UV F2B505
SECURITE INFORMATIQUE ET RESEAUX
TP INTRUSION DETECTION SYSTEMS - SNORT

VO LE MINH QUAN
KOUNGA FREDERIC FRANCK

1. LAB ENVIRONMENT

TELECHARGEMENT

- Aller sur le site http://wiki.netkit.org/index.php/Download_Official
  - ✓ Il y a 3 archives zippées à récupérer : Netkit Core, Netkit FileSystem, Netkit Kernel
- Télécharger ces 3 archives et mettre dans le répertoire */users/local/lvo/MyNetKit*
- Décompresser les 3 archives dans le même répertoire avec l'option *-S* (spare files)
  - ✓ *tar -xjSf fichier.tar.bz2*

CONFIGURATION LES VARIABLES D'ENVIRONNEMENT

- Sur le terminal, faire
  - ✓ *export NETKIT_HOME=/users/local/lvo/MyNetKit/netkit*
  - ✓ *export MANPATH=$MANPATH::/users/local/lvo/MyNetKit/netkit/man*
  - ✓ *export PATH=$NETKIT_HOME/bin:$PATH*

TESTER LA CONFIGURATION

- En faisant
  - ✓ *cd netkit*
  - ✓ *./check_configuration.sh*

DEMARRER ET ETEINDRE DES MACHINES VIRTUELLES

- Faire *lstart* sur le répertoire *lab-ids/* pour lancer toutes les machines virtuelles
- Pour s'arrêter, on tape *poweroff* sur chaque machine et après *lhalt* sur *lab-ids/*

2. SNORT

CONFIGURER L'INTERFACE ETH0 SUR LA MACHINE IDS

- *ifconfig eth0 promisc up*

OBSERVER DES ATTAQUES AVEC SNORT

- Lancer *snort* avec *l'ids*
  - ✓ */etc/init.d/snort start*
- Lancer sur le pc1
  - ✓ *ftp ipOfPC8*
  - ✓ *name : anonymous*
  - ✓ *notremotdepasse*

COMMANDE CD

- *cd dir*
  - Avec *dir* est un nom de dossier non existant et trop longue (plus de 100 caractères)
    => Les params de la commande FTP sont trop longs (125:3:1), et la réponse de FTP est trop longue (125:6:1)

```
125 || 1 || ftp_pp: Telnet command on FTP command channel
125 || 2 || ftp_pp: Invalid FTP command
125 || 3 || ftp_pp: FTP parameter length overflow
125 || 4 || ftp_pp: FTP malformed parameter
125 || 5 || ftp_pp: Possible string format attempt in FTP command/parameter
125 || 6 || ftp_pp: FTP response length overflow
125 || 7 || ftp_pp: FTP command channel encrypted
```

COMMANDE LS

- *ls ../..*
  => L'alerte de type « FTP LIST directory traversal attempt » 1:1992:8 avec 1 est la règle générale, 8 est la version
  => L'essai d'aller vers la racine du répertoire qu'on a le droit d'accès

```
1 || 1 || snort general alert
2 || 1 || tag: Tagged Packet
3 || 1 || snort dynamic alert
100 || 1 || spp_portscan: Portscan Detected
```

3. SCANNING A TARGET

- Lire des portes TCP ouvertes
  - ✓ *nmap ipDuPC8*

```
pc1:~# nmap 192.168.8.2

Starting Nmap 4.68 ( http://nmap.org ) at 2019-02-26 13:35 UTC
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled.
 Try using --system-dns or specify valid servers with --dns-servers
Interesting ports on 192.168.8.2:
Not shown: 1712 closed ports
PORT     STATE SERVICE
21/tcp   open  ftp
111/tcp  open  rpcbind
2000/tcp open  callbook

Nmap done: 1 IP address (1 host up) scanned in 0.702 seconds
```

- Lire le système d'exploitation
  - ✓ *nmap -U ipDuPC8*

```
pc1:~# nmap -U 192.168.8.2

Starting Nmap 4.68 ( http://nmap.org ) at 2019-02-26 13:46 UTC
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled.
 Try using --system-dns or specify valid servers with --dns-servers
Interesting ports on 192.168.8.2:
Not shown: 1712 closed ports
PORT     STATE SERVICE
21/tcp   open  ftp
111/tcp  open  rpcbind
2000/tcp open  callbook
Device type: general purpose
Running: Linux 2.6.X
OS details: Linux 2.6.11 - 2.6.20
Uptime: 0.033 days (since Tue Feb 26 12:59:16 2019)
Network Distance: 6 hops

OS detection performed. Please report any incorrect results at http://nmap.org/s
ubmit/ .
Nmap done: 1 IP address (1 host up) scanned in 2.114 seconds
```

- Lire des portes UDP ouvertes
  - ✓ *nmap -sP ipDuPC8*
  - ✓ On voit qu'il y a un service qui est en cours sur la porte 2002

```
Interesting ports on 192.168.8.2:
Not shown: 1486 closed ports
PORT     STATE         SERVICE
111/udp  open|filtered rpcbind
2002/udp open|filtered globe
```

4. EXPLOITING A WEAKNESS

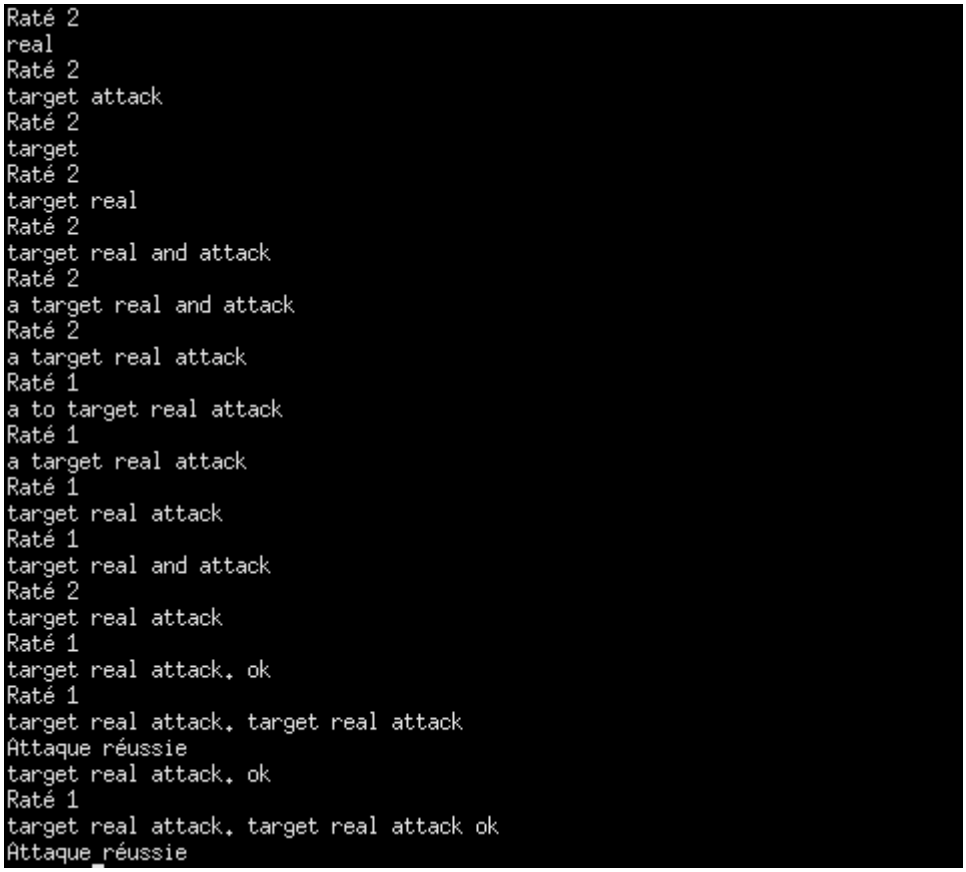
TROUVER LA PORTE

- On utilise *nmap* pour détecter la porte
  - ✓ *nmap -sP ipDuPC8*
  - ✓ Si elle est la porte correcte, le pc8 va imprimer sur sa console

ATTAQUER EN UTILISANT D'UN SCRIPT PYTHON

- Sur pc1, on fait une attaque avec
  - ✓ *python /hostlab/client votretexte*

- ✓ Le résultat de chaque attaque est affiche sur la machine pc8



DEVINER LA REGLE

- ❖ La règle pour réussir d'attaquer est un texte comme la suite
  - ✓ *[Caractères] [s1] [une espace] [Caractères] [s2] [Caractères] [s3] [Caractères]*
  - ✓ Avec s1, s2, s3 est un des mots « target », « real », « attack » et s1, s2, s3 sont différents

4.1 WRITING A RULE

- ❖ Avec le contenu de la charge util « content »
  - ✓ 1e essai
    - Faire l'avertissement tous les paquets traverses
    - *alert udp any any -> any any (msg: "first attempt"; sid: 123456789;)*
  - ✓ 2e essai
    - Tous les paquets avec son contenu est égal « ok », cas insensible avec « no-case »
    - *alert udp any any -> any any (msg: "second attempt"; content: "ok"; nocase; sid: 123456789;)*
- ❖ Avec « *pcre* » => l'expression régulière
  - ✓ 3e essai
    - La charge utile est égale « *real attacktarget* », insensible à la casse avec */i*
    - *alert udp any any -> any any (msg: "third attempt"; pcre: "/real attacktarget/i"; sid: 123456789;)*
  - ✓ 4e essai
    - La charge utile est comme « *[caractères]real attacktarget* »
    - Par exemple : « *qsfdfhdsreal attacktarget* »

- *alert udp any any -> any any (msg: "fourth attempt"; pcre: "/.*real\sattack-target/i"; sid: 123456789;)*

✓ 5e essai

- La charge utile est comme « *[caractères]real attack[caractères]target[caractères]* »
- Par exemple : « *qsfdfhdsreal attackdsqdsqdsqdtargetdsqdsqdsqd* »
- *alert udp any any -> any any (msg: "fifth attempt"; pcre: "/.*real\sattack.*target.*/i"; sid: 123456789;)*

✓ 6e essai

- La règle est presque la même que la 5e, mais avec la porte et l'adresse IP de la machine vulnérable, les paquets reçus est encore depuis n'importe quelle source
- *alert udp any any -> 192.168.8.2 2002 (msg: "sixth attempt"; pcre: "/.*real\sattack.*target.*/i"; sid: 123456789;)*

5. IDS EVASION TECHNIQUE

5.1 INSERTION

❖ Si on ne teste pas sur la machine pc1 => ça ne va pas marcher à cause de l'adresse IP de l'expéditeur

❖ *tcprewite* => réécrire l'adresse IP, la porte, etc.

❖ *frag3* pour détecter des anomalies => par exemple la faute de la segmentation

✓ *preprocessor frag3_global*

✓ *preprocessor frag3_engine : policy bsd*