

Cybersécurité des services informatiques

PROFESSEURS : MR JOBARD

TP N°5 : Kali linux

Objectif : Objectif : découvrir l'intérêt de sécuriser correctement une machine sous Windows ou Linux et savoir se protéger en se mettant à la place de l'attaquant via Kali Linux.

Introduction :

Dans le contexte actuel où les cybermenaces se multiplient, sécuriser correctement une machine, qu'elle fonctionne sous Windows ou Linux, est devenu une nécessité impérieuse. La protection de nos systèmes informatiques contre les attaques malveillantes est essentielle pour préserver l'intégrité, la confidentialité et la disponibilité des données. L'objectif de cette démarche est de découvrir l'intérêt de mettre en place des mesures de sécurité robustes et de comprendre les techniques utilisées par les attaquants pour mieux se protéger.

Pour ce faire, nous allons explorer l'utilisation de Kali Linux, une distribution spécialisée dans les tests d'intrusion et l'évaluation de la sécurité.

- 🌈 Tout d'abord on va installer kali linux. Notre machine virtuelle Kali Linux est prête, on peut l'utiliser.



- Ensuite on va identifier son IP et l'adresse MAC de sa carte réseau, en tapant cette commande `ip a`.

```
mame@kali: ~  
(mame@kali)-[~]  
└─$ ip a  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        valid_lft forever preferred_lft forever  
    inet6 ::1/128 scope host noprefixroute  
        valid_lft forever preferred_lft forever  
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000  
    link/ether 00:0c:29:51:2d:74 brd ff:ff:ff:ff:ff:ff  
    inet 192.168.140.139/24 brd 192.168.140.255 scope global dynamic noprefixroute eth0  
        valid_lft 1558sec preferred_lft 1558sec  
    inet6 fe80::20c:29ff:fe51:2d74/64 scope link noprefixroute  
        valid_lft forever preferred_lft forever  
(mame@kali)-[~]  
└─$
```

Ethernet Interface (eth0)

MAC Address: 00:0c:29:51:2d:74

IPv4 Address: 192.168.140.139

- Par la suite je vais installer une machine virtuelle sous **Windows** et une autre sous **Linux**. Puis identifie de la même manière les adresses IP/MAC correspondantes.

```
mame@debian: ~  
mame@debian:~$ ip a  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        valid_lft forever preferred_lft forever  
    inet6 ::1/128 scope host noprefixroute  
        valid_lft forever preferred_lft forever  
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000  
    link/ether 00:0c:29:e5:e6:3c brd ff:ff:ff:ff:ff:ff  
    altname enp2s1  
    inet 192.168.140.140/24 brd 192.168.140.255 scope global dynamic noprefixroute ens33  
        valid_lft 1663sec preferred_lft 1663sec  
    inet6 fe80::20c:29ff:fee5:e63c/64 scope link noprefixroute  
        valid_lft forever preferred_lft forever  
mame@debian:~$
```

MAC Address: 00:0c:29:e5:e6:3c

IPv4 Address: 192.168.140.140

```
C:\Windows\System32\cmd.exe
Ethernet adapter Ethernet0:

Connection-specific DNS Suffix . . : localdomain
Description . . . . . : Intel(R) 82574L Gigabit Network Connection
Physical Address. . . . . : 00-0C-29-8D-CD-83
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
Link-local IPv6 Address . . . . . : fe80::60dd:4260:c8a5:abd0%7(Preferred)
IPv4 Address. . . . . : 192.168.140.141(Preferred)
Subnet Mask . . . . . : 255.255.255.0
Lease Obtained. . . . . : Tuesday, June 25, 2024 5:43:50 PM
Lease Expires . . . . . : Tuesday, June 25, 2024 6:51:11 PM
Default Gateway . . . . . : 192.168.140.2
DHCP Server . . . . . : 192.168.140.254
DHCPv6 IAID . . . . . : 100666409
DHCPv6 Client DUID. . . . . : 00-01-00-01-2E-0D-97-C0-00-0C-29-8D-CD-83
DNS Servers . . . . . : 192.168.140.2
Primary WINS Server . . . . . : 192.168.140.2
NetBIOS over Tcpip. . . . . : Enabled

Ethernet adapter Bluetooth Network Connection:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . . :
Description . . . . . : Bluetooth Device (Personal Area Network)
Physical Address. . . . . : D0-A4-6F-82-B0-70
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes

C:\Windows\system32>
```

MAC Address: 00:0C:29:8D:CD:83

IPv4 Address : 192.168.140.141

✚ Après je vais Vérifier qu'elles sont capables de communiquer entre elles.

Ping kali-----» Debian

```
mame@kali: ~
inet6 fe80::20c:29ff:fe51:2d74/64 scope link noprefixroute
valid_lft forever preferred_lft forever

(mame@kali)-[~]
└─$ ping 192.168.140.140
PING 192.168.140.140 (192.168.140.140) 56(84) bytes of data:
64 bytes from 192.168.140.140: icmp_seq=1 ttl=64 time=0.759 ms
64 bytes from 192.168.140.140: icmp_seq=2 ttl=64 time=0.567 ms
64 bytes from 192.168.140.140: icmp_seq=3 ttl=64 time=0.535 ms
64 bytes from 192.168.140.140: icmp_seq=4 ttl=64 time=0.520 ms
64 bytes from 192.168.140.140: icmp_seq=5 ttl=64 time=0.555 ms
64 bytes from 192.168.140.140: icmp_seq=6 ttl=64 time=0.597 ms
64 bytes from 192.168.140.140: icmp_seq=7 ttl=64 time=0.434 ms
64 bytes from 192.168.140.140: icmp_seq=8 ttl=64 time=0.573 ms
64 bytes from 192.168.140.140: icmp_seq=9 ttl=64 time=0.354 ms
64 bytes from 192.168.140.140: icmp_seq=10 ttl=64 time=0.407 ms
64 bytes from 192.168.140.140: icmp_seq=11 ttl=64 time=0.552 ms
64 bytes from 192.168.140.140: icmp_seq=12 ttl=64 time=0.528 ms
64 bytes from 192.168.140.140: icmp_seq=13 ttl=64 time=0.303 ms
64 bytes from 192.168.140.140: icmp_seq=14 ttl=64 time=0.506 ms
64 bytes from 192.168.140.140: icmp_seq=15 ttl=64 time=0.423 ms
```

Debian-----» kali Linux

```
mame@debian: ~  
    inet6 ::1/128 scope host noprefixroute  
        valid_lft forever preferred_lft forever  
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP gro  
up default qlen 1000  
    link/ether 00:0c:29:e5:e6:3c brd ff:ff:ff:ff:ff:ff  
    altname enp2s1  
    inet 192.168.140.140/24 brd 192.168.140.255 scope global dynamic noprefixrou  
te ens33  
        valid_lft 1663sec preferred_lft 1663sec  
    inet6 fe80::20c:29ff:fee5:e63c/64 scope link noprefixroute  
        valid_lft forever preferred_lft forever  
mame@debian:~$ ping 192.168.140.139  
PING 192.168.140.139 (192.168.140.139) 56(84) bytes of data.  
64 bytes from 192.168.140.139: icmp_seq=1 ttl=64 time=0.625 ms  
64 bytes from 192.168.140.139: icmp_seq=2 ttl=64 time=0.508 ms  
64 bytes from 192.168.140.139: icmp_seq=3 ttl=64 time=0.499 ms  
64 bytes from 192.168.140.139: icmp_seq=4 ttl=64 time=0.462 ms  
64 bytes from 192.168.140.139: icmp_seq=5 ttl=64 time=0.395 ms  
64 bytes from 192.168.140.139: icmp_seq=6 ttl=64 time=0.458 ms  
64 bytes from 192.168.140.139: icmp_seq=7 ttl=64 time=0.500 ms  
64 bytes from 192.168.140.139: icmp_seq=8 ttl=64 time=0.553 ms  
^Z  
[1]+  Stoppé                               ping 192.168.140.139  
mame@debian:~$
```

Debian-----» Windows

```
mame@debian: ~  
mame@debian:~$ ping 192.168.140.141  
PING 192.168.140.141 (192.168.140.141) 56(84) bytes of data.  
64 bytes from 192.168.140.141: icmp_seq=1 ttl=128 time=1.23 ms  
64 bytes from 192.168.140.141: icmp_seq=2 ttl=128 time=0.735 ms  
64 bytes from 192.168.140.141: icmp_seq=3 ttl=128 time=0.661 ms  
64 bytes from 192.168.140.141: icmp_seq=4 ttl=128 time=0.583 ms  
64 bytes from 192.168.140.141: icmp_seq=5 ttl=128 time=0.675 ms  
64 bytes from 192.168.140.141: icmp_seq=6 ttl=128 time=1.05 ms  
64 bytes from 192.168.140.141: icmp_seq=7 ttl=128 time=0.498 ms  
64 bytes from 192.168.140.141: icmp_seq=8 ttl=128 time=0.530 ms  
64 bytes from 192.168.140.141: icmp_seq=9 ttl=128 time=0.640 ms  
64 bytes from 192.168.140.141: icmp_seq=10 ttl=128 time=0.614 ms  
64 bytes from 192.168.140.141: icmp_seq=11 ttl=128 time=0.504 ms  
64 bytes from 192.168.140.141: icmp_seq=12 ttl=128 time=0.563 ms  
64 bytes from 192.168.140.141: icmp_seq=13 ttl=128 time=0.481 ms  
64 bytes from 192.168.140.141: icmp_seq=14 ttl=128 time=0.586 ms  
64 bytes from 192.168.140.141: icmp_seq=15 ttl=128 time=0.525 ms  
64 bytes from 192.168.140.141: icmp_seq=16 ttl=128 time=0.579 ms  
64 bytes from 192.168.140.141: icmp_seq=17 ttl=128 time=0.549 ms  
64 bytes from 192.168.140.141: icmp_seq=18 ttl=128 time=0.571 ms  
64 bytes from 192.168.140.141: icmp_seq=19 ttl=128 time=0.641 ms  
64 bytes from 192.168.140.141: icmp_seq=20 ttl=128 time=0.509 ms  
64 bytes from 192.168.140.141: icmp_seq=21 ttl=128 time=0.601 ms  
64 bvtes from 192.168.140.141: icm0 seq=22 ttl=128 time=0.526 ms
```


Windows-----» Linux/Debian et Windows-----» Kali Linux

```
C:\Windows\System32\cmd.exe

C:\Windows\system32>ping 192.168.140.140

Pinging 192.168.140.140 with 32 bytes of data:
Reply from 192.168.140.140: bytes=32 time<1ms TTL=64
Reply from 192.168.140.140: bytes=32 time=1ms TTL=64
Reply from 192.168.140.140: bytes=32 time<1ms TTL=64
Reply from 192.168.140.140: bytes=32 time=1ms TTL=64

Ping statistics for 192.168.140.140:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

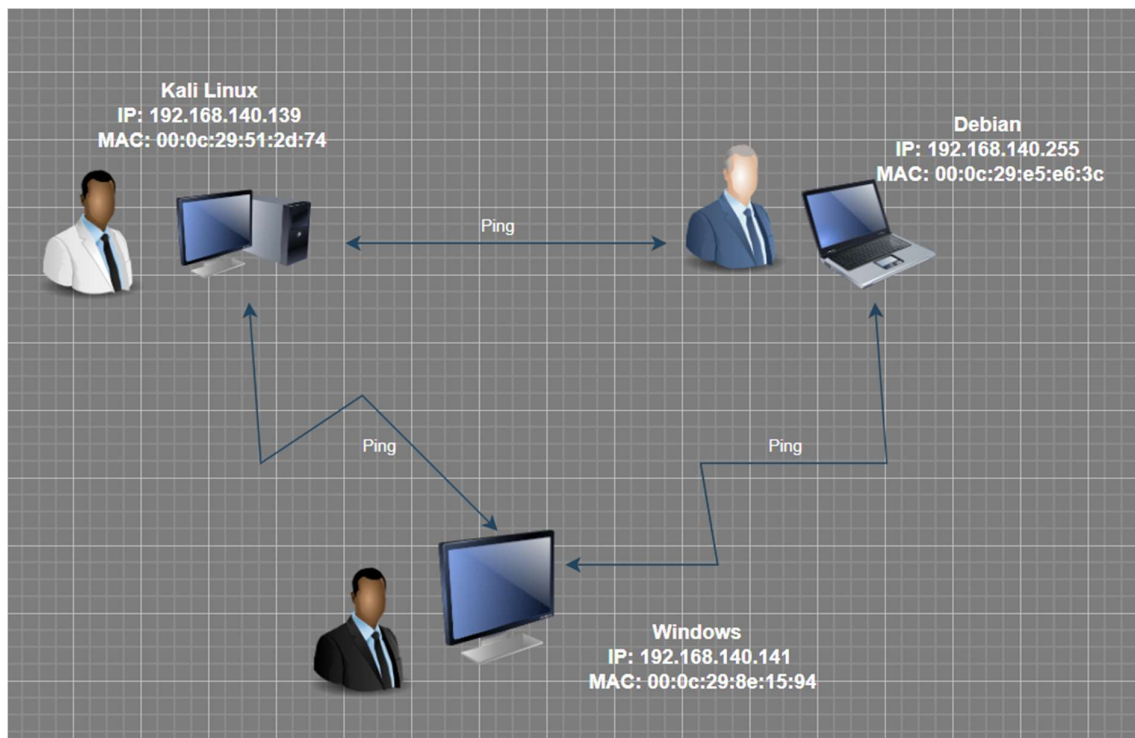
C:\Windows\system32>ping 192.168.140.129

Pinging 192.168.140.129 with 32 bytes of data:
Reply from 192.168.140.129: bytes=32 time=1ms TTL=64
Reply from 192.168.140.129: bytes=32 time=1ms TTL=64
Reply from 192.168.140.129: bytes=32 time<1ms TTL=64
Reply from 192.168.140.129: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.140.129:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\Windows\system32>
```

🔗 Schéma infrastructures

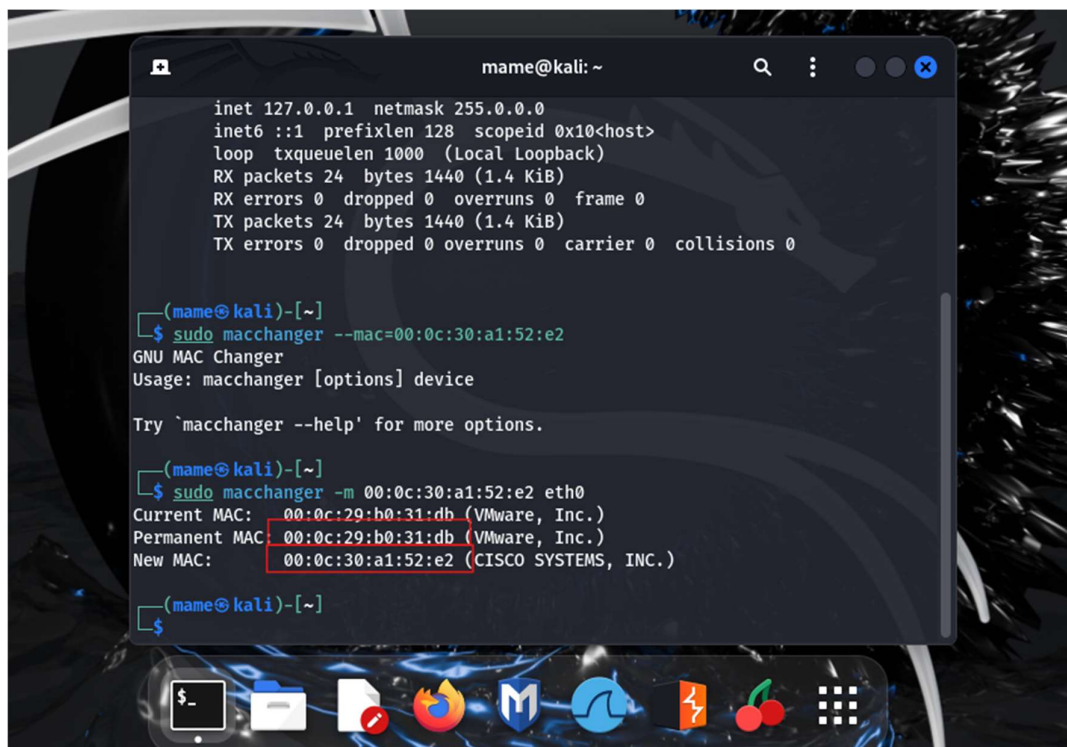


✚ L'application **macchanger** est situé dans le répertoire **/usr/bin**



```
mame@kali: ~  
macchanger  
macof  
macptopbm  
mactime  
mag  
magicrescue  
magicsort  
mailer  
mailmail3  
mailodf  
mailsnarf  
make  
make-cadir  
make-first-existing-target  
make-ssl-cert  
makeconv  
makedtx  
makeglossaries  
mkpasswd  
mkpic  
mksquashfs  
mkswap  
mktemp  
mktexfmt  
mktexlsr  
mktexmf  
mktexpk  
mktexfm  
mkvcalproba  
mm2gv  
mmcat  
mmcli  
mmdbresolve  
mmls  
mmstat  
moc  
  
(mame@kali)-[~]  
└─$ which macchanger  
/usr/bin/macchanger  
  
(mame@kali)-[~]  
└─$
```

✚ Je change l'adresse MAC du carte réseau avec la commande suivante **sudo macchanger -m nouveau adresse mac ethn0**



```
mame@kali: ~  
inet 127.0.0.1 netmask 255.0.0.0  
inet6 ::1 prefixlen 128 scopeid 0x10<host>  
loop txqueuelen 1000 (Local Loopback)  
RX packets 24 bytes 1440 (1.4 KiB)  
RX errors 0 dropped 0 overruns 0 frame 0  
TX packets 24 bytes 1440 (1.4 KiB)  
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
(mame@kali)-[~]  
└─$ sudo macchanger --mac=00:0c:30:a1:52:e2  
GNU MAC Changer  
Usage: macchanger [options] device  
  
Try `macchanger --help' for more options.  
  
(mame@kali)-[~]  
└─$ sudo macchanger -m 00:0c:30:a1:52:e2 eth0  
Current MAC: 00:0c:29:b0:31:db (VMware, Inc.)  
Permanent MAC: 00:0c:29:b0:31:db (VMware, Inc.)  
New MAC: 00:0c:30:a1:52:e2 (CISCO SYSTEMS, INC.)  
  
(mame@kali)-[~]  
└─$
```

- ✚ J'ai pris environ 8 minutes pour réussir cette étape.
- ✚ Quels sont les enjeux/dangers possibles avec une telle application ?

L'utilisation de **macchanger** pour modifier l'adresse MAC sur Kali Linux peut présenter plusieurs enjeux et dangers.

- Conflits d'adresse MAC : Si deux machines sur le même réseau ont la même adresse MAC, cela peut provoquer des conflits, rendant le réseau instable ou certaines machines inaccessibles.
- Problèmes de sécurité : Changer l'adresse MAC peut être utilisé pour contourner des contrôles d'accès réseau basés sur les adresses MAC. Cela peut aussi être utilisé pour masquer l'identité de l'attaquant dans des attaques réseau.

✚ Comment s'en protéger ?

- Utilisation d'adresses MAC légitimes : Évitez les conflits en utilisant des adresses MAC uniques.
- Surveillance et détection : Utilisez des outils comme arpswatch pour surveiller les changements d'adresses MAC.
- Contrôles d'accès basés sur les adresses MAC : Configurez des listes blanches sur vos équipements réseau.

✚ L'application **zenmap-kbx** se situe dans **/usr/bin**.



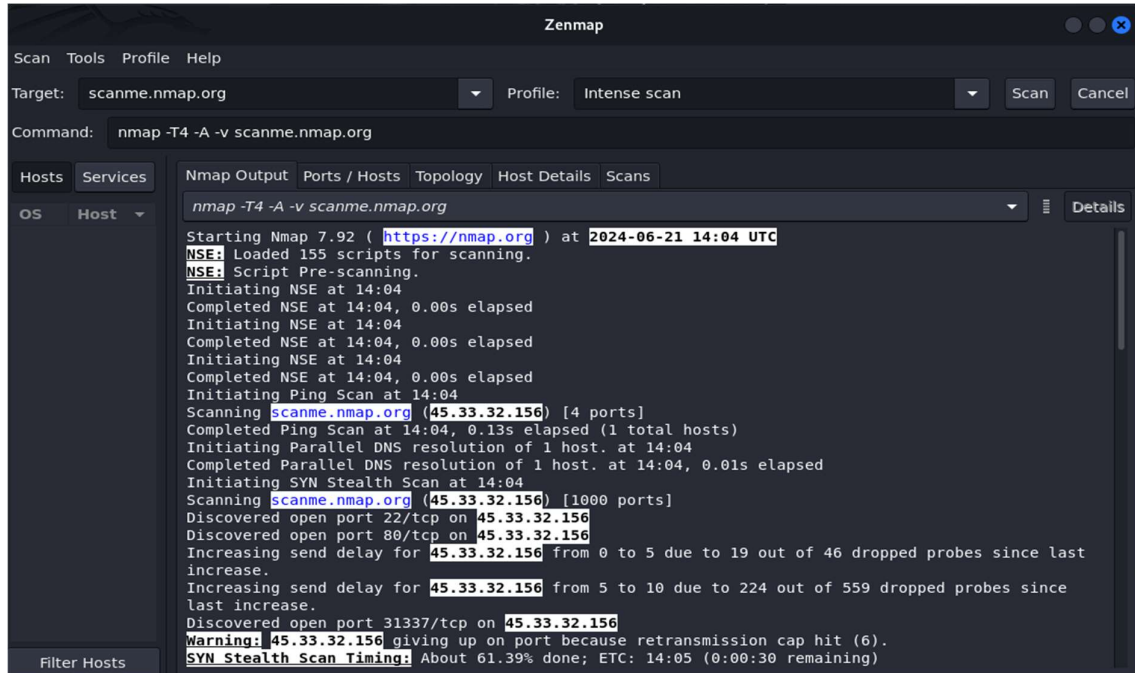
```
mame@kali: /usr/bin
zip
zipcloak
zipdetails
zipgrep
zipinfo
zipnote
zipsplit
zless
zmore
znew
zsh
zsh5
zstd
zstdcat
zstdgrep
zstdless
zstdmt

(mame@kali)-[/usr/bin]
└─$ which zenmap-kbx
/usr/bin/zenmap-kbx

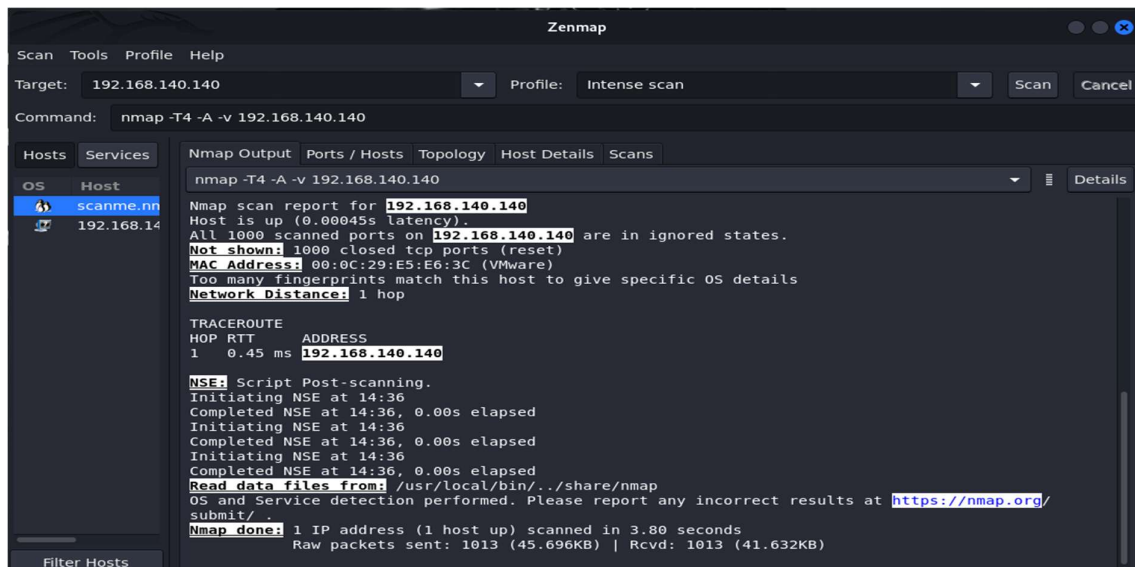
(mame@kali)-[/usr/bin]
└─$
```

🚦 **Zenmap-kbx** est une interface graphique pour Nmap, un outil puissant de scan de ports et de sécurité réseau. Expérimentons là avec le serveur scanme.nmap.org et les clients que j'ai installés avant (clients Windows/ Linux).

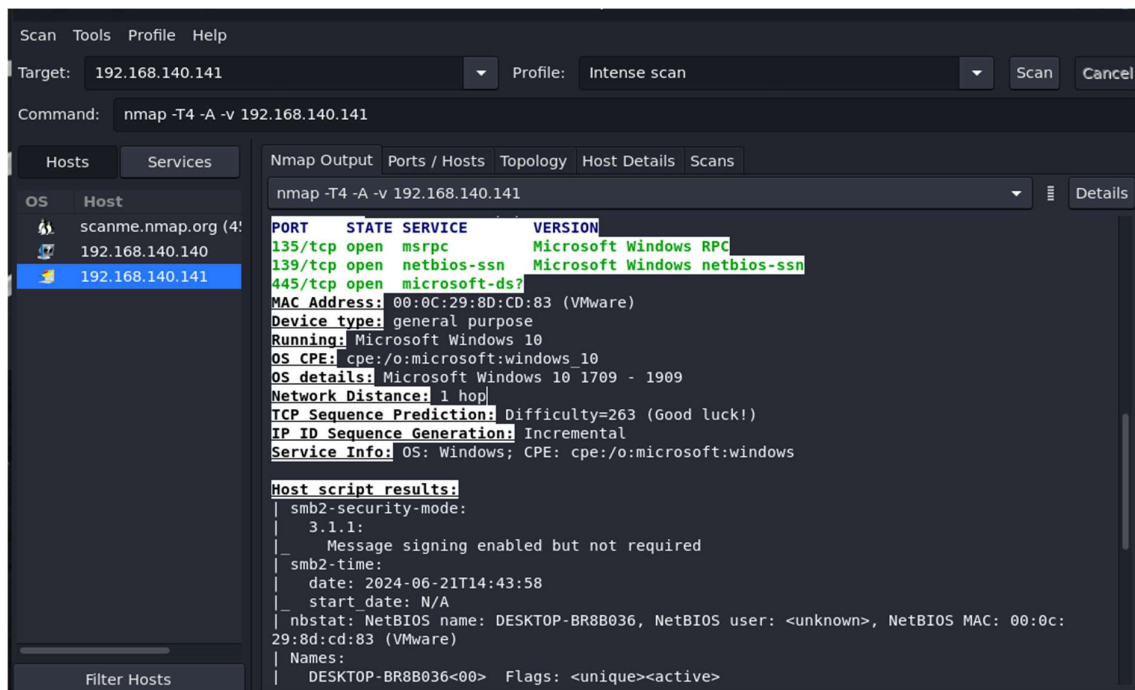
✓ Avec le serveur scanme.nmap.org



✓ Avec le client linux\Debian



✓ Avec le client Windows



🚦 Quelle application se cache derrière Zenmap-kbx ?

Zenmap-kbx est une interface graphique pour **Nmap**, qui est un outil de scan de ports et d'analyse de la sécurité réseau.

🚦 Que permet cette application ?

Nmap (Network Mapper) permet de :

- ✓ Scanner les réseaux pour découvrir les hôtes actifs.
- ✓ Identifier les services en cours d'exécution sur ces hôtes.
- ✓ Détecter les versions des logiciels et systèmes d'exploitation.
- ✓ Repérer les vulnérabilités potentielles.
- ✓ Cartographier le réseau pour une meilleure compréhension de sa structure.

🚦 Quels sont les enjeux/dangers possibles avec une telle application ?

- Utilisation malveillante : Les attaquants peuvent utiliser Nmap pour explorer les réseaux et identifier les cibles vulnérables.
- Des scans non autorisés peuvent être le prélude à des attaques plus graves, telles que des intrusions ou des exfiltrations de données.
- Charges réseau : Des scans intensifs peuvent saturer le réseau, provoquant des ralentissements et des interruptions de service.

🚧 Comment s'en protéger ?

- ✓ Contrôles d'accès et permissions :
 - Restreindre l'accès à Nmap aux administrateurs réseau et aux équipes de sécurité.
 - Utiliser des comptes avec des permissions limitées pour exécuter des scans.
- ✓ Surveillance et détection :
 - Configurer les IDS et les systèmes de gestion des événements de sécurité (SIEM) pour détecter et alerter sur les scans de ports suspects.
 - Surveiller les journaux de réseau pour des activités de scan inhabituelles.
- ✓ Politiques et formation :
 - Établir des politiques claires sur l'utilisation de Nmap et d'autres outils de sécurité réseau.
 - Former les utilisateurs autorisés sur les bonnes pratiques et les implications de l'utilisation de ces outils.
- ✓ Segmentation et filtrage réseau :
 - Mettre en œuvre la segmentation du réseau pour limiter la portée des scans.
 - Utiliser des pare-feux et des listes de contrôle d'accès (ACL) pour filtrer les tentatives de scan non autorisées.

Note de Service

Objet : Précautions à prendre avec Macchanger et Zenmap-kbx

À : Tous les employés

De : Ciss

Date :17/07/2024

Sujet : Utilisation et précautions pour Macchanger et Zenmap-kbx

Introduction

Dans notre effort pour sécuriser notre réseau informatique, nous avons évalué les applications Macchanger et Zenmap-kbx. Cette note résume leurs fonctions, les risques associés et les mesures de protection nécessaires.

1. Macchanger

- **Fonction** : Change l'adresse MAC de la carte réseau.
- **Risques** :
 - Anonymat et traçabilité : Difficile de suivre les appareils sur le réseau.
 - Contournement des contrôles d'accès : Accès non autorisé possible.
 - Activités malveillantes : Masquer l'identité ou imiter des adresses MAC autorisées.

- Mesures :
Limiter l'utilisation à l'administration réseau.
Surveiller les changements d'adresse MAC.
Interdire l'utilisation non autorisée.

2. Zenmap-kbx

- **Fonction** : Interface graphique pour Nmap, scanne les ports et analyse la sécurité réseau.
- Risques :
Utilisation malveillante : Identifier des cibles vulnérables.
Charges réseau : Scans intensifs peuvent saturer le réseau.
Détection IDS : Peut provoquer des faux positifs.
- Mesures :
Restreindre l'accès à l'administration réseau.
Configurer IDS pour détecter les scans suspects.
Former les utilisateurs autorisés.

Conclusion

Encadrer strictement l'utilisation de Macchanger et Zenmap-kbx avec des politiques et une surveillance adéquate pour protéger notre réseau.

Pour toute question, contactez le service informatique.

CISS
Informaticien
0729057247
SOS-FUTUR

Conclusion :

Ces expérimentations avec Kali Linux m'a fourni une compréhension pratique des outils de sécurité et de réseau, tel quels Macchanger et Zenmap-kbx. On a démontré comment Kali Linux peut être utilisé pour des audits de sécurité, la détection de vulnérabilités, et la gestion des réseaux. Cependant, on a aussi découvert l'importance de l'éthique et de la responsabilité lors de l'utilisation de tels outils, en insistant sur la nécessité de les utiliser dans des environnements contrôlés et avec l'autorisation appropriée.