Passwords 101

Jeff Deifik

jeff@deifik.com

About Me

- MS in Cybersecurity, CISSP, C|CISO
- Software for first e-commerce system (from 1985-1995)
- Software for the first orbiting radio telescope satellite
- Software for the most advanced pulse oxineter
- Cybersecurity for government satellite ground control, balancing sound cybersecurity with cost and schedule. Currently employed at The Aerospace Corp.
- Interest in the intersection of cybersecurity and software development began with white hat password cracking over 30 years ago.

Tools

- John The Ripper
  - Infrequent official releases, Many unofficial releases
  - Poor Graphical Processor Unit (GPU) windows support
  - Easy to make custom rules
  - Good mailing list support
  - Doesn’t parallelize small number of passwords
- HashCat
  - Frequent releases – sometimes they don’t work
  - Great GPU acceleration
  - Primitive rule syntax

Wordlists

- Some very high quality
- Most stuffed full of junk and require editing
  - Very long lines, often thousands of characters long
  - Non ASCII letters
  - Separators that are not newlines
  - Since they are big, specialized tools are needed
- Rockyou2021 is a bit big, but very high quality

My Custom Wordlist Tools

- short -s 40 foo > foo.40
- short -l 41 foo > foo.41
  - splits foo into 2 files, 40 chars and shorter, and 41 chars and longer
- msort -l foo > foo.l
  - sorts foo by line length
- ascii-lines -p foo > foo.p
  - only outputs lines of foo compromised solely of printable ascii characters
- multi-merge foo.1 foo.2 foo.3 > foo.123
  - merges any number of sorted files into a big sorted file
- sample -10000 foo > foo.10k
  - outputs one line every 10000 lines, for sampling foo
- line_len foo
  - Prints line length counts
- count foo
  - Character frequency count

Standard Wordlist Tools

- gnu sort
  - You generally want to process sorted wordlists
  - Works with files bigger than RAM using tmp files
- uniq
  - Remove duplicate words
- comm
  - Removes duplicate words in different files
- emacs
  - The one true editor, regular expressions, can process gigabyte files

Hashing Speed

- NTLM Speed 41825.0 MH/s
- md5 Speed 24943.1 MH/s
- LM Speed 18382.7 MH/s
- descrypt Speed 906.7 MH/s
- SHA1 Speed 788.2 MH/s
- scrypt Speed 435.1 kH/s
- WPA2 Speed 396.8 kH/s
- bcrypt Speed 13094 H/s
  https://gist.github.com/epixoip/a83d38f412b4737e99bbef804a270c40

Salt

- 1979 Unix 12 bits, 4,096 different salts
  - https://spqr.eecs.umich.edu/courses/cs660sp11/papers/10.1.1.128.1635.pdf
- 1980's Unix 48 bits, 281,474,976,710,656
- 1996 bcrypt 128 bits, 3.4 x 10^38 salts
- Argon2 128 bits, 3.4 x 10^38 salts
- Descrypt uses 12 bits of salt
- LM and NTLN doesn't use salt ☹

My descrypt dump – basic John the Ripper (JTR)

- Data encryption standard crypt (descrypt)
  - 8 char max length
- 1576 passwords
- Took @5 years to find all of them
- JTR dev pack 2023_05_14, no options (CPU)
  - 537 words in 126 seconds
  - 667 words in 633 seconds
  - 758 words in 1,201 seconds
  - 884 words in 3,985 seconds (wordlist password.lst)
  - 1022 words in 22,128 seconds (incremental:ASCII)

My descrypt dump - wordlists

Using JTR dev pack 2023_05_14 with no options (using my CPU) found

- Rockyou (139mb) found 455 words in 52 sec
- Weakpass 2a (37gb) found 469 words in 5,675 sec
- Rockyou2021 (98gb) found 967 words in 12,363 sec
- Prince & Rockyou found 634 words in 9,243 sec
  - (Probability INfinite Chained Elements)

Hashing Speed

- I have a 64 core AMD EPYC milan system
- I have a Nvidia 3060ti 8gb graphics card
  - 2.3 times faster than a Nvidia 1070 graphics card
  - It is 5-30 times faster than my EPYC
  - I picked it to maximize the performance / price. I got it in 2022 for $400

Performance

- Wordlists
  - Can be very fast depending on quality & size
- Wordlists with rules
  - Faster than brute force
- Brute force, 1 password
  - All 8 char lower @3 minutes on GPU
  - All 8 char letters @13 hours on GPU
  - All 8 lower, number, special @5.5 days on GPU ☹
  - All 8 char printable @80 days on GPU ☹ ☹

Rainbow Tables

- Doesn’t play nice with salt
- Very very fast ☺
- Works with LM, NTLM, MD5, etc.
- Defcon data duplication village – 6tb drives
  - freerainbowtables.com GSM A51 and MD5 hash tables
  - more rainbowtables, lanman, mysqlsha1, ntlm, and some word lists

Password Statistics

Length:

3 : 0.3 % (3)
4 : 0.3 % (3)
5 : 2.8 % (30)
6 : 21.6 % (233)
7 : 25.5 % (276)
8 : 49.6 % (536)

Chars:

All lower: 60.2 % (651)
All lower digit: 15.9 % (172)
All lower upper: 10.8 % (117)
All lower upper digit: 3.6 % (39)
All lower special: 5.5 % (59)
All upper: 0.2 % (2)
All digits: 0.1 % (1)
All special: 0.1 % (1)
All upper digit: 0.1 % (1)
All digit special: 0.1 % (1)
All lower upper special: 1.7 % (18)
All lower digit special: 1.4 % (15)
All lower upper digit special: 0.4 % (4)

String Classes:

All alpha: 71.1 % (769)
Alphas + numbers: 11.0 % (119)
Alphas + specials: 2.1 % (23)
Alphas + numbers + alphas: 6.5 % (70)
Alphas + specials + alphas: 4.3 % (47)

Odd passwords

- One tab character
  - No standard character set includes tabs
  - I created a custom character set to include tab
- One control-R character
  - No standard character set includes control chars
  - After failing an exhaustive search, I knew there was a control character.
  - I tested current linux to see what control chars were allowed in a psssword.
  - With the help of john the ripper mailing list, I made a special rule to insert a control char, as well as replace a char with a control char. Using a big dictionary, I found the password.

JTR rule for control chars

[List.Rules:insert_control_1]

>\r[00-6] '7 i\p[0-7][\x7f\x80\x01-\x1f]

[List.Rules:replace_control_1]

>[0-7] '8 o\0[\x7f\x80\x01-\x1f]

This would be 32*8 lines for each rule in hashcat.

1980 BSD Unix password dump

- Most passwords quickly cracked
- Bill Joy's password was the last. It is chess^Win
- Ken Thompson's password is p/q2-q4!
- Dennis Richie's password is dmac
- Stephen Bourne's password is bourne

1980 BSD Unix password stats

- All lower: 42.3 % (11)
- All lower digit: 15.4 % (4)
- All lower special: 23.1 % (6)
- All special: 3.8 % (1)
- All lower digit special: 15.4 % (4)

- All alpha: 38.5 % (10)
- Alphas + numbers: 7.7 % (2)
- Alphas + specials: 19.2 % (5)
- Alphas + numbers + alphas: 7.7 % (2)
- Alphas + specials + alphas: 7.7 % (2)

1980 BSD passwords

dmac	uio	bourne
foobar	network	whatnot
axolotl	sacristy	uucpuucp
cowperso	jilland1	/.,/.,
apr1744	...hello	sherril.
wendy!!!	5%ghj	pdq;dq
theik!!!	sn74193n	p/q2-q4!
graduat;	12ucdort	561cml..
..pnn521	chess^Win	

Defense

- 2 factor authentication
  - What you have - Titan security key, yubikey, smartcard
  - What you are - Fingerprint, Face ID
- Use cryptographically strong random passwords
- Use a password manager
  - keepass, 1password, bitwarden
- I wrote a password generator, here is some output:

password is K)dE;pN%(]R~H6L-11!R bits 129
password is GAw->8k?+Qou#(*#L:Z0 bits 129
password is YmytLWazQ[g{0R@}I2ha bits 129
password is _a^W9h8[J~jsO)*6ahaQ bits 129
password is [q;)y_):BTJAfHZU)7.* bits 129

Other Stuff

- You will want to undervolt / underclock your GPU to save power
  - MSI Afterburner works well, windows specific
- https://www.openwall.com/presentations/OffensiveCon2024-Password-Cracking/

Dictionaries

47,085,595 linked.dic
72,382,568 SkullSecurityComp.dic
93,559,564 10-million-passwords.dic
94,461,698 ignis-10M.dic
139,749,969 10-million-user-pass.dic
139,921,988 rockyou.dic
362,881,958 hk_hlm_founds.dic
382,000,913 collection_1_5_v3.dic
1,075,899,306 superpass_fixed.dic
1,305,699,616 facebook-lastnames.dic.l33t
1,643,295,189 kac.dic
2,266,396,047 Super_mega_dic.dic
2,277,681,952 exploit.in.dic
3,107,889,706 thedefinitvepasswordlist_complete_.dic
4,276,546,161 HashesOrg.dic
5,403,987,782 hibp_515_found.dic
11,432,450,014 b0n3z.dic
13,675,962,135 hashesorg2019.dic
13,832,356,359 crackstation_fixed.dic
17,264,739,583 Md5decrypt-awesome-wordlist.dic
17,539,451,065 collection_1_5_v1.dic
17,868,066,068 DCHTPassv1.0.dic
18,166,067,612 naxxatoe-dict-total-new-unsorted.dic
18,624,885,828 HYPER-WORDLIST-DIC.dic
21,102,866,314 b0n3z-sorted-wordlist.dic
37,241,758,679 weakpass_2a.dic
41,514,529,952 collection_1_5_v2.dic
98,378,212,907 rockyou2021.dic
123,968,583,755 WordlistBySheez_v8.dic