

Mask & Hybrid Attacks

Move beyond wordlists. When you know the password structure, mask attacks let you define exact character patterns. Hybrid attacks combine wordlists with masks for even more power.

Learning Objectives

- Understand **mask attack mode** (`-a 3`) and built-in charset placeholders
 - Write masks that match known password patterns (PINs, names+digits, etc.)
 - Use **incremental mode** (`-i`) to sweep across multiple lengths
 - Define **custom charsets** with `-1`, `-2`, `-3`, `-4`
 - Run **hybrid attacks** (`-a 6` and `-a 7`) combining wordlists with masks
-

Mask Attack Basics (`-a 3`)

A mask attack generates candidates based on character position patterns. Instead of reading from a wordlist, Hashcat builds every possible combination matching the mask.

Built-in Charset Placeholders

Placeholder	Character Set	Example Characters
<code>?l</code>	Lowercase letters	<code>a-z</code>
<code>?u</code>	Uppercase letters	<code>A-Z</code>
<code>?d</code>	Digits	<code>0-9</code>
<code>?s</code>	Special characters	<code>!@#\$%^&*()...</code>
<code>?a</code>	All printable ASCII	<code>?l?u?d?s</code> combined
<code>?b</code>	Full byte range	<code>0x00-0xff</code> (256 values)

Each placeholder represents **one character position** in the candidate password.

Mask Examples

Each position in the mask defines what characters are allowed at that position.

4-Digit PIN

Mask: `?d?d?d?d`

Generates: 0000 through 9999

Keyspace: **10,000** candidates

Name + Digits Pattern

Mask: `?u?l?l?l?d?d`

Example: `Alex42`, `John99`

Keyspace: **26 x 26 x 26 x 26 x 10 x 10 = 45,697,600**

Any 4 Characters

Mask: `?a?a?a?a`

All printable 4-char passwords

Command

```
hashcat -m 0 -a 3 -o results.txt \  
hashes.txt ?d?d?d?d
```

Keyspace: $95^4 = 81,450,625$

Incremental Mode (-i)

When you don't know the exact password length, incremental mode gradually increases the mask length from a minimum to a maximum.

```
# Try all printable passwords from 4 to 8 characters
hashcat -m 0 -a 3 -i --increment-min 4 --increment-max 8 \
hashes.txt ?a?a?a?a?a?a?a
```

How It Works

- Starts at length `--increment-min` (default: 1)
- Runs the mask truncated to that length
- Increments by 1 and repeats until `--increment-max`
- The mask you provide must be at least as long as `--increment-max`

Warning: Incremental mode with `?a` grows exponentially. At 8 characters that is $95^8 = 6.6$ trillion

candidates. Use more constrained charsets when possible.

Custom Charsets

When built-in placeholders are too broad or too narrow, define your own charsets with `-1` through `-4`.

```
# Custom charset 1: lowercase + digits
hashcat -m 0 -a 3 -1 ?l?d hashes.txt ?1?1?1?1?1?1

# Custom charset mixing: uppercase first, then custom
hashcat -m 0 -a 3 -1 ?l?d -2 ?u?s \
hashes.txt ?2?1?1?1?1?d

# Literal characters in a custom charset
hashcat -m 0 -a 3 -1 abc123 hashes.txt ?1?1?1?1
```

Custom Charset Slots

- `-1 <charset>` — referenced as `?1`
- `-2 <charset>` — referenced as `?2`
- `-3 <charset>` — referenced as `?3`
- `-4 <charset>` — referenced as `?4`

You can combine built-in placeholders and literal characters in a single charset definition.

Hybrid Attacks

Hybrid attacks combine a wordlist with a mask, covering the common pattern of users appending or prepending characters to dictionary words.

Mode 6: Wordlist + Mask

Appends mask characters **after** each dictionary word.

```
# password123, summer99, etc.  
hashcat -m 0 -a 6 \  
hashes.txt rockyou.txt ?d?d?d
```

If rockyou has "password", generates:
password000 through password999

Mode 7: Mask + Wordlist

Prepends mask characters **before** each dictionary word.

```
# 123password, 42summer, etc.  
hashcat -m 0 -a 7 \  
hashes.txt ?d?d?d rockyou.txt
```

If rockyou has "password", generates:
000password through 999password

Tip: Hybrid attacks are extremely effective because many users simply add digits or special characters to the end of a common word.

Attack Mode Comparison

Mode	Flag	Source	Best For
Dictionary	<code>-a 0</code>	Wordlist	Common passwords, leaked lists
Combinator	<code>-a 1</code>	Wordlist + Wordlist	Two-word passwords
Mask / Brute-Force	<code>-a 3</code>	Pattern	Known structure (PINs, patterns)
Hybrid (dict+mask)	<code>-a 6</code>	Wordlist + Pattern	Words with appended chars
Hybrid (mask+dict)	<code>-a 7</code>	Pattern + Wordlist	Words with prepended chars

Strategy: Start with dictionary (`-a 0` + rules), then hybrid (`-a 6` / `-a 7`), then targeted masks (`-a 3`). Full brute-force is a last resort.

Lab & Homework

Lab Exercises

1. Create `hashes.txt` with MD5 hashes of known test passwords
2. Run a **4-digit PIN mask** attack with `?d?d?d?d`
3. Run a **complex mask** attack: `?u?1?1?1?d?d`
4. Run **incremental mode** from 4 to 8 characters
5. Run a **hybrid attack** appending 3 digits to dictionary words

Homework Tasks

1. Design custom charsets for a locale-specific password policy
 2. Benchmark mask vs hybrid vs dictionary attack speeds
 3. Create a multi-stage cracking script that chains attack modes
-

