Relazione di Progetto: Configurazione dei Servizi di Rete e Cracking dell'Autenticazione

Strumento Focus: THC-Hydra
Ambiente: Kali Linux (Attaccante & Target), Rete Locale

---

1. Introduzione

L'obiettivo di questa esercitazione è stato consolidare la conoscenza dei servizi di rete attraverso la loro configurazione manuale e il successivo audit di sicurezza utilizzando lo strumento di password cracking **Hydra**. L'esercizio si è svolto in due fasi: prendere di mira un servizio SSH e configurare/attaccare un servizio FTP. Una componente fondamentale del laboratorio è stata la risoluzione dei problemi legati ai meccanismi difensivi intrinseci nei moderni servizi di rete.

---

2. Preparazione: Ottimizzazione delle Wordlist

Per ottimizzare il processo di cracking, come suggerito dall'istruttore, ho creato wordlist più piccole e mirate derivate dal repository SecLists.

Sfida: I percorsi dei file nella documentazione dell'assegnazione differivano dall'attuale struttura delle directory di Kali Linux.
Soluzione: Ho individuato il percorso corretto (/Common-Credentials/) e ho utilizzato grep per filtrare le voci rilevanti contenenti "test".
Comando Utilizzato:

```
cat /usr/share/seclists/Passwords/Common-Credentials/xato-net-10-million-passwords-1000000.txt | grep test > xato-passwords.txt
```

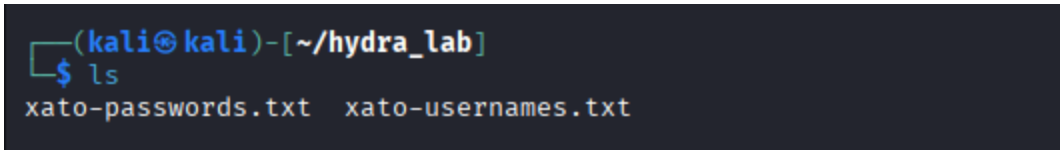

Figura 1: Verifica delle wordlist ottimizzate xato-usernames.txt e xato-passwords.txt create nella directory di lavoro.

---

3. Fase 1: Configurazione e Attacco SSH

3.1 Configurazione del Target

Ho creato uno specifico utente target test_user con una password nota (testpass) sulla macchina Kali (192.168.100.10) per fungere da vittima.

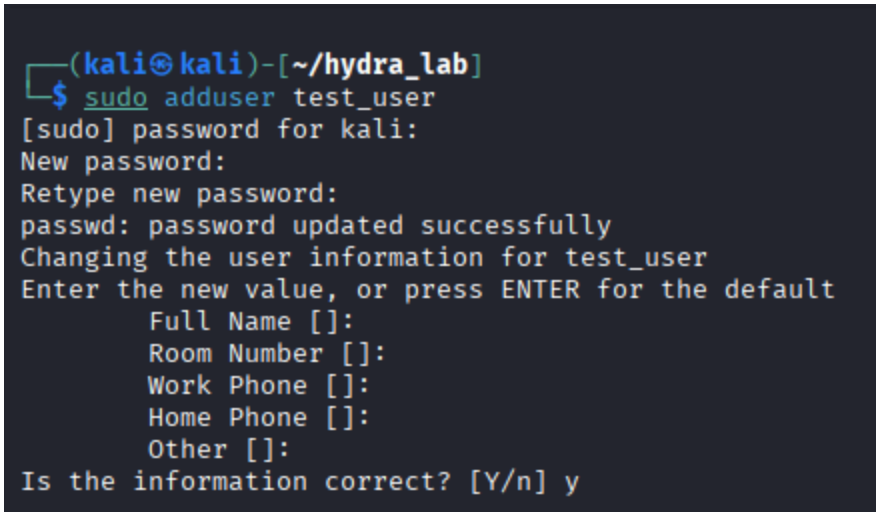

Figura 2: Creazione dell'account test_user utilizzando il comando adduser.

Ho inoltre verificato il file di configurazione SSH per comprendere le impostazioni predefinite, assicurandomi specificamente che la porta di default fosse la 22.

Figura 3: Revisione di /etc/ssh/sshd_config. Non sono state apportate modifiche per preservare il comportamento predefinito per il test iniziale.

3.2 La Sfida del Cracking (Risoluzione Problemi)

I tentativi iniziali di craccare il servizio SSH utilizzando la lista completa di nomi utente sono falliti. Hydra ha riportato errori di connessione quasi immediatamente.

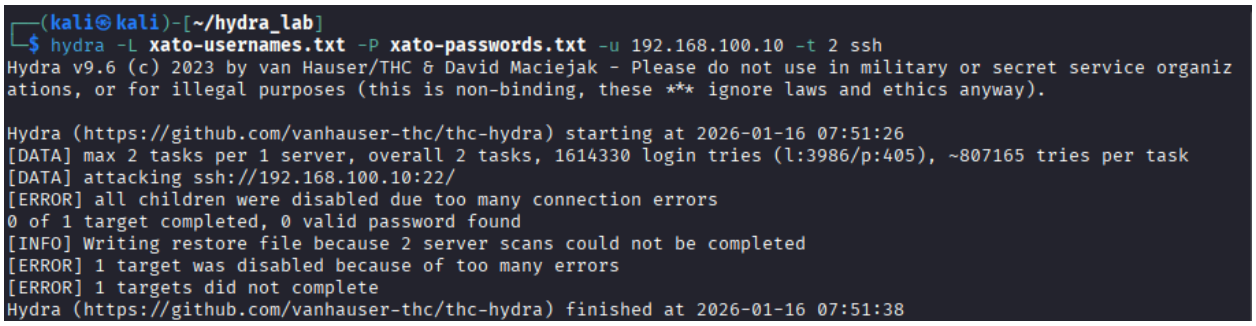

Figura 4: Hydra fallisce con l'errore "all children were disabled due to many connection errors".

Analisi Tecnica del Fallimento:
Ho analizzato i log di sistema (/var/log/auth.log) per diagnosticare il problema. I log hanno rivelato che il demone SSH (sshd) stava rifiutando attivamente le connessioni a causa di un numero eccessivo di fallimenti di autenticazione.

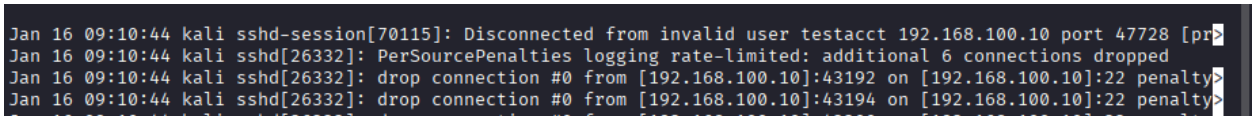

Figura 5: Log di sistema che mostrano violazioni di PerSourcePenalties e MaxAuthTries.

- **Osservazione:** La voce di log PerSourcePenalties logging rate-limited indica che il server SSH ha bloccato interamente l'indirizzo IP di origine dopo aver rilevato una serie di tentativi di accesso falliti per utenti non validi (es. "testing", "admin").
- **Perché le modifiche non hanno funzionato:**
  - **Modifica Tentata (-u):** Ho provato il flag -u per ciclare i nomi utente ed evitare il blocco di account specifici. Questo tentativo è fallito perché il ban era basato sull'**IP**, non sull'utente.
  - **Modifica Tentata (-W):** Ho provato ad aggiungere un tempo di attesa (-W 1). Questo è fallito perché la difesa del server si basa su una **soglia totale di fallimenti**, non su un limite di velocità (rate-limit).

3.3 La Soluzione: Attacco Mirato

Per aggirare il "rumore" generato dagli utenti non validi che innescava il ban dell'IP, ho modificato l'attacco riducendo il numero di entries nel file con gli username (in alternativa anche -l user_specifico avrebbe sortito un effetto ancora piú veloce e mirato). Ciò ha impedito la generazione di log di fallimento e ha permesso all'attacco di procedere.

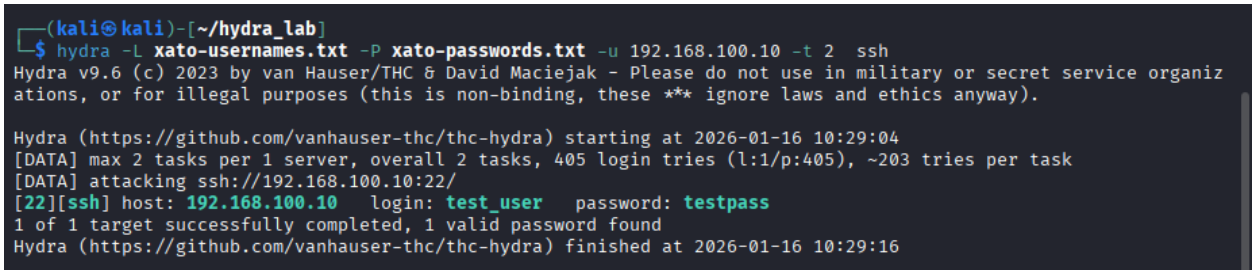

Figura 6: Cracking SSH riuscito. Utilizzando un file con meno entries per mirare a un utente specifico, è stato evitato il ban dell'IP.

---

4. Fase 2: Configurazione e Attacco del Servizio FTP

Seguendo le istruzioni dell'assegnazione, ho installato e configurato il servizio vsftpd per testare un protocollo diverso.

Comandi di Setup:

```
sudo apt install vsftpd
sudo service vsftpd start
```

Poiché l'FTP è generalmente meno rigoroso riguardo al rate-limiting e ai controlli di pre-autenticazione rispetto all'SSH, ho utilizzato lo stesso approccio con wordlist mirata.

Comando:

```
hydra -L xato-usernames.txt -P xato-passwords.txt 192.168.100.10 -t 2 ftp
```

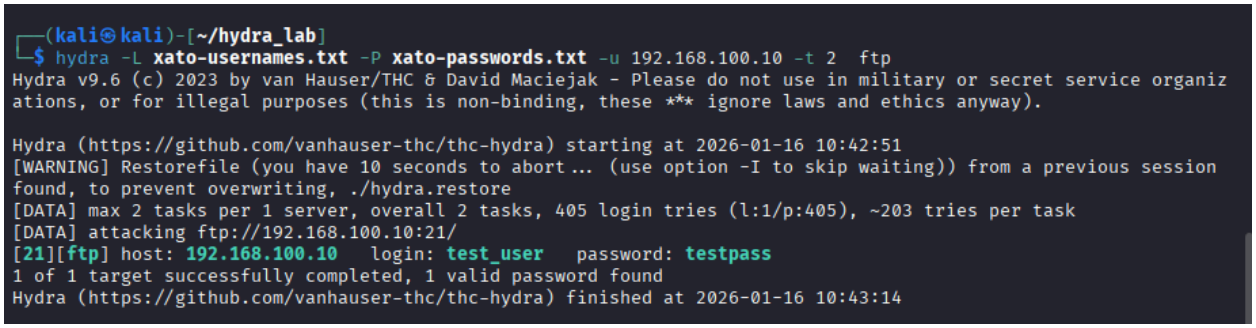

Figura 7: Cracking FTP riuscito. Hydra ha identificato la password testpass quasi immediatamente.

Dopo il test, ho assicurato che il servizio fosse arrestato per mantenere la sicurezza del laboratorio.

---

5. Conclusione

Questo laboratorio ha dimostrato che un attacco brute-force di successo non riguarda solo l'esecuzione di uno strumento, ma la comprensione delle difese del bersaglio.

Punti Chiave:

1. **Difese del Servizio:** Le configurazioni moderne di SSH (OpenSSH) includono PerSourcePenalties che bannano gli IP che generano "rumore" (nomi utente non validi), rendendo inefficaci gli attacchi a dizionario standard senza l'uso di proxy o tecniche di evasione.
2. **Sintassi di Hydra:** Distinguere tra -L (lista utenti) e -l (utente singolo) è critico per gli attacchi mirati.
3. **Risoluzione Problemi:** L'analisi dei log del server è l'unico modo per diagnosticare accuratamente perché un attacco brute-force sta fallendo (ad esempio, distinguendo tra una caduta di rete e un ban di sicurezza).