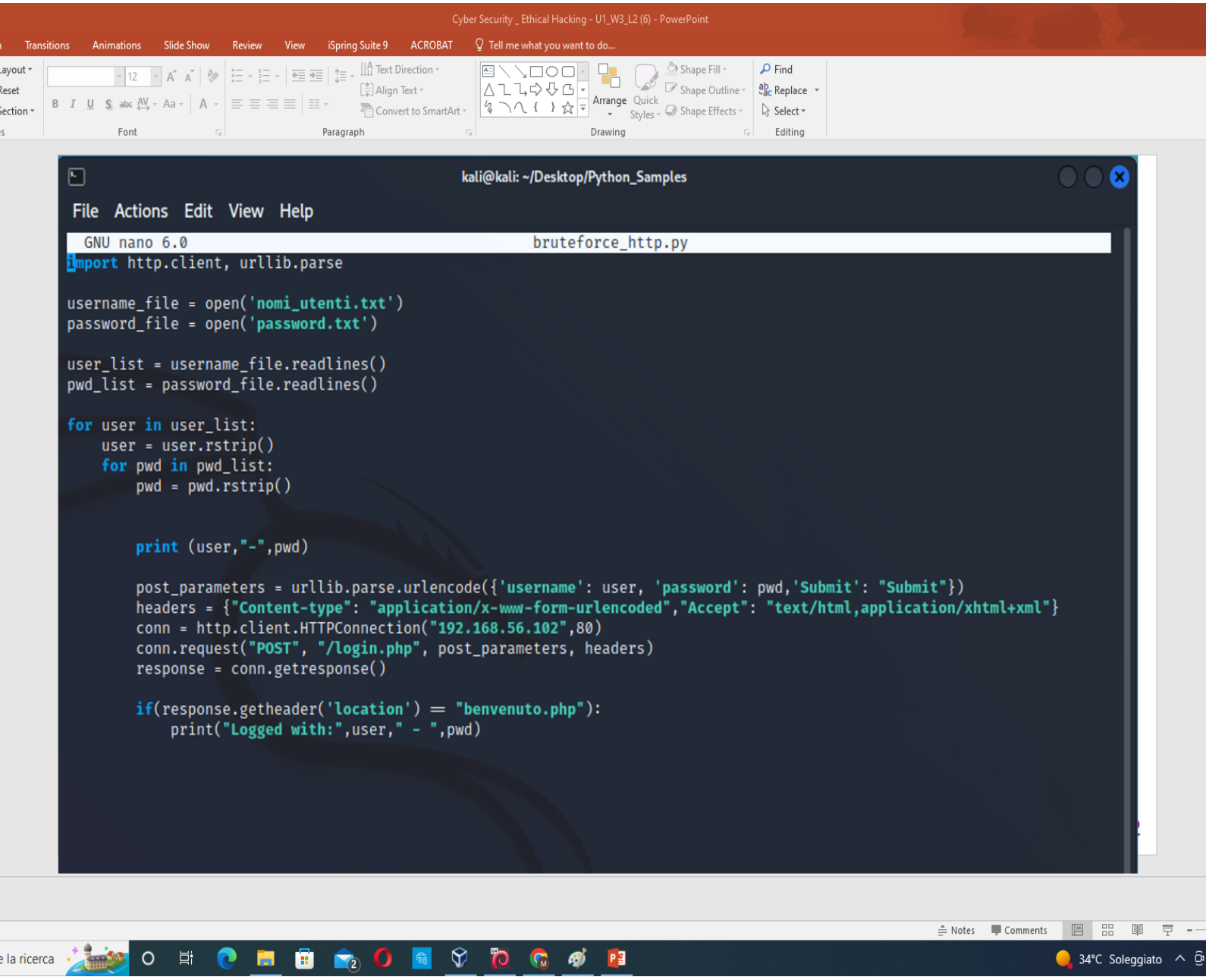
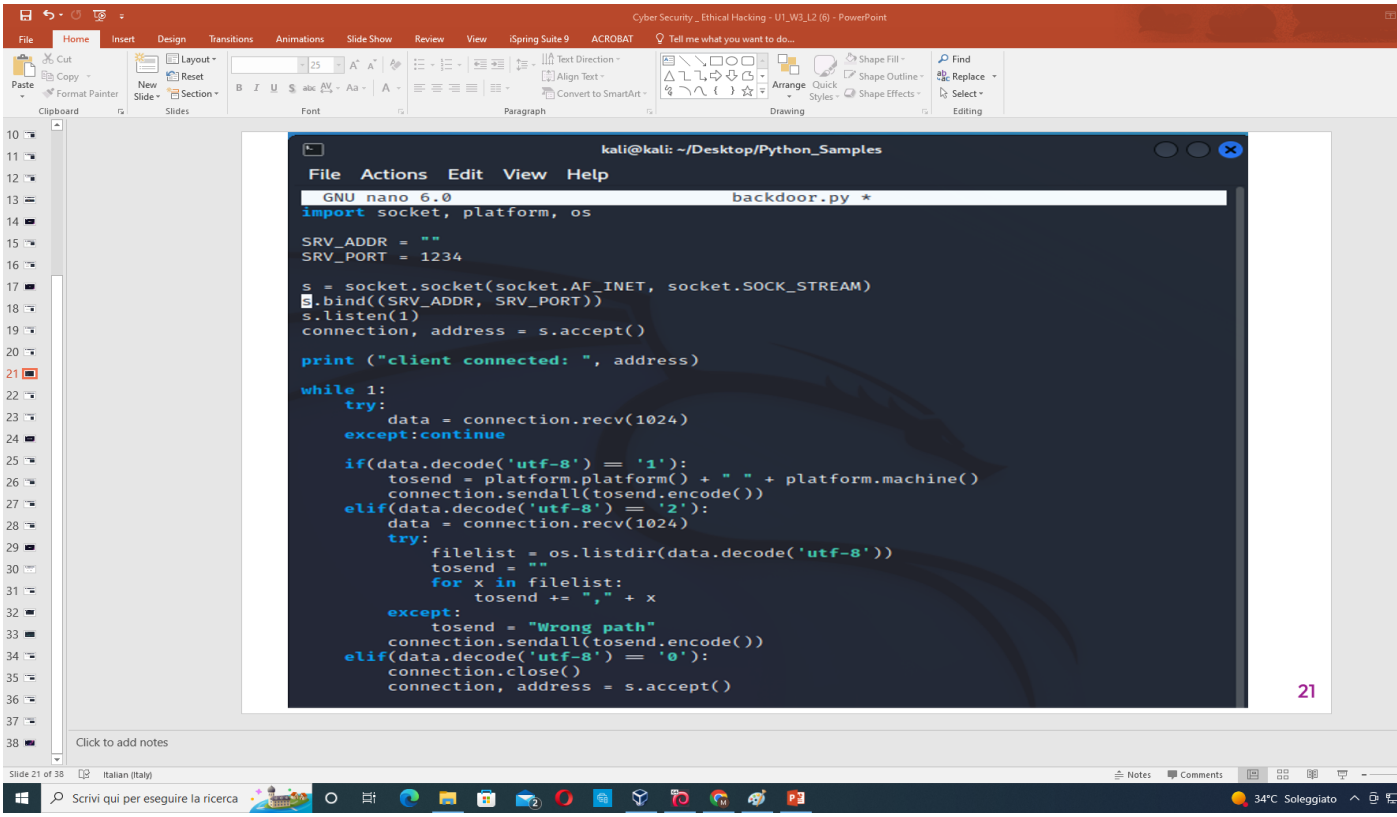


Il Brute Force Attack è un metodo utilizzato dai cybercriminali per craccare le password degli account e scoprire credenziali di accesso.
Ora andiamo a vedere nello specifico il codice utilizzato.

Import; Serve per importare funzioni esterne dentro il programma python sono fatti da creatori esterni. I moduli esterni non possono essere scritti in maniera differente da

come sono stati strutturati in questo caso il modulo è (http.client,urllib.parse)
Username/password_file: Sono le variabili che richiamano la funzione open. La funzione open serve per aprire il file txt all'interno delle variabili.
User/pass_list: Sono le variabili che permettono di richiamare la lettura dei file con il comando readlines().
For:E' un ciclo di controllo che ci permette di eseguire un operazione per un certo di numero di volte.
In questo caso il ciclo cercherà user e password nelle liste txt.
print: Print è quello che vedrà l'utente a schermo . in output nome e password.
da post parameters a response:Il programma proverà gli user e le password sul headers scritto tra le graffe dalla porta 80 e dall' indirizzo ip 192.168.56.102.Ci sarà una richiesta di connessione al sito con i parametri user e pass,subito una responso di connessione, Se è positivo arriverà il messaggio di "logged with" con user e password printati a schermo.
if: E' una funzione condizionale di python il comando esegue le istruzioni assegnate dentro if.



Import:Serve per importare funzioni esterne dentro il programma python sono fatti da creatori esterni. I moduli esterni non possono essere scritti in maniera differente da come sono stati strutturati(socket, platform,os).

Srv_addr/port: Sono i nomi delle variabili. uno per l'indirizzo ip uno per la porta da usare.

S= socket.socket:E' la parte del modulo da scrivere obbligatoriamente, è la funzione che ci fa creare il socket. Socket.AF_INET(lavora con indirizzo ip)socket.SOCK_STREAM(lavora con il protocollo tcp).

S.bind:E' sempre obbligatorio scriverlo collega il socket all'indirizzo ip questa operazione si chiama binding.

S.listen:Decide quante connessioni deve andare a gestire in questo caso (1).

Connection, address = s.accept():Accetta la connessione in base all'indirizzo ip e la porta usata qualora vada a buon fine

Print:output a schermo per l'utente dove si vedrà client connected, e l'indirizzo ip dove sono connesso.
WHILE 1: Serve a eseguire un istruzione o più istruzioni finche una certa condizione viene verificata. il costrutto while definisce dei cicli (1) sta a significare che è infinito finche la condizione è vera.
TRY:Il costrutto try ci da la possibilità di verificare e identificare gli errori dentro il programma. In questo caso try va a ricevere i pacchetti dati di connessione, se c'è un errore interviene l'except che dice di continuare a provare.
IF: If è un costrutto condizionale in questo programma ci da la possibilità di decodificare i dati se "utf-8" ==1 si andrà ad eseguire la stringa di codice scritta sotto il primo if altrimenti si passerà avanti all'elif.
ELIF:E' un abbreviazione di IF in questo caso se Utf-8 == 2 verrà presa in considerazione la stringa sotto compreso il try con il ciclo for all'interno. (Del ciclo FOR ne ho parlato nella prima relazione sull' attacco bruteforce).
EXCEPT: L'eccenzione messa in questo caso sarà se utf-8 != 1 utf!=2 a questo punto il programma farà uscire in output "wrong path" Output (la scherma che vede l'utente).
ELIF: L'ultimo elif determina l'uscita dal programma se utf-8 ==0. Il programma si chiuderà.
!=: DIVERSO DA.
==: UGUALE.