

HACKY' NOV

WRITEUP

Réseau

Invest'PCAP

Billy Desquiens



Table des matières

1. Etude du PCAP :	1
2. Reconstitution du zip :	3
3. Bruteforce du zip :	4
4. Inspection du fichier de configuration cisco	6



1. Etude du PCAP :

On sait que l'on doit chercher des informations de connexion SSH, la première chose à tenter est de vérifier si ce protocole est utilisé :

No.	Time	Source	Destination	Protocol	Length	Info
				ssh		

L'un des moyens de transférer des informations de connexions, est par un fichier, en se tournant du côté des échanges FTP non chiffrés, on peut y voir un nom d'utilisateur : zack.

3656	1637146016.6...	192.168.1.93	192.168.1.41	FTP	65	Request: USER zack
3657	1637146016.6...	192.168.1.41	192.168.1.93	FTP	87	Response: 331 Password required for zack.
3658	1637146016.6...	192.168.1.93	192.168.1.41	FTP	61	Request: P
3659	1637146016.6...	192.168.1.41	192.168.1.93	FTP	80	Response: 230 User zack logged in.
3660	1637146016.6...	192.168.1.93	192.168.1.41	FTP	60	Request: S
3661	1637146016.6...	192.168.1.41	192.168.1.93	FTP	73	Response: 215 UNIX type: L
3662	1637146016.6...	192.168.1.93	192.168.1.41	FTP	60	Request: FEAT
3663	1637146016.6...	192.168.1.41	192.168.1.93	FTP	106	Response: 211-Extensions supported:
3664	1637146016.6...	192.168.1.93	192.168.1.41	TCP	54	41250 → 21 [ACK] Seq=31 Ack=169 Win=88064 Len=0
3665	1637146016.8...	192.168.1.93	192.168.1.41	FTP	61	Request: CWD /
3666	1637146016.8...	192.168.1.41	192.168.1.93	FTP	116	Response: 250 CWD command successful. "/D:/ftp/" is current directory.
3667	1637146016.8...	192.168.1.93	192.168.1.41	TCP	54	41250 → 21 [ACK] Seq=38 Ack=231 Win=88064 Len=0
3668	1637146017.0...	192.168.1.93	192.168.1.41	FTP	60	Request: STAT
3669	1637146017.0...	192.168.1.41	192.168.1.93	FTP	91	Response: 500 'STAT': command not understood.
3670	1637146017.0...	192.168.1.93	192.168.1.41	TCP	54	41250 → 21 [ACK] Seq=44 Ack=268 Win=88064 Len=0
3671	1637146017.0...	192.168.1.93	192.168.1.41	FTP	59	Request: PWD
3672	1637146017.0...	192.168.1.41	192.168.1.93	FTP	92	Response: 257 "/D:/ftp/" is current directory.
3673	1637146017.0...	192.168.1.93	192.168.1.41	FTP	59	Request: PWD
3674	1637146017.0...	192.168.1.41	192.168.1.93	FTP	92	Response: 257 "/D:/ftp/" is current directory.
3675	1637146017.0...	192.168.1.93	192.168.1.41	FTP	59	Request: PWD
3676	1637146017.0...	192.168.1.90	157.240.21.16	TLSv1.2	86	Application Data
3677	1637146017.0...	192.168.1.41	192.168.1.93	FTP	92	Response: 257 "/D:/ftp/" is current directory.
3678	1637146017.0...	192.168.1.93	192.168.1.41	FTP	60	Request: PASV
3679	1637146017.0...	192.168.1.41	192.168.1.93	FTP	105	Response: 227 Entering Passive Mode (192,168,1,41,253,172).

Si on suit les transferts de fichiers via ftp, on trouve ces trames:

3684	1637146017.0...	192.168.1.41	192.168.1.93	FTP-DA...	119	FTP Data: 65 bytes (PASV) (ST -a myzip.zip)
4123	1637146026.4...	192.168.1.41	192.168.1.93	FTP-DA...	119	FTP Data: 65 bytes (PASV) (ST -a myzip.zip)
4146	1637146026.5...	192.168.1.41	192.168.1.93	FTP-DA...	1514	FTP Data: 1460 bytes (PASV) (RETR myzip.zip)
4147	1637146026.5...	192.168.1.41	192.168.1.93	FTP-DA...	772	FTP Data: 718 bytes (PASV) (RETR myzip.zip)

On peut voir qu'un fichier nommé myzip.zip a été transféré, on peut donc chercher à savoir ce qu'il contenait. On va donc suivre l'échange avec Wireshark en faisant un clic droit sur une des trames, Suivre et Flux TCP.

On obtient alors une chaîne de caractère qui ne font pas de sens dont voici un extrait:

```
PK.... ....TqSg+B.....*
.....ssh/id_rsaUT ..%.aS..aux....."^.:....cO.5..R..v.
..?./-...R.R|.B....C.Hn..V.Q+...}Ma"...S...15j.....e...ngS.|p..Qd.....
w.l.....:>O..V0...Z...L .....>...c...?n...U...@.....]^
```



Cependant on peut y voir `ssh/id_rsa`, ce qui évoque une clé SSH.

```
.ssh/id_rsa
```

2. Reconstitution du zip :

On va chercher à reconstituer le fichier zip échangé à l'aide de la trame à l'aide de la version en ligne de Cyberchef (<https://gchq.github.io/CyberChef/>).

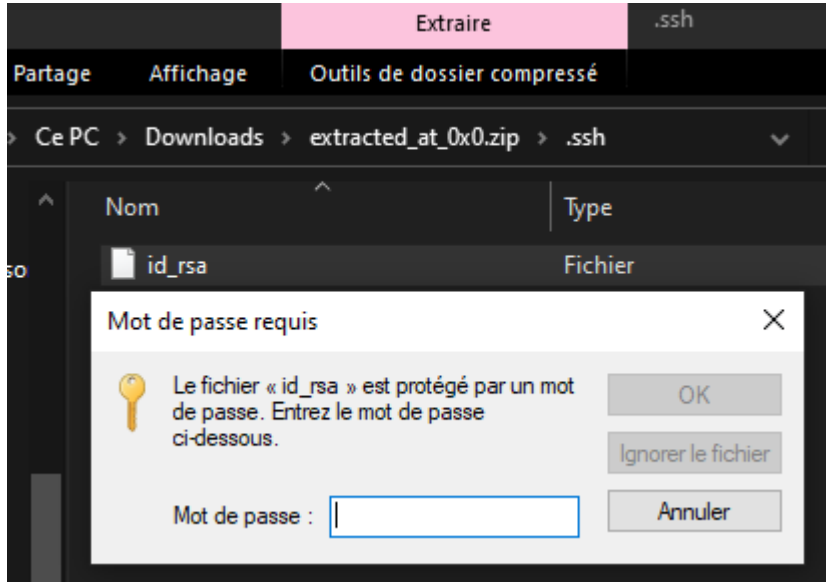
Je demande au site de reconstituer des fichiers à partir du format HexDump, la partie Recipe ressemble donc à ça:

Sur wireshark je mets donc l'échange au format hexdump et je copie le contenu pour le coller dans le champ input. On obtient alors ce résultat:

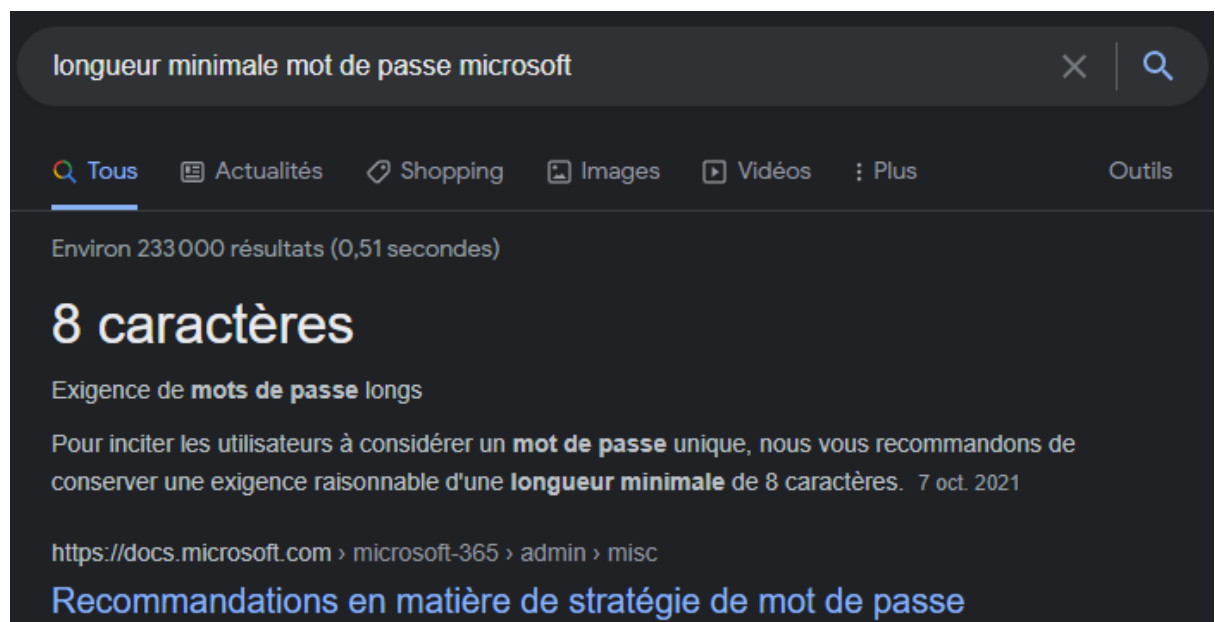
Deux fichiers zip ont été trouvés, cependant un des deux n'est pas exploitable car corrompu, on se penche alors sur le deuxième (qui est le plus volumineux des deux).

3. Bruteforce du zip :

On peut l'ouvrir sur un système Windows et on y voit un dossier .ssh, qui comporte lui-même la fameuse clé id_rsa. Cependant quand on essaye de l'ouvrir, on nous demande un mot de passe:



Si on se rappelle de la description du challenge, on sait que les utilisateurs se basent sur la longueur minimale conseillée par Microsoft, après une petite recherche Google on sait vite que le mot de passe fait 8 caractères.



Quand on sait que les utilisateurs ne respectent pas les bonnes pratiques de mot de passe, que ceux-ci font huit caractères et que le nom d'utilisateur prend quatre caractères, on peut imaginer que le mot de passe doit suivre le pattern suivant: zack%%%, où le % est un chiffre.

On génère donc un dictionnaire à l'aide de crunch comportant huit caractères et respectant ce pattern:

```
(tialoc@ptbd)-[~]
$ crunch 8 8 -t zack%% -o persowordlist
```

On essaye ensuite, d'effectuer une attaque par dictionnaire sur notre fichier zip avec ce dictionnaire:

```
(tialoc@ptbd)-[~]
$ fcrackzip -v -D -u -p ./persowordlist Téléchargements/extracted at 0x0.zip
found file '.ssh/id_rsa', (size cp/uc 1990/ 2602, flags 9, chk 540b)

PASSWORD FOUND!!!!: pw = zack3003
```

On obtient le mot de passe.

Attention: ici un dictionnaire restreint a été généré, on peut obtenir le même résultat en générant un dictionnaire comportant tous les mots de passe à huit caractères cependant cela prendrait beaucoup plus de temps.

4. Inspection du fichier de configuration cisco

On peut maintenant extraire les données du zip, et utiliser la clé ssh pour se connecter au serveur ssh:

```
ssh -i id_rsa -p port zack@adress_ip_serveur
```

```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Dec 15 14:22:09 2021 from 192.168.1.95
$ ls
cisco.conf
$
```

Sur le serveur, on peut voir un fichier, en lisant son contenu, on s'aperçoit que c'est un fichier de configuration cisco.

Un premier flag attire l'attention:

```
23 radius server AUTH-SERV
24 address ipv4 192.168.1.251 auth-port 1812 acct-port 1813
25 key H4CKYNov{Try4g41n}
26
```

Il est erroné, ce n'est pas le bon format.

C'est en allant tout en bas du fichier de configuration que nous trouvons un utilisateur zack avec un mot de passe type 7.

```
297 !
298 username zack privilege 1 password 7 08096D6D22202B38241022543E38303A633B25470B11531C
299 !
```

En mettant ce mot de passe dans un décodeur nous obtenons notre flag:

Enter Your Encrypted Password Below:

Encrypted Password: 08096D6D22202B38241022543E38303A633B25470B11531C

Submit

Decrypted Password: HACKYNOV{N0tstr0ng4lg0}