Hashes; Smothered, Covered, and Scattered: Modern Password Cracking as a Methodology

HELLO!

Lee Wangenhiem

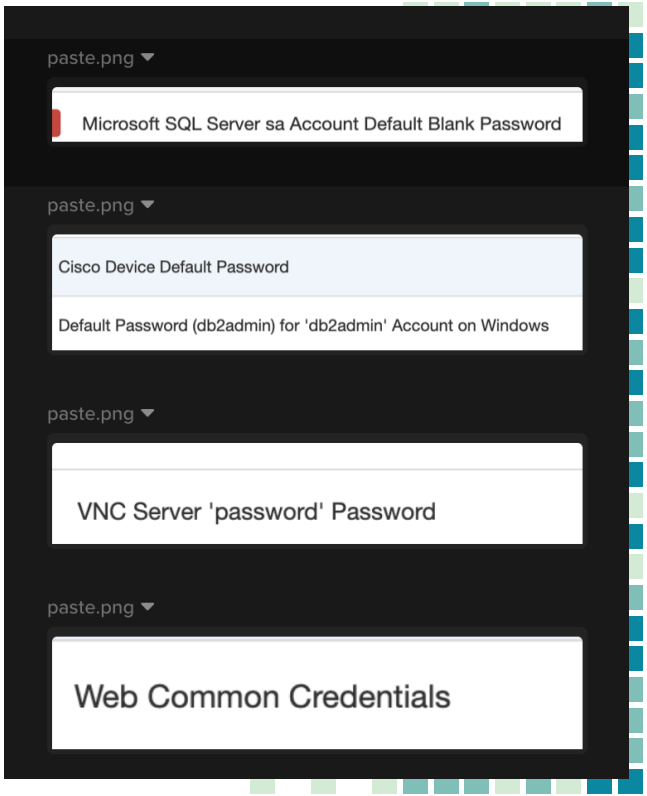
- Security Consultant @ Optiv
- Hacks things for fun as well as for a job
- 5 years Infosec Experience
- Helps run the crackers at Optiv

aspserver

Why does it matter?

Fall2019

Because it only takes one...



1.
Hardware

Set Up For Success

So you want to do some cracking?

The Old:

CPU – Not really worth it at all

Rainbow Tables – Mostly irrelevant

The New:

GPU – Any Desktop gaming setup will do

Cloud – Scalable but spendy

Laptops – Good for when hashes cannot leave client site

Mining Rigs – Great source of used video cards, with an income stream

CLOUD

AI and Machine Learning Opening New Doors

So cracking in the CLOUD?

Summer 2017:

AWS's best GPU enabled system is the G2.8XLarge powered by 4 Nvidia GRID 520's **($2.28)**

Hashes at 16 GH's Per Second

14 Cents per GH Hour

Fall 2017:

AWS releases their new P3.16XLarge instances powered by 8 Nvidia Tesla V100's **($24.48)**

Hashes at 633 GH's Per Second

3 Cents per GH Hour

Optiv Built Cracker 2017:

6x1080 GPU's in fully redundant server configuration **(~$25,000)**

Hashes at 250 GH's Per Second

If used 80% of the time for 2 years

.7 Cents per GH Hour

Estimated Password Recovery Times — 1x Terahash Brutalis, 44x Terahash Inmanis (448x Nvidia RTX 2080)

Alphanumeric mask attack with Terahash Hashstack

Speed	Length 4	Length 5	Length 6	Length 7	Length 8	Length 9	Length 10	Length 11	Length 12	Length 13
NTLM 31.82 TH/s	Instant	Instant	Instant	Instant	Instant	7 mins 6 secs	7 hrs 19 mins	2 wks 4 days	3 yrs 2 mos	199 yrs 2 mos
MD5 17.77 TH/s	Instant	Instant	Instant	Instant	Instant	12 mins 42 secs	13 hrs 7 mins	1 mo 0 wk	5 yrs 9 mos	356 yrs 7 mos
NetNTLMv1 / NetNTLMv1+ESS 16.82 TH/s	Instant	Instant	Instant	Instant	Instant	13 mins 25 secs	13 hrs 51 mins	1 mo 0 wk	6 yrs 0 mo	376 yrs 10 mos
LM 15.81 TH/s	Instant	Instant	Instant	Instant						
SHA1 5.89 TH/s	Instant	Instant	Instant	Instant	Instant	38 mins 18 secs	1 day 15 hrs	3 mos 1 wk	17 yrs 4 mos	1.1 mil
SHA2-256 2.42 TH/s	Instant	Instant	Instant	Instant	1 min 31 secs	1 hr 33 mins	4 days 0 hr	8 mos 0 wk	42 yrs 2 mos	2.6 mil
NetNTLMv2 1.22 TH/s	Instant	Instant	Instant	Instant	3 mins 0 sec	3 hrs 5 mins	1 wk 0 day	1 yr 4 mos	83 yrs 10 mos	5.2 mil
SHA2-512 801.9 GH/s	Instant	Instant	Instant	Instant	4 mins 33 secs	4 hrs 41 mins	1 wk 5 days	2 yrs 0 mo	127 yrs 5 mos	7.9 mil
descrypt, DES (Unix), Traditional DES 647.59 GH/s	Instant	Instant	Instant	Instant	5 mins 38 secs	5 hrs 48 mins	2 wks 1 day	2 yrs 6 mos	157 yrs 10 mos	9.8 mil
Kerberos 5, etype 23, TGS-REP 206.97 GH/s	Instant	Instant	Instant	Instant	17 mins 35 secs	18 hrs 10 mins	1 mo 2 wks	7 yrs 11 mos	493 yrs 11 mos	30.6 mil
Kerberos 5, etype 23, AS-REQ Pre-Auth 206.78 GH/s	Instant	Instant	Instant	Instant	17 mins 36 secs	18 hrs 11 mins	1 mo 2 wks	7 yrs 11 mos	494 yrs 5 mos	30.7 mil
md5crypt, MD5 (Unix), Cisco-IOS 1 (MD5) 7.61 GH/s	Instant	Instant	Instant	7 mins 44 secs	7 hrs 58 mins	2 wks 6 days	3 yrs 5 mos	216 yrs 9 mos	13.4 mil	833.9 mil
LastPass + LastPass sniffed 1.78 GH/s	Instant	Instant	Instant	32 mins 54 secs	1 day 9 hrs	2 mos 3 wks	14 yrs 10 mos	924 yrs 0 mo	57.3 mil	3554 mil
macOS v10.8+ (PBKDF2-SHA512) 335.09 MH/s	Instant	Instant	2 mins 50 secs	2 hrs 55 mins	1 wk 0 day	1 yr 3 mos	79 yrs 4 mos	4.9 mil	305.3 mil	18926.3 mil
WPA-EAPOL-PBKDF2 277.23 MH/s					1 wk 2 days	1 yr 6 mos	95 yrs 11 mos	6 mil	369 mil	22876.6 mil
TrueCrypt RIPEMD160 + XTS 512 bit 211.78 MH/s	Instant	Instant	4 mins 29 secs	4 hrs 37 mins	1 wk 4 days	2 yrs 0 mo	125 yrs 7 mos	7.8 mil	483 mil	29947.1 mil
7-Zip 181.51 MH/s	Instant	Instant	5 mins 13 secs	5 hrs 23 mins	1 wk 6 days	2 yrs 4 mos	146 yrs 6 mos	9.1 mil	563.6 mil	34940.7 mil
sha512crypt 6, SHA512 (Unix) 119.46 MH/s	Instant	Instant	7 mins 56 secs	8 hrs 11 mins	3 wks 0 day	3 yrs 7 mos	222 yrs 7 mos	13.8 mil	856.3 mil	53090.5 mil
DPAPI masterkey file v1 47.23 MH/s	Instant	Instant	20 mins 3 secs	20 hrs 42 mins	1 mo 3 wks	9 yrs 0 mo	563 yrs 1 mo	34.9 mil	2165.7 mil	134271.5 mil
RAR5 28.15 MH/s	Instant	Instant	33 mins 39 secs	1 day 10 hrs	2 mos 4 wks	15 yrs 2 mos	944 yrs 11 mos	58.6 mil	3634.4 mil	225334 mil
DPAPI masterkey file v2 27.82 MH/s	Instant	Instant	34 mins 2 secs	1 day 11 hrs	2 mos 4 wks	15 yrs 5 mos	955 yrs 11 mos	59.3 mil	3676.7 mil	227953.7 mil
RAR3-hp 20.84 MH/s	Instant	Instant	45 mins 26 secs	1 day 22 hrs	3 mos 4 wks	20 yrs 6 mos	1.3 mil	79.2 mil	4907.7 mil	304274.7 mil
KeePass 1 (AES/Twofish) and KeePass 2 (AES) 17.8 MH/s	Instant	Instant	53 mins 12 secs	2 days 6 hrs	4 mos 2 wks	24 yrs 1 mo	1.5 mil	92.7 mil	5746.9 mil	356305.7 mil
bcrypt $2*$, Blowfish (Unix) 11.37 MH/s	Instant	1 min 21 secs	1 hr 23 mins	3 days 14 hrs	7 mos 1 wk	37 yrs 8 mos	2.3 mil	145.1 mil	8996 mil	557755.1 mil
Bitcoin/Litecoin wallet.dat 3.55 MH/s	Instant	4 mins 18 secs	4 hrs 26 mins	1 wk 4 days	1 yr 11 mos	120 yrs 8 mos	7.5 mil	464.2 mil	28782.1 mil	1784492.8 mil

Thanks @TerahashCorp

2.
Glossary

So We Can Speak The Same Language

A Couple Terms

Masks – The makeup of a word, broken into it's character set

Hybrid Attack – An attack where a Brute-Force or mask is either appended or prepended to a wordlist

Wordlist – A file which contains a list of candidate words to either run by themselves or be modified with rules, typically dictionary words

Password Dump – A file which contains passwords obtained from previous cracking attempts, will contain more complex words than a wordlist

3.
Tools

Creating Your Environment

Building an arsenal

"If Your Only Tool Is a Hammer Then Every Problem Looks Like a Nail"

Mark Twain

The Kit

Hashcat – The Hammer

Hashtopolis – The Toolbelt

HashID – Magnifying Glass

PW_Spy – Measuring Tape

Hashcat

Defacto standard. Supports almost every hash imaginable. Fast. Constant updates/improvements.

Replaced JohnTheRipper

Easy to setup and integrate with other tools

Hashtopolis

Hashtoplis

Wrapper for Hashcat

Manage agents, jobs, wordlists, and hashcat binaries from a central location

Distributed cracking made easy!!!

HashID

```
sh-3.2$ md5 -s "Password123"
MD5 ("Password123") = 42f749ade7f9e195bf475f37a44cafcb
sh-3.2$ hashid -m -j 42f749ade7f9e195bf475f37a44cafcb
Analyzing '42f749ade7f9e195bf475f37a44cafcb'
[+] MD2 [JtR Format: md2]
[+] MD5 [Hashcat Mode: 0][JtR Format: raw-md5]
[+] MD4 [Hashcat Mode: 900][JtR Format: raw-md4]
[+] Double MD5 [Hashcat Mode: 2600]
[+] LM [Hashcat Mode: 3000][JtR Format: lm]
[+] RIPEMD-128 [JtR Format: ripemd-128]
[+] Haval-128 [JtR Format: haval-128-4]
[+] Tiger-128
[+] Skein-256(128)
[+] Skein-512(128)
[+] Lotus Notes/Domino 5 [Hashcat Mode: 8600][JtR Format
```

Find likely hashing algorithms

If its not helpful

- Research Application – Source code?
- Try a commonly used password first
- Self register known password

PW_spy

Tool built out of our Enterprise Password Audits

Finds:

- Most common masks
- Weak Passwords
- Password Lengths
- Base words

https://github.com/lwangenheim/pw_spy

4.
Techniques

Honing Your Skills

How do I begin?

What's the best way to crack a hash?

This is a loaded question, what's the best way to use Nmap?

Think about the engagement, are you going after one hash? Multiple hashes?

What algorithm are you trying to crack?

NTLM is MUCH faster than WPA2

How do I look?

Where do we get hashes?

- Hashdump – local accounts
- /etc/shadow or .conf files
- Mimikatz
- WebApps
- Responder
- DCSync/NTDS

Developing a Methodology

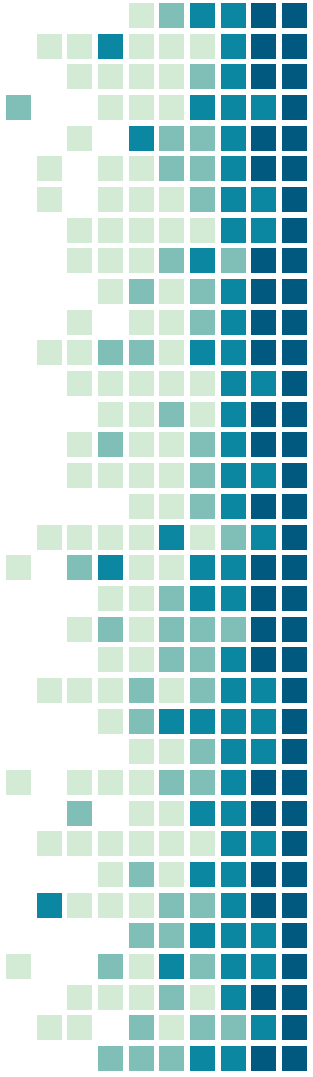
Methodology – Password Audit

- Creating a repeatable process for others to follow
- Looks at the entire enterprise
- Very analysis based
- Looking for patterns, common words, easy wins, etc.
- Some claim they do pw audits but they don't do it effectively
- Need heavy hitting cracking rigs / cloud setup

What do I do?

How did we get there? (quick wins)

- Proprietary wordlists without rules
- Adding rules to those same lists
- Loopback attacks
- 1-8 char brute force
- Masks (start with uppercase, end with digits/special chars)



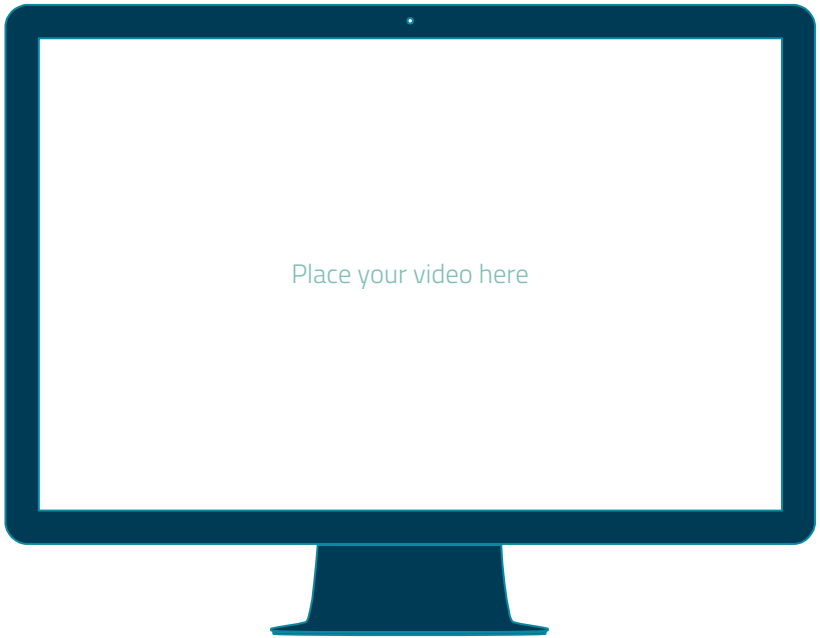
Executing the Methodology

Help Your Future Self

- Pot Files
  - Historical record of your cracked hashes
  - Useful to see if you've already cracked a hash on another engagement
  - Be wary of bloat, it can slow down the process as each hash is run through the existing potfile
- Common Masks
  - Build a list of masks > 8 characters to run through

5.
Demo

Demo Gods Hate Me



THANKS!

Any questions?

You can find me at:

@Hx_fifty

CREDITS

Special thanks to all the people who dedicate time to content in this presentation:

- Hashcat
- Hashtopolis
- Optiv
- Presentation template by SlidesCarnival
- Photographs by Unsplash