

Prolamování hrubou silou

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>

Úvod

Prolamování hrubou silou je nejzákladnější metoda na prolamování hesel. Spočívá v generování postupných kombinací znaků. Zároveň se jedná o jeden z nejpomalejších způsobů. Proto se používá jako poslední možnost.

John the Ripper

V programu John the Ripper je prolamování hrubou silou nazýváno jako *Incremental mode*. Pokud bychom nechali automatický mód (nezvolili žádný mód), tak John the Ripper použije *Incremental mode* jako poslední možnost.[1, 2, 3, 4]

Incremental mode zapneme pomocí `--incremental`. Následně ještě předáme informaci, že se jedná o MD5 hash přes argument `--format`. Nakonec napíšeme název souboru, který hash obsahuje.

```
john --incremental --format=Raw-MD5 password.hash
```

Po spuštění se vypíše informace, že se načtl jeden hash typu Raw-MD5. Následně se spustí samotné prolamování. V terminálu není žádný výstup. Ten se zobrazí až poté, co nějaký hash prolomíme. Vypsáný řádek bude obsahovat hash a vedle něho prolomené heslo. V našem případě se vedle hashe zobrazí *password*. Po prolomení všech hesel se program vypne a budeme moci zadat další příkaz.

Pokud by trvalo prolamování příliš dlouho, můžeme program zastavit pomocí klávesové zkratky *Ctrl + C* nebo zmáčknout klávesu *Q*. Program se zastaví a hesla, která jsme už prolomili, si zapamatuje.

Statistiky

Během prolamování můžeme zmáčknout klávesu *S*, pomocí které se nám ukáže statistika aktuální situace. Vypíše se nám jeden řádek s několika informacemi.

První údaj *g* určuje celkový počet prolomených hesel. Následuje čas, jak dlouho daný útok už běží. Údaj *g/s* ukazuje počet prolomených hesel za vteřinu. *P/s*, *c/s* a *C/s* zobrazují počet hesel vyzkoušených za jednu vteřinu. A na konci řádku jsou aktuálně zkoušená hesla.

Vypsání prolomených hesel

Abychom vypsali hesla, která John the Ripper prolomil, použijeme možnost `--show`. Zároveň musíme specifikovat typ hashe. Většinou se jedná o stejný typ, jako jsme zvolili u samotného prolamování.

```
john --show --format=Raw-MD5 password.hash
```

Ve výstupu se nám zobrazí hashe a vedle nich odpovídající hesla, stejně jako u samotného prolamování.

Vymazání prolomených hesel

Prolomená hesla si John the Ripper ukládá do souboru *john.pot*. Pokud bychom chtěli znovu spustit příkaz na prolamování, program by se vypnul s hláškou, že daný hash je již prolomen. Abychom donutili program, aby znova prolomil již prolomené heslo, musíme soubor *john.pot* smazat. Soubor můžeme najít pomocí programu `find`.^[5]

```
find / -name john.pot
```

Program nám vypíše všechny cesty, kde se soubor nachází. Můžeme soubor smazat manuálně (např. přes příkaz `rm`), nebo přidáme do přechozího příkazu argument `-delete`.

```
find / -name john.pot -delete
```

Nyní se nám nevypíše žádný výstup do konzole. Pro kontrolu, jestli se soubory skutečně smazaly, zadáme znova příkaz bez `-delete`. V konzoli by se neměla zobrazit žádná cesta.

Hashcat

Hashcat měl útok hrubou silou, avšak vývojem se z něho stal útok maskou. Neznamená to však, že by nešel zvolit, pouze se z něho stal speciální případ jiného útoku.^[6, 7]

Typ útoku vybíráme pomocí argumentu `--attack-mode`. Tomu následně předáme typ útoku pomocí čísla. V našem případě chceme zvolit útok maskou, která má číslo 3. Následně musíme specifikovat typ hashe, pomocí kterého je heslo hashované. Heslo v souboru *password.hash* je hashované pomocí hashe MD5. Proto předáme argumentu `--hash-type` parametr 0. Potom zadáme název souboru, který hash obsahuje. Nakonec specifikujeme, jaké charaktery se mají použít a jak dlouhé hádané heslo může být. Abychom použili útoku hrubou silou, musíme použít všechny charaktery (*?a*), a zadáme ho tolikrát, kolik odhadujeme, že prolamované heslo má znaků.

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

Na rozdíl od John the Ripper je Hashcat mnohem „ukecanější“. To může být výhoda i nevýhoda. Ve velkém množství výpisu se těžko orientuje, zato v něm najdeme více informací, které mohou být užitečné.

Hashcat nejdříve zobrazí informace o útoku, který se spustil. Prvně se ukáže, na kterých zařízeních se daný hash bude vypočítávat a jakou API použije. Většinou si Hashcat zvolí procesor s OpenCL API. Pokud je ale dostupná grafická karta, bude preferovat ji. V případě grafických karet od společnosti NVIDIA umí Hashcat využít CUDA rozhraní.

Vybrání zařízení provádí hashcat automaticky sám. Pokud bychom si chtěli zvolit, jaké zařízení má použít, použijeme argument `--backend-devices` nebo `--opencl-device-types`. Ve většině případů to není nutné, naopak by mohly nastat nějaké obtíže.

Po informaci o zařízeních se vypíše minimální a maximální délka podporovaných hesel, které umí Hashcat prolomit. Pokud by prolamované heslo bylo mimo hranice, Hashcat nedokáže dané heslo prolomit.

Údaj *Hashes* udává, kolik hashů bylo načteno, kolik z nich je jedinečných a kolik jedinečné soli bylo načteno. V našem případě by všude měla být jednička.

Následuje seznam optimalizací, které byly použity. Tyto optimalizace značně zrychlují hashování, nijak však neovlivňují samotný princip útoku.

Hashcat má vestavěnou funkci sledování teploty zařízení. Tento údaj se nazývá *Watchdog*. Pokud by byla překročena daná teplota, tak se Hashcat zastaví. K tomuto okamžiku však často nedochází.

A jako poslední ve výpisu je požadovaná velikost paměti, kterou Hashcat potřebuje. Pravidlem je, že čím více hesel najednou chceme prolomit, tím více paměti budeme potřebovat.

Nyní začne samotné prolamování. Stejně jako v John the Ripper se nám začnou vypisovat prolomená hesla. V našem případě se vypíše pouze jeden řádek s následujícím textem:
`5f4dcc3b5aa765d61d8327deb882cf99:password.`

Během toho, co Hashcat prolamuje hesla, můžeme zmáčknout několik kláves. Nejdůležitější jsou dvě: *S* a *Q*. Pomocí klávesy *Q* ukončíme program. Pomocí klávesy *S* vypíšeme aktuální statistiky.

Statistiky

Při prolomení všech hesel, při stisku klávesy *S* nebo při ukončení vypisuje Hashcat *status* neboli statistiky.

V těchto statistikách je spousta informací. Nejdůležitější však jsou: *Time Estimated*, *Recovered* a *Progress*. Ostatní informace nejsou zbytečné, ale slouží buď v případě, kdy máme spuštěných více útoků najednou, nebo se dané informace dají odvodit.

Time Estimated (předpokládaný čas) předpovídá datum a čas, kdy budou vyzkoušeny všechny možnosti. Tento čas je vypočítán pomocí vydělení zbývajících možností rychlostí generování hashů. Kromě datumu a času je v závorce uveden zbývajíc čas od doby, kdy se statistika vypsala. Důležité je si uvědomit, že tento čas se může měnit, většinou z důvodu změny rychlosti generování hesel. To již závisí na samotném hardwaru. Například se postupně rychlost může snižovat s tím, jak se zařízení zahřívá.

Recovered (obnoveno nebo prolomeno) udává počet prolomených hesel. Jsou zde dvě kategorie: *total* a *new*. *Total* je počet všech prolomených a načtených hesel. *New* je počet hesel, která nebyla nalezena v souboru obsahujícím již prolomená hesla. První číslo před lomítkem je počet prolomených hesel, druhé číslo je celkový počet hesel.

Progress (postup) udává počet hesel, která byla již vyzkoušena. Pokud provádíme útok hrubou silou nebo útok maskou, bude se jednat o počet kombinací znaků. V případě, že bychom prováděli slovníkový útok, bude se jednat o počet slov ve slovníku.

Vypsání prolomených hesel

Pro vypsání prolomených hesel použijeme možnost `--show`, stejně jako u John the Ripper. Zároveň specifikujeme typ hashe pomocí argumentu `--hash-type`.

```
hashcat --show --hash-type 0 password.hash
```

Program nám vypíše hashe a vedle nich odpovídající hesla.

Vymazání prolomených hesel

Stejně jako u John the Ripper si Hashcat již prolomená hesla ukládá do souboru. Proto pokud bychom chtěli znovu prolamovat již prolomený hash, musíme smazat tento soubor. Postup je stejný jako u John the Ripper, pouze místo souboru *john.pot* chceme smazat *hashcat.potfile*.

```
find / -name hashcat.potfile -delete
```

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 the Ripper FAQ. *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/FAQ.shtml>
- [5] GNU Findutils 4.9.0. *The GNU Operating System and the Free Software Movement* [online]. GNU Operating System [cit. 2023-03-18]. Dostupné z: https://www.gnu.org/software/findutils/manual/html_mono/find.html
- [6] hashcat. *hashcat: advanced password recovery* [online]. hashcat [cit. 2023-03-18]. Dostupné z: <https://hashcat.net/wiki/doku.php?id=hashcat>
- [7] Mask Attack. *hashcat: advanced password recovery* [online]. hashcat [cit. 2023-03-18]. Dostupné z: https://hashcat.net/wiki/doku.php?id=mask_attack