收集

在进行信息收集的时候，我们既要全面详细的获取目标的信息，又要尽量隐藏自己不被发现。Metasploit 作为一个非常全面的渗透工具，用来收集信息也非常好用。本文会详细的介绍如何利用 Metasploit 进行信息收集。

信息收集分为主动和被动两种方式。

1、被动信息收集

被动信息收集是指在不直接接触目标系统的情况下寻找信息。比如，通过搜索引擎等方式可以获得目标的操作系统，开放的端口，web 服务器软件等信息。

2、主动信息收集

主动信息收集中，我们可以直接和系统交互，从而获得更多的信息。比如通过扫描目标系统开放的端口来确定对方开放的服务。每一个开放的服务都可能给我们提供了入侵的机会。需要注意的是，主动的信息收集很可能被 IDS 和 IPS 抓住踪迹。

2.1 启动 msfconsole

首先启动数据库

```
service postgresql start
```

启动 msfploit 服务

```
service metasploit start
```

```
root@hr-X:~# service postgresql start
[ ok ] Starting PostgreSQL 9.1 database server: main.
root@hr-X:~# service metasploit start
[ ok ] Starting Metasploit rpc server: prosv.
[ ok ] Starting Metasploit web server: thin.
[ ok ] Starting Metasploit worker: worker.
```

启动 msfconsole

```
msfconsole
```

2.2 使用 db_status 确认数据库是否正确连接

```
root@hr-X:~# msfconsole

Metasploit

Using notepad to track pentests? Have Metasploit Pro report on hosts,
services, sessions and evidence -- type 'go_pro' to launch it now.

*=[ metasploit v4.8.2-2014010101 [core:4.8 api:1.0]
+ -- --[ 1256 exploits - 762 auxiliary - 212 post
+ -- --[ 324 payloads - 32 encoders - 8 nops

msf > db_status
[*] postgresql connected to msf3
```

2.3 将 nmap 的扫描结果导入到 Metasploit

当一个团队同时做事的时候，会有针对不同目标，不同时间的扫描结果。这时候知道如何把 xml 格式的 nmap 结果导入到 metasploit 框架里有助于高效的工作。

首先，使用 -ox 参数扫描我们的 windows 虚拟机，生成扫描结果的 xml 格式文件。

```
#nmap -Pn -sS -A -oX Target 192.168.20.0/24
```

```

Nmap scan report for 192.168.20.128
Host is up (0.0011s latency).
Not shown: 994 closed ports
PORT      STATE SERVICE        VERSION
80/tcp    open  http           Apache httpd 2.4.4 ((Win32) OpenSSL/0.9.8y PHP/5.4.19)
|_ http-methods: No Allow or Public header in OPTIONS response (status code 302)
|_ http-title: Object not found!
|_ Requested resource was splash.php
135/tcp   open  msrpc          Microsoft Windows RPC
139/tcp   open  netbios-ssn    Microsoft Windows XP microsoft-ssn
445/tcp   open  microsoft-ds   Microsoft Windows XP microsoft-ds
3306/tcp  open  mysql          MySQL (unauthorized)
8080/tcp  open  http           Apache httpd 2.4.4 ((Win32) OpenSSL/0.9.8y PHP/5.4.19)
|_ http-methods: No Allow or Public header in OPTIONS response (status code 302)
|_ http-open-proxy: Proxy might be redirecting requests
|_ http-title: Did not follow redirect to http://192.168.20.128/xampp/
MAC Address: 00:0C:29:63:4C:38 (VMware)
Device type: general purpose
Running: Microsoft Windows XP
OS CPE: cpe:/o:microsoft:windows_xp::sp2 cpe:/o:microsoft:windows_xp::sp3
OS details: Microsoft Windows XP SP2 or SP3
Network Distance: 1 hop
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_ nbstat: NetBIOS name: MR-9F58607A2A6E, NetBIOS user: <unknown>, NetBIOS MAC: 00:0c:29:63:4c:38 (VMware)
|_ smb-os-discovery:
|   OS: Windows XP (Windows 2000 LAN Manager)
|   OS CPE: cpe:/o:microsoft:windows_xp::-
|   Computer name: mr-9f58607a2a6e
|   NetBIOS computer name: MR-9F58607A2A6E

```

使用 db_import 命令导入扫描结果到数据库中。使用 hosts 命令查看刚刚导入的数据。

```

msf > db_import Subnet1.xml

msf> hosts

```

```

msf > db_import Target
[*] Importing 'Nmap XML' data
[*] Import: Parsing with 'Nokogiri v1.6.0'
[*] Importing host 192.168.20.1
[*] Importing host 192.168.20.128
[*] Importing host 192.168.20.134
[*] Successfully imported /root/Target
msf > hosts

Hosts
=====
address      mac          name  os_name  os_flavor
os_sp purpose info  comments
-----
-----
192.168.20.1 00:50:56:C0:00:01 Microsoft Windows 7
device
192.168.20.128 00:0C:29:63:4C:38 Microsoft Windows XP
server
192.168.20.133

```

2.4 从 MSFconsole 中启动 nmap

在 msfconsole 中使用 db_nmap 命令启动扫描，可以将结果自动存储在数据库中。

```
#msf > db_nmap -sS -A 172.16.32.131
```

```
msf > db_nmap -sS -A 192.168.20.128
[*] Nmap: Starting Nmap 6.40 ( http://nmap.org ) at 2014-04-09 18:00 IST
[*] Nmap: Nmap scan report for 192.168.20.128
[*] Nmap: Host is up (0.0019s latency).
[*] Nmap: Not shown: 994 closed ports
[*] Nmap: PORT      STATE SERVICE      VERSION
[*] Nmap: 80/tcp    open  http         Apache httpd 2.4.4 ((Win32) OpenSSL/0.9.8y PHP/5.4.19)
[*] Nmap: |_http-methods: No Allow or Public header in OPTIONS response (status code 302)
[*] Nmap: |_http-title: Object not found!
[*] Nmap: |_Requested resource was splash.php
[*] Nmap: 135/tcp   open  msrpc        Microsoft Windows RPC
[*] Nmap: 139/tcp   open  netbios-ssn  Microsoft Windows XP microsoft-ds
[*] Nmap: 445/tcp   open  microsoft-ds Microsoft Windows XP microsoft-ds
[*] Nmap: 3306/tcp  open  mysql        MySQL (unauthorized)
[*] Nmap: 8080/tcp  open  http         Apache httpd 2.4.4 ((Win32) OpenSSL/0.9.8y PHP/5.4.19)
[*] Nmap: |_http-methods: No Allow or Public header in OPTIONS response (status code 302)
[*] Nmap: |_http-open-proxy: Proxy might be redirecting requests
[*] Nmap: |_http-title: Did not follow redirect to http://192.168.20.128/xampp/
[*] Nmap: MAC Address: 00:0C:29:63:4C:38 (VMware)
[*] Nmap: Device type: general purpose
[*] Nmap: Running: Microsoft Windows XP
[*] Nmap: OS CPE: cpe:/o:microsoft:windows_xp::sp2 cpe:/o:microsoft:windows_xp::sp3
[*] Nmap: OS details: Microsoft Windows XP SP2 or SP3
[*] Nmap: Network Distance: 1 hop
[*] Nmap: Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows
[*] Nmap: Host script results:
[*] Nmap: |_nbstat: NetBIOS name: MR-9F58607A2A6E, NetBIOS user: <unknown>, NetB
```

使用 db_services 命令查看存储在数据库中的扫描结果。

```
#msf > db_services
```



```
msf > services

Services
=====

host      port  proto name      state info
----
192.168.20.1 135  tcp  msrpc     open  Microsoft Windows RPC
192.168.20.1 139  tcp  netbios-ssn open  Microsoft Windows RPC
192.168.20.1 443  tcp  http      open  VMware VirtualCenter Web service
192.168.20.1 445  tcp  netbios-ssn open  Microsoft Windows RPC
192.168.20.1 554  tcp  rtsp      open  Realtek RTSP
192.168.20.1 902  tcp  vmware-auth open  VMware Authentication Daemon 1.10 Uses VM
C, SOAP
192.168.20.1 912  tcp  vmware-auth open  VMware Authentication Daemon 1.0 Uses VM
, SOAP
192.168.20.1 2869  tcp  http      open  Microsoft HTTPAPI httpd 2.0 SSDP/UPnP
192.168.20.1 6646  tcp  unknown   open
192.168.20.1 10243  tcp  http      open  Microsoft HTTPAPI httpd 2.0 SSDP/UPnP
192.168.20.1 49157  tcp  msrpc     open  Microsoft Windows RPC
192.168.20.1 49153  tcp  msrpc     open  Microsoft Windows RPC
192.168.20.1 49154  tcp  msrpc     open  Microsoft Windows RPC
192.168.20.1 49156  tcp  msrpc     open  Microsoft Windows RPC
192.168.20.1 49152  tcp  msrpc     open  Microsoft Windows RPC
192.168.20.128 80  tcp  http      open  Apache httpd 2.4.4 (Win32) OpenSSL/0.9.8
PHP/5.4.19
192.168.20.128 135  tcp  msrpc     open  Microsoft Windows RPC
192.168.20.128 139  tcp  netbios-ssn open  Microsoft Windows RPC
192.168.20.128 445  tcp  microsoft-ds open  Microsoft Windows XP microsoft-ds
192.168.20.128 3306  tcp  mysql     open  MySQL unauthorized
192.168.20.128 8080  tcp  http      open  Apache httpd 2.4.4 (Win32) OpenSSL/0.9.8
PHP/5.4.19
192.168.20.134 22  tcp  ssh       open  OpenSSH 4.7p1 Debian 8ubuntu1 protocol 2
```

2.5 使用 metasploit 自带的端口扫描器

metasploit 自带了多个端口扫描器，我们也可以利用这些端口扫描器来寻找目标。查看端口扫描器列表可以使用 search 命令。

```
#msf > search portscan
```

```
msf > search portscan

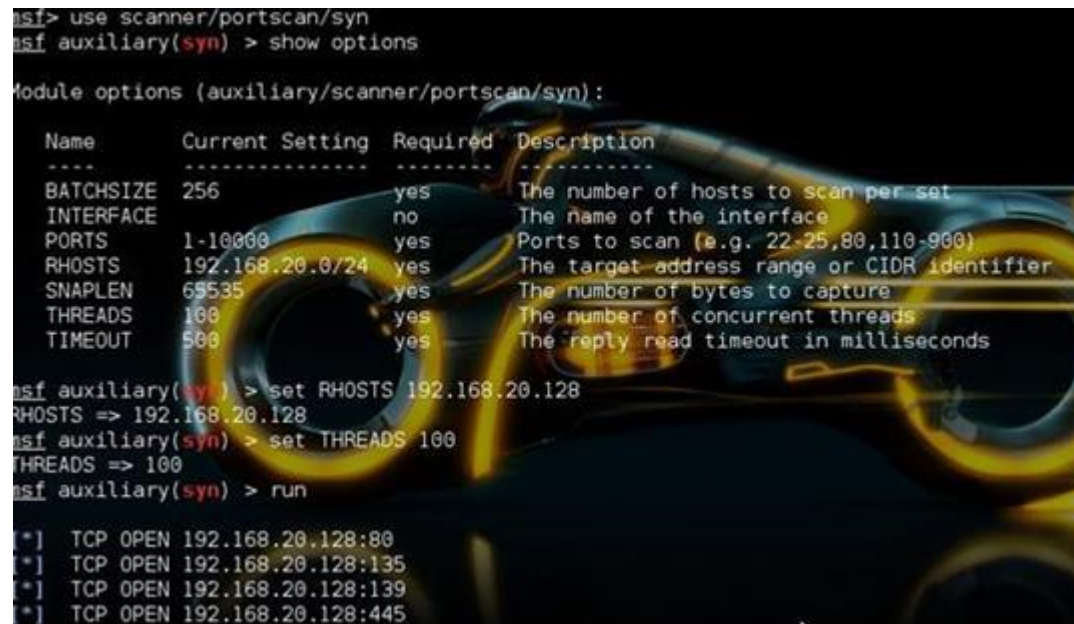
Matching Modules
=====

Name                                     Disclosure Date Rank Description
----
auxiliary/scanner/http/wordpress_pingback_access normal WordPress Pingback Locator
auxiliary/scanner/http/wordpress_pingback_access normal WordPress Pingback Locator
auxiliary/scanner/natpmp/natpmp_portscan normal NAT-PMP External Port Scanner
auxiliary/scanner/natpmp/natpmp_portscan normal NAT-PMP External Port Scanner
auxiliary/scanner/portscan/ack normal TCP ACK Firewall Scanner
auxiliary/scanner/portscan/ack normal TCP ACK Firewall Scanner
auxiliary/scanner/portscan/ftpbounce normal FTP Bounce Port Scanner
auxiliary/scanner/portscan/ftpbounce normal FTP Bounce Port Scanner
auxiliary/scanner/portscan/syn normal TCP SYN Port Scanner
auxiliary/scanner/portscan/syn normal TCP SYN Port Scanner
auxiliary/scanner/portscan/tcp normal TCP Port Scanner
auxiliary/scanner/portscan/tcp normal TCP Port Scanner
auxiliary/scanner/portscan/xmas normal TCP "XMas" Port Scanner
auxiliary/scanner/portscan/xmas normal TCP "XMas" Port Scanner
```

我们来使用 SYN 端口扫描器进行一次简单的扫描练习一下，使用

scanner/portscan/syn ,设置 RHOSTS 为 192.168.20.0/24,设置线程 THREADS 为 100, 然后使用 run 命令开始扫描。

```
#msf > use scanner/portscan/syn
```



```
msf> use scanner/portscan/syn
msf auxiliary(syn) > show options

Module options (auxiliary/scanner/portscan/syn):

  Name      Current Setting  Required  Description
  ----      -
  BATCHSIZE  256              yes       The number of hosts to scan per set
  INTERFACE  nil              no        The name of the interface
  PORTS      1-10000          yes       Ports to scan (e.g. 22-25,80,110-900)
  RHOSTS     192.168.20.0/24 yes       The target address range or CIDR identifier
  SNAPLEN    65535            yes       The number of bytes to capture
  THREADS    100              yes       The number of concurrent threads
  TIMEOUT    500              yes       The reply read timeout in milliseconds

msf auxiliary(syn) > set RHOSTS 192.168.20.128
RHOSTS => 192.168.20.128
msf auxiliary(syn) > set THREADS 100
THREADS => 100
msf auxiliary(syn) > run

[*] TCP OPEN 192.168.20.128:80
[*] TCP OPEN 192.168.20.128:135
[*] TCP OPEN 192.168.20.128:139
[*] TCP OPEN 192.168.20.128:445
```

2.6 Server Message Block Scanning

metasploit 可以通过 smb_version 模块来尝试识别 windows 的版本。

```
#msf > use scanner/smb/smb_version
```



```
msf> use scanner/smb/smb_version
msf auxiliary(smb_version) > show options

Module options (auxiliary/scanner/smb/smb_version):

  Name      Current Setting  Required  Description
  ----      -
  RHOSTS     192.168.20.128  yes       The target address range or CIDR identifier
  SMBDomain  WORKGROUP        no        The Windows domain to use for authentication
  SMBPass    nil              no        The password for the specified username
  SMBUser    nil              no        The username to authenticate as
  THREADS    100              yes       The number of concurrent threads

msf auxiliary(smb_version) > set RHOSTS 192.168.20.128
RHOSTS => 192.168.20.128
msf auxiliary(smb_version) > set THREADS 100
THREADS => 100
msf auxiliary(smb_version) > run
```

扫描结果会存储在 metasploit 的数据库中，使用 hosts 命令可以查看。

```
#msf auxiliary(smb_version) > hosts
```

```
msf auxiliary(mssql_ping) > hosts
```

Hosts									

address	mac	name	os_name	os_flavor	os_sp	purpose	info	comments	
-----	---	----	-----	-----	-----	-----	-----	-----	-----
192.168.20.1	00:50:56:C0:00:01		Microsoft Windows	7		device			
192.168.20.128	00:0C:29:63:4C:38		Microsoft Windows	XP	SP3	client			

2.7 收集 MS SQL server 信息

很多系统管理员自己都没有意识到自己的服务器上可能已经安装了 MS SQLserver。因为安装一些软件需要预装数据库，比如 Microsoft Visual Studio。默认 MS SQL server 会监听 1433 端口或者一个随机的 TCP 端口。如果监听的是随机端口的话，可以通过 UDP 在 1434 端口查询具体监听的是哪个端口。

metasploit 有一个模块可以自动实现这些事情。叫做 mssql_ping。

```
#msf > use scanner/mssql/mssql_ping
msf auxiliary(mssql_ping) > set RHOSTS 192.168.1.0/24
RHOSTS => 192.168.1.0/24
msf auxiliary(mssql_ping) > set THREADS 255
THREADS => 255
msf auxiliary(mssql_ping) > run
```



```

msf > use scanner/mssql/mssql_ping
msf auxiliary(mssql_ping) > show options

Module options (auxiliary/scanner/mssql/mssql_ping):

  Name          Current Setting  Required  Description
  ----          -
  PASSWORD      no               no        The password for the specified username
  RHOSTS        yes              yes        The target address range or CIDR identifier
  THREADS       1               yes        The number of concurrent threads
  USERNAME      sa               no        The username to authenticate as
  USE_WINDOWS_AUTH false            yes        Use windows authentication (requires DOMAIN option set)

msf auxiliary(mssql_ping) > set RHOSTS 192.168.20.0/24
RHOSTS => 192.168.20.0/24
msf auxiliary(mssql_ping) > set THREADS 100
THREADS => 100
msf auxiliary(mssql_ping) > run

[*] Scanned 100 of 256 hosts (039% complete)
[*] SQL Server information for 192.168.20.120:
[*] ServerName = MR-9F58607A2A6E
[*] InstanceName = SQLEXPRESS
[*] IsClustered = No
[*] Version = 9.00.1399.06
[*] tcp = 1035
[*] np = \\MR-9F58607A2A6E\pipe\MSSQL$SQLEXPRESS\sql\query
[*] via = MR-9F58607A2A6E,0:1433
[*] Scanned 152 of 256 hosts (059% complete)
[*] Scanned 200 of 256 hosts (078% complete)
[*] Scanned 208 of 256 hosts (081% complete)
[*] Scanned 231 of 256 hosts (090% complete)

```

如上图所示 ,metasploit 除了可以获得监听的端口 ,还可以获得实例的名字 ,服务器版本。

2.8 获取 SSH 版本信息

ssh 是一种安全协议 ,有很多 ssh 的实现都被发现过漏洞。所以我们需要先识别 ssh 使用的软件版本。采用 ssh_version 模块。

```
msf > use scanner/ssh/ssh_version
```

```

msf auxiliary(mssql_ping) > use scanner/ssh/ssh_version
msf auxiliary(ssh_version) > show options

Module options (auxiliary/scanner/ssh/ssh_version):

  Name      Current Setting  Required  Description
  ----      -
  RHOSTS    yes              yes        The target address range or CIDR identifier
  RPORT     22               yes        The target port
  THREADS   1               yes        The number of concurrent threads
  TIMEOUT   30               yes        Timeout for the SSH probe

msf auxiliary(ssh_version) > set RHOSTS 192.168.20.134
RHOSTS => 192.168.20.134
msf auxiliary(ssh_version) > set THREADS 50
THREADS => 50
msf auxiliary(ssh_version) > run

[*] 192.168.20.134:22, SSH server version: SSH-2.0-OpenSSH_4.7p1 Debian-8ubuntu1
[*] Scanned 1 of 1 hosts (100% complete)

```


2.9 扫描 FTP 版本

FTP 服务器通常都是一个网络中最薄弱的地方，一定要仔细的扫描目标网络中的 FTP 服务器。可以使用 ftp_version 模块来寻找目标网络中的 FTP server。

```
msf > use auxiliary/scanner/ftp/ftp_version
msf auxiliary(ftp_version) > show options

Module options (auxiliary/scanner/ftp/ftp_version):

  Name      Current Setting  Required  Description
  ----      -
  FTPPASS   mozilla@example.com no         The password for the specified username
  FTPUSER   anonymous        no         The username to authenticate as
  RHOSTS    yes             yes        The target address range or CIDR identifier
  RPORT     21              yes        The target port
  THREADS   1               yes        The number of concurrent threads

msf auxiliary(ftp_version) > set RHOSTS 192.168.20.134
RHOSTS => 192.168.20.134
msf auxiliary(ftp_version) > set THREADS 50
THREADS => 50
msf auxiliary(ftp_version) > run

[*] 192.168.20.134:21 FTP Banner: '220 (vsFTPd 2.3.4)\x0d\x0a'
```

我们幸运的找到了一台 FTP 服务器，现在试试能不能匿名登陆。使用 scanner/ftp/anonymous 模块。扫描器结果显示，可以匿名登陆，但是只有读的权限。

```
msf auxiliary(ftp_version) > use auxiliary/scanner/ftp/anonymous
msf auxiliary(anonymous) > show options

Module options (auxiliary/scanner/ftp/anonymous):

  Name      Current Setting  Required  Description
  ----      -
  FTPPASS   mozilla@example.com no         The password for the specified username
  FTPUSER   anonymous        no         The username to authenticate as
  RHOSTS    yes             yes        The target address range or CIDR identifier
  RPORT     21              yes        The target port
  THREADS   1               yes        The number of concurrent threads

msf auxiliary(anonymous) > set RHOSTS 192.168.20.134
RHOSTS => 192.168.20.134
msf auxiliary(anonymous) > set THREADS 50
THREADS => 50
msf auxiliary(anonymous) > run

[*] 192.168.20.134:21 Anonymous READ (220 (vsFTPd 2.3.4))
```

使用 metasploit 进行信息收集主要的方法就是上文所述了，接下来可以开始利用 metasploit 进行漏洞扫描和利用。