

# Útok maskou

Vojtěch Žák

2022–2023

## Požadavky

Program	Verze
Hashcat	6.2.6
John the Ripper	1.9.0

## Použité soubory

Název	Popis
password.hash	Soubor obsahuje MD5 hash hesla <i>password</i>
mask.hcmask	Soubor obsahuje masku pro Hashcat

# Úvod

Prolamování hrubou silou má jeden zásadní problém a tím je rychlost. Proto vznikla takzvaná maska, se kterou omezíme znaky, které se mohou na danou pozici dosadit.

John the Ripper a Hashcat mají každý svoji definici, jak masku zapisovat. Je však pár základních možností, které mají společné. V následující tabulce jsou vypsány ty, které jsou společné a nejvíce se používají.

Maska	Charaktery
?l	abcdefghijklmnopqrstuvwxyz
?u	ABCDEFGHIJKLMNOPQRSTUVWXYZ
?d	0123456789
?s	speciální charaktery
?a	všechny ASCII znaky

## John the Ripper

John the Ripper má několik způsobů, jak složit masku. Ale základem je vždy argument `--mask`. Do něho se jako parametr dávají definice použitelných charakterů.[1, 2, 3, 4]

První způsob, jak zapsat masku je, že určíme pro každé místo použitelné charaktery. V našem příkladě chceme generovat hesla dlouhá osm znaků a použitelné charaktery, na všech místech budou malá písmena anglické abecedy. Proto jako parametr argumentu `--mask` zadáme `?l?l?l?l?l?l?l?l`. Je lepší masku dát do apostrofů, aby nenastaly problémy se speciálními znaky. Ty by mohly být špatně interpretovány příkazovým řádkem nebo samotným programem.

Protože útok maskou se bere jako vylepšení útoku hrubou silou, ostatní argumenty jsou stejné, jako v útoku hrubou silou. To znamená, že specifikujeme *Incremental mode* pomocí `--incremental` a zvolíme typ hashe. V našem případě se jedná o MD5 hash, proto jako parametr argumentu `--format` dáme `Raw-MD5`.

```
john --incremental --format=Raw-MD5
    --mask='?l?l?l?l?l?l?l?l' password.hash
```

Druhý způsob je, že v argumentu `--mask` určíme pouze povolené charaktery a délku určíme pomocí argumentů `--min-length` a `--max-length`. Abychom docílili toho samého, co v prvním způsobu, do `--mask` dáme parametr `?l`, do `--min-length` parametr 8 a do `--max-length` také 8.

```
john --incremental --format=Raw-MD5 --mask='?1'
--min-length=8 --max-length=8 password.hash
```

Tento způsob má výhodu v „pružnosti“. Představme si, že máme prolomit hesla, o kterých víme, že jsou dlouhá 8 až 16 znaků. Kdybychom používali první způsob (jenom `--mask`), museli bychom postupně manuálně přidávat do masky další znaky. Pokud bychom ale použili druhý způsob, stačí jenom dát do argumentu `--min-length` hodnotu 8 a do `--max-length` číslo 16. Masky se nám bude automaticky sama natahovat.

## Hashcat

Hashcat měl útok hrubou silou, ale nahradil ho právě útokem maskou. Proto neexistuje žádný argument, do kterého by se maska zadávala.[5, 6]

V Hashcatu se maska zadává na konec příkazu hned za souborem s hashem. V našem případě, kde chceme generovat hesla dlouhá osm znaků a pouze z malých písmen anglické abecedy, zadáme nakonec `?1?1?1?1?1?1?1?1`. Zároveň musíme zvolit útok maskou. To uděláme tak, že argumentu `--attack-mode` dáme hodnotu 3. Dále musíme specifikovat typ hashe v `--hash-type`. V našem případě se jedná o MD5, tudíž zadáme 0. Před maskou nesmíme zapomenout zadat soubor obsahující hash. V našem příkladě se jedná o soubor *password.hash*.

```
hashcat --attack-mode 3 --hash-type 0 password.hash
?1?1?1?1?1?1?1?1
```

Pokud bychom chtěli masku natahovat automaticky, musíme přidat argument `--increment`. Ten změní chování generování hesel tak, že maska se postupně od jednoho znaku bude natahovat do délky masky, kterou jsme zadali.

```
hashcat --attack-mode 3 --hash-type 0 --increment
password.hash ?1?1?1?1?1?1?1?1
```

Hashcat navíc umí načíst soubor s maskami. Tento soubor má většinou příponu *.hcmask*. Díky tomu můžeme zadat více masek najednou do jednoho příkazu. Struktura samotného souboru je velmi jednoduchá. Na každém řádku je vždy jedna maska. Příkladový soubor *mask.hcmask* vypadá následovně:

```
?1
?1?1
?1?1?1
?1?1?1?1
?1?1?1?1?1
```

```
?1?1?1?1?1?1  
?1?1?1?1?1?1?1  
?1?1?1?1?1?1?1?1
```

Cesta k tomuto souboru se zadává místo samotné masky, to znamená na konec příkazu. Pokud použijeme v našem příkladě soubor *mask.hcmask*, příkaz bude stejný jako v předchozím příkladě s argumentem `--increment`.

```
hashcat --attack-mode 3 --hash-type 0 password.hash  
mask.hcmask
```

## Reference

- [1] John the Ripper's cracking modes. *Openwall: bringing security into open computing environments* [online]. Openwall, 2013/05/29 17:57:56 [cit. 2023-03-18]. Dostupné z: <https://www.openwall.com/john/doc/MODES.shtml>
- [2] John the Ripper's command line syntax. *Openwall: bringing security into open computing environments* [online]. Openwall, 2016/01/21 05:10:00 [cit. 2023-03-18]. Dostupné z: <https://www.openwall.com/john/doc/OPTIONS.shtml>
- [3] John the Ripper usage examples. *Openwall: bringing security into open computing environments* [online]. Openwall, 2019/05/19 15:10:04 [cit. 2023-03-18]. Dostupné z: <https://www.openwall.com/john/doc/EXAMPLES.shtml>
- [4] john/MASK at bleeding-jumbo · openwall/john. *GitHub: Let's build from here* [online]. Github, Feb 18, 2022 [cit. 2023-03-18]. Dostupné z: <https://github.com/openwall/john/blob/bleeding-jumbo/doc/MASK>
- [5] hashcat. *hashcat: advanced password recovery* [online]. hashcat [cit. 2023-03-18]. Dostupné z: <https://hashcat.net/wiki/doku.php?id=hashcat>
- [6] Mask Attack. *hashcat: advanced password recovery* [online]. hashcat [cit. 2023-03-18]. Dostupné z: [https://hashcat.net/wiki/doku.php?id=mask\\_attack](https://hashcat.net/wiki/doku.php?id=mask_attack)