TryHackMe Advent of Cyber 2025

Day 9 Challenge Report

Password Cracking & File Encryption

1. Executive Summary

This report documents the completion of Day 9 of the TryHackMe Advent of Cyber 2025 event. The challenge focused on password cracking techniques for encrypted files using dictionary attacks. Successfully cracked passwords for both PDF and ZIP archives using pdfcrack and John the Ripper, revealing hidden flags. Additionally explored detection methodologies for identifying password cracking activity in enterprise environments.

2. Challenge Overview

Objective: Crack passwords protecting encrypted PDF and ZIP files to recover hidden flags using dictionary attacks.

Tools Used: pdfcrack, John the Ripper (john), zip2john, rockyou.txt wordlist

3. Password-Based Encryption Fundamentals

Password-based encryption protects file confidentiality through cryptographic algorithms, but security depends entirely on password strength.

3.1 Key Principles

- **Password Strength:** Short/common passwords easily guessed; long random passwords harder to break
- **Algorithm Variation:** Different formats (PDF, ZIP) use different key derivation methods
- **Legacy Weaknesses:** Consumer tools often support weak encryption modes (especially older ZIP)
- **Offline Attacks:** Encryption prevents reading but doesn't stop offline password guessing

3.2 Encryption vs Password Protection

Encryption makes content unreadable without the correct password. If the password is weak, attackers simply try likely passwords until one succeeds - no need to break the encryption itself.

4. Password Attack Methodologies

4.1 Dictionary Attacks

Dictionary attacks use predefined wordlists to test potential passwords systematically.

Common Wordlist Contents:

- Leaked passwords from previous breaches
- Common substitutions (password123, P@ssw0rd)
- Predictable combinations of names and dates
- Frequently used patterns

Effectiveness: Fast and highly effective due to widespread use of weak passwords.

4.2 Brute-Force & Mask Attacks

Brute-Force: Systematically tries every possible character combination. Time grows exponentially with password length.

Mask Attacks: Limit guesses to specific formats to reduce search space.

Mask Example:

```
?l?l?l?d?d = three lowercase letters + two digits
```

Mask attacks balance speed and thoroughness, especially when password structure is known.

4.3 Attacker Strategy

1. **Start with wordlists** (rockyou.txt, common-passwords.txt) - fast wins
2. **Targeted wordlists** (company names, project names, target-specific data)
3. **Mask/incremental attacks** on short passwords
4. **GPU acceleration** dramatically speeds up certain algorithms

5. Practical Exploitation

5.1 File Type Identification

Before cracking, confirm file type using the file command or hex viewer to select appropriate tools.

5.2 Tool Selection

- **PDF:** pdfcrack, john (via pdf2john)
- **ZIP:** fcrackzip, john (via zip2john)
- **General:** john (flexible), hashcat (GPU acceleration)

5.3 PDF Password Cracking

Tool: pdfcrack with rockyou.txt wordlist

```
pdfcrack -f flag.pdf -w /usr/share/wordlists/rockyou.txt
```

Results:

Password Recovered: naughtylist

Opened PDF with recovered password, revealing hidden flag:

Flag: THM{Cr4ck1ng_PDFs_1s_34$y}

5.4 ZIP Password Cracking

Tool: John the Ripper with zip2john converter

Step 1: Hash Extraction

Convert ZIP to John-compatible hash format:

```
zip2john flag.zip > ziphash.txt
```

Step 2: Dictionary Attack

Run John with rockyou.txt wordlist:

```
john --wordlist=/usr/share/wordlists/rockyou.txt ziphash.txt
```

Results:

Password Recovered: winter4ever

Extracted ZIP contents with recovered password, revealing flag:

Flag: THM{Cr4ck1n6_z1p$_1s_34$yyyy}

6. Detection & Defensive Considerations

6.1 Detection Challenges

Offline cracking doesn't hit login services, so traditional failed login monitoring is ineffective. Detection must focus on endpoint activity.

6.2 Key Detection Indicators

Process Creation:

- **Binaries:** john, hashcat, fcrackzip, pdfcrack, zip2john, pdf2john.pl, 7z, qpdf
- **Command-line:** --wordlist, -w, --rules, --mask, -a 3, -m, rockyou.txt, SecLists
- **State files:** ~/.john/john.pot, .hashcat/hashcat.potfile, john.rec

GPU & Resource Artifacts:

- nvidia-smi showing long-running hashcat/john processes
- High, steady GPU utilization and power draw
- Libraries: nvcuda.dll, OpenCL.dll, libcuda.so, amdocl64.dll

Network Indicators:

- Downloads of large text files (rockyou.txt)
- Git clones of wordlist repositories
- Package installs: apt install john hashcat

6.3 Detection Rules

Sysmon (Windows):

```
(ProcessName="C:\\Program Files\\john\\john.exe" OR
ProcessName="C:\\Tools\\hashcat\\hashcat.exe" OR  CommandLine="*pdf2john.pl*" OR
CommandLine="*zip2john*")
```

Linux Audit Rules:

```
auditctl -w /usr/share/wordlists/rockyou.txt -p r -k wordlists_read auditctl -a
always,exit -F arch=b64 -S execve -F exe=/usr/bin/john -k crack_exec auditctl -a
always,exit -F arch=b64 -S execve -F exe=/usr/bin/hashcat -k crack_exec
```

6.4 Response Playbook

- **Isolate:** Host if malicious activity detected; tag/suppress if lab
- **Capture:** Process list, memory dump, nvidia-smi output, open files
- **Preserve:** Working directory, wordlists, hash files, shell history
- **Review:** Which files decrypted; search for follow-on access/exfiltration
- **Identify:** Origin and intent; escalate if unauthorized
- **Remediate:** Rotate keys/passwords, enforce MFA
- **Educate:** Place tools in approved sandboxes

7. Key Skills Developed

- Password-based encryption fundamentals
- Dictionary attack methodology
- PDF cracking with pdfcrack
- ZIP cracking with John the Ripper
- Hash extraction with zip2john
- Wordlist-based attacks with rockyou.txt
- Detection strategies for cracking activity
- Incident response for password cracking events

8. Conclusion

Day 9 of the TryHackMe Advent of Cyber 2025 provided comprehensive training in password cracking techniques and detection methodologies. Successfully cracked passwords for both PDF and ZIP files using dictionary attacks, demonstrating the vulnerability of weak passwords even with strong encryption algorithms.

The challenge emphasized that encryption strength is only as good as password complexity. Additionally covered critical defensive aspects including detection rules, monitoring strategies, and incident response procedures for identifying and responding to password cracking activity in enterprise environments.

Challenge Status: **COMPLETED ✓**