Pentest

- Password Cracking — ZIP & PDF

Password Cracking — ZIP & PDF

> **Nur gegen eigene Dateien oder mit schriftlicher Erlaubnis verwenden.** Knacken fremder Archive ist in DE nach §202c StGB strafbar.

Setup (BlackArch / Arch)

```
sudo pacman -S zip john hashcat fcrackzip pdfcrack qpdf wordlists
```

Wordlist-Standard: `/usr/share/wordlists/seclists/Passwords/Leaked-Databases/rockyou.txt`

Workflow

```
[file.zip | file.pdf]
        |
        ▼
   zip2john / pdf2john   →   hash-Datei
        |
        ▼
   john (CPU)  —oder—  hashcat (GPU)
        |
        ▼
    Klartext-Passwort
```

ZIP cracken

1. Hash extrahieren

```
zip2john secret.zip > secret.hash
```

2a. John (CPU)

```
john --wordlist=/usr/share/wordlists/seclists/Passwords/Leaked-Databases/rockyou.txt
secret.hash
john --show secret.hash
```

2b. Hashcat (GPU)

Hashcat will nur den Hash-Teil — Filename-Prefix mit `cut` entfernen:

```
cut -d: -f2- secret.hash > secret.hccap
```

Mode	Format
`17200`	PKZIP Deflated (komprimiert)
`17210`	PKZIP Stored (unkomprimiert)
`17220`	PKZIP (mehrere komprimierte Dateien)
`13600`	WinZip AES

```
hashcat -m 17210 -a 0 secret.hccap /usr/share/wordlists/seclists/Passwords/Leaked-
Databases/rockyou.txt
hashcat -m 17210 secret.hccap --show
```

2c. fcrackzip

```
fcrackzip -u -D -p ./rockyou.txt secret.zip    # Dictionary
fcrackzip -u -c a -l 1-6 secret.zip            # Brute-Force, 1–6 Zeichen lowercase
```

> **Hinweis:** Aktuelle fcrackzip-Version hat Buffer-Overflow bei langen Wordlist-Pfaden. Wordlist immer lokal kopieren.

PDF cracken

1. Hash extrahieren

```
pdf2john locked.pdf > locked.hash
```

2a. John

```
john --wordlist=/usr/share/wordlists/seclists/Passwords/Leaked-Databases/rockyou.txt
locked.hash
john --show --format=PDF locked.hash
```

2b. Hashcat

Mode	PDF-Version	Verschlüsselung
`10400`	PDF 1.1 – 1.3	RC4 40 bit
`10500`	PDF 1.4 – 1.6	RC4 128 bit
`10600`	PDF 1.7 Level 3	AES 128 bit
`10700`	PDF 1.7 Level 8	AES 256 bit

```
cut -d: -f2- locked.hash > locked.hccap
hashcat -m 10700 -a 0 locked.hccap /usr/share/wordlists/seclists/Passwords/Leaked-
Databases/rockyou.txt
```

2c. pdfcrack

```
pdfcrack -f locked.pdf -w wordlist.txt
pdfcrack -f locked.pdf -n 4 -m 8 -c abcdefghijklmnopqrstuvwxyz
```

Attack Modes (Hashcat Cheatsheet)

Mode	Bedeutung	Beispiel
`-a 0`	Dictionary	`hashcat -m 17210 -a 0 h rockyou.txt`
`-a 1`	Combinator	`hashcat -m 17210 -a 1 h dict1 dict2`
`-a 3`	Brute-Force / Mask	`hashcat -m 17210 -a 3 h '?l?l?l?d?d'`
`-a 6`	Hybrid Wordlist+Mask	`hashcat -m 17210 -a 6 h dict.txt '?d?d'`
`-a 9`	Association	für targeted Hints

Mask-Charsets

Charset	Bedeutung
`?l`	a-z
`?u`	A-Z
`?d`	0-9
`?s`	Symbole
`?a`	`?l?u?d?s` (alles)
`?h`	0-9 a-f (hex lower)
`?H`	0-9 A-F (hex upper)

Eigene Charsets

```
hashcat -1 ?l?u?d -m 17210 -a 3 h '?1?1?1?1?1?1?1?1'
```

Inkrementell (unbekannte Länge)

```
hashcat -m 17210 -a 3 h '?a?a?a?a?a?a?a?a' --increment --increment-min=4 --increment-max=10
```

GPU vs. CPU (gemessen auf RX 7900 XTX)

Hash-Typ	Speed	Brute-Force 8x lowercase (26^8)
PKZIP (17210)	33,8 GH/s	~6 Sekunden
PDF AES-256 (10700)	17 kH/s	~388 Jahre

PDFs mit AES-256 sind ~2 Millionen mal langsamer zu cracken als PKZIP — durch bewusstes Key-Stretching (SHA-256 + Iterationen) im PDF 1.7 Standard.

Best Practices (Pentest-Workflow)

1. **Quick wins zuerst** — Dictionary mit rockyou.txt, dann SecLists Top-Passwords.
2. **Rules anwenden** — `john --rules=Jumbo` oder `hashcat -r /usr/share/hashcat/rules/best64.rule` mutiert PWs (Leetspeak, Suffixe, ...).
3. **Hybrid-Attacks** — `wordlist + ?d?d?d?d` deckt typische `pw + Jahr/PIN`-Muster ab.
4. **Mask Brute-Force** — strukturiert nach Wahrscheinlichkeit (`?u?l?l?l?d?d?s` ist häufiger als `?a?a?a?a?a?a?a?a`).
5. **Inkrementell** — bei unbekannter Länge `--increment --increment-min=4 --increment-max=10` .
6. **Sessions speichern** — `--session=run1` + `--restore` für lange Läufe.
7. **Potfile checken** — Hashcat speichert Hits in `~/.local/share/hashcat/hashcat.potfile` → `--show`.
8. **GPU-Backend prüfen** — `hashcat -I` listet erkannte Devices. Auf AMD: `hashcat -d 1` zwingt GPU-only.
9. **Benchmark vor Brute-Force** — `hashcat -b -m <mode>` zeigt realistische Zeiten: `Cracking-Time = Keyspace / Speed`.
10. **Optimized Kernel** — `-O` aktiviert schnellere Kernel (PW-Länge ≤ 32).