A Report on the 1Password Cracking Challenge

Jeffrey Goldberg
1Password

A belated Report on the 2018 1Password Cracking Challenge

Jeffrey (Slowpoke) Goldberg
1Password

These slides

https://github.com/agilebits/crackme/blob/master/doc/PasswordsCon2020.pdf

How strong should your master password be?

Old answer: The strongest you can reasonably and reliably remember and use.

New answer: It depends, but generally strongest you can reasonably and reliably remember and use.

Threat model

1Password is weird

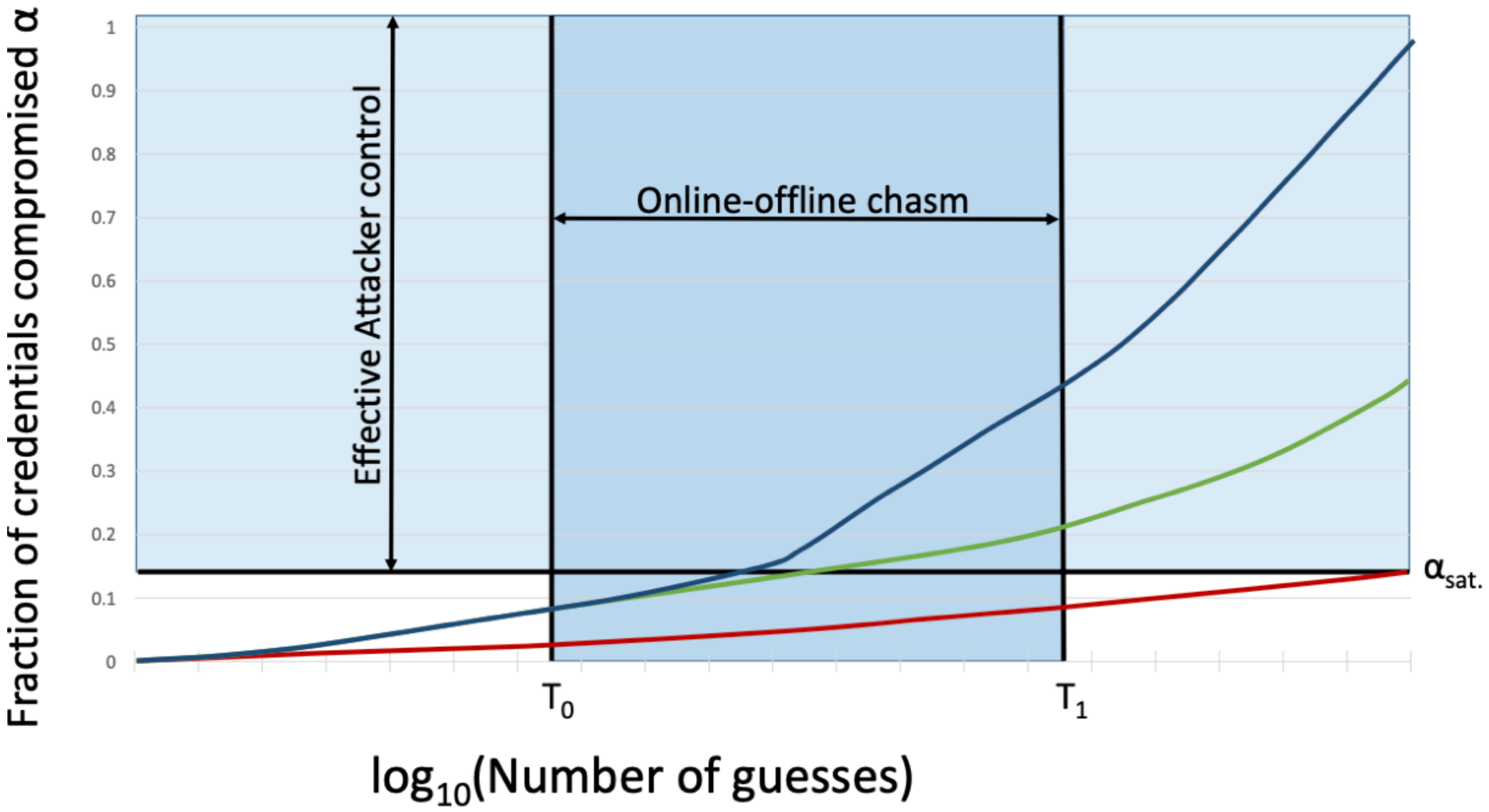
1. Master Password (MP) is combined during key derivation client side with a high entropy secret (Secret Key), which is stored on the client

2. Thus, what is stored server side is uncrackable

3. Secret Key is (typically) available to an attacker who gains read access to the user's local device

4. Thus, MP defends against attacker who gets user data from the user's device.

Chasm or Mountaink



Dinei et al. [2016] argued that there is an important class of situations where there is little gain in improving the strength of a password if it can't be made invulnerable to an off-line attack. They called this the "chasm of don't care."

Mountain of Must Care

A password manager master password is different. We must care a great deal about that region

Attacker Alice gets victim Victor’s 1Password data from one of V’s devices. How quickly must *V* change his passwords?

If *V* has a terrible MP, then he is toast. If he has an uncrackable one, he is just fine. But if – as is most likely – he has something in between then the amount of time he has increases with the strength of the MP

What we were pushing

During 1Password sign-up (as it stood in those days) we steered users to use a four word password from our generator (56.65 bits), but our UI also allowed for three word generation (42.45 bits).

We wanted to know whether 42.45 bits was acceptable.

Money is better than time*

- “How long does it take to crack” is the wrong question
- “How much does it cost to crack” is a better question
- Spoiler: 6USD per 2^{32} guesses (2018 figures). $4300 for 42.5 bits.

**This of course is not generally true, as the backwards bending supply curve for labor exemplifies*

Contest desiderata

- Attract serious, experienced, crackers
- Be winnable
- Passwords chosen from a known *uniform* distribution
- Be hard enough so that there is a good mixture of fixed costs and running costs
- Not break the bank
- Be open and transparent

Setup

- We worked with Bugcrowd to provide independence and transparency with awarding prizes.

- We published the source used to generate the hashes along with test samples ahead of the actual launch date

- The day before launch I generated the real challenges to be published at the pre-announced time

- Published PGP signature of the challenge file and of the solution file prior to launch time

- Challenge hashes published Noon, EDT on World Password Day, May 3, 2018. 2018-05-03 16:00:00 +0000 UTC

A copy of the solutions

- Solution file was encrypted with symmetric PGP, with a strong password stored in 1Password). The encrypted file was written to a CD and kept hidden away. It only briefly lived on a network connected device
- It was kept in case of dispute if there were an accusation that we didn’t generate the passwords according to our stated rules
- It turned out to be really good that we did keep a copy

The KDF

- The 1Password KDF (Key Derivation Function) is complicated and messy. (And includes the Secret Key)

- The relevant (slow) part is simple: PBKDF2-HMAC-SHA256, 100,000 rounds

A challenge

Seven challenges were published

```
"id": "NO4VRU4S",
"hint": "3 words",
"prf": "HMAC-SHA256",
"rounds": 100000,
"salt": "8ad1712ab5d632d8c4dac07b792ebb17",
“derived":
  "a3a8b8eb8e739c86f67332d17364b149cd88f33bb11eedae066ac3
  66711ec266"
```

A sample

Three samples were published

```
"id": "3UOKUEBO",
"hint": "3 words",
"sample": true,
"prf": "HMAC-SHA256",
"rounds": 100000,
"salt":
 “e65814e4382759f85550029e723dc7e7",
"derived":
 "5f37a3bd08ac1c7d163294a3cb192ed1407b62
 bbc6a6259fee55f6e53f754273",
"pwd": "governor washout beak”
```

Getting the incentives right

- If we'd known how much it would cost to crack a 42.5 bit 1Password Master Password password we could have set the challenge and incentives better.
- If we'd known how much it would cost to crack a 42.5 bit 1Password Master Password password we wouldn't have needed to run the the contest.
- First guess was between 500USD and 4000USD.

Prizes

Initial (May 3, 2018)

USD for Nth person/team to crack an as yet uncracked challenge

N	Initial	June 11	July 26
1st	4096	8192	12288
2nd	2048	4096	8192
3rd	1024	2048	6144
4th	0	1024	4096

Prizes

First doubling (June 11, 2018)

USD for Nth person/team to crack an as yet uncracked challenge

N	Initial	June 11	July 26
1st	4096	8192	12288
2nd	2048	4096	8192
3rd	1024	2048	6144
4th	0	1024	4096

Prizes

Final (July 26, 2018)

USD for Nth person/team to crack an as yet uncracked challenge

N	Initial	June 11	July 26
1st	4096	8192	12288
2nd	2048	4096	8192
3rd	1024	2048	6144
4th	0	1024	4096

Hints

- Quadrupling prizes wasn’t enough
- Hints needed to be fair to those both those who had started working on a challenge and those who hadn’t.
- Hints had to be uniform and measurable
- Consult with participants about approach to offering hints

Hints

- Compute (using the saved solutions) the first N bits of a fast hash of the password.
- Hints needed to be fair to those both those who had started working on a challenge and those who hadn’t.
- Hints had to be uniform and measurable
- Turned out to be good that we’d kept a copy of the solutions

Hints

Leading (big endian) bit of ~~SHA1~~ SHA256(password)

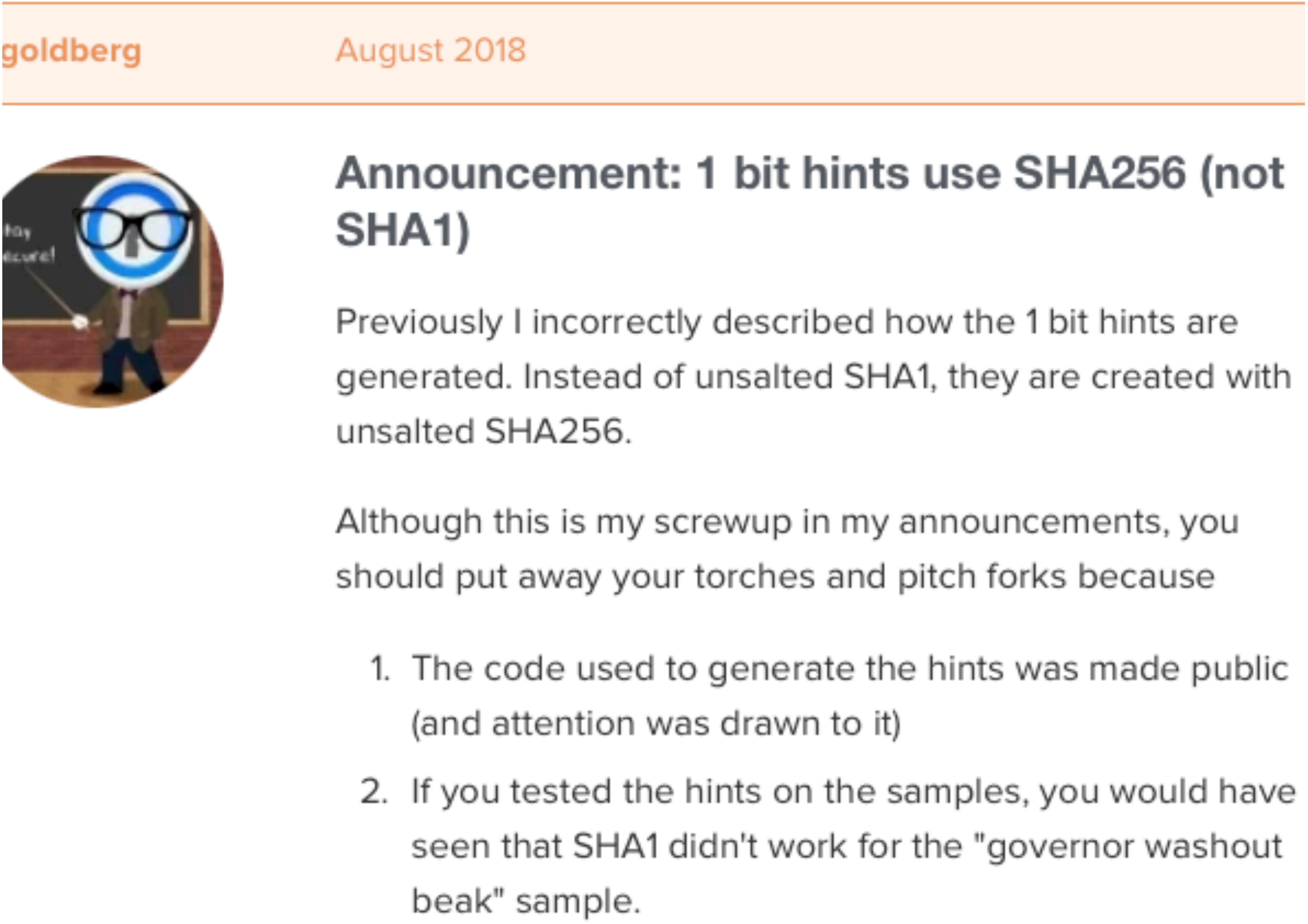
CHESAPEAKE
BEVERAGE

Crypto
means
Cryptography

plano

Hints

- August 5: Code for generating hints published result from samples.
- August 23: First hints (1 bit of fast hash of pwd) published
- August 25: Corrected description of the hints, as code used SHA256, but I had said SHA1 in some places.
- September 24: Second hint (additional bit of fast hash) published



goldberg August 2018

Announcement: 1 bit hints use SHA256 (not SHA1)

Previously I incorrectly described how the 1 bit hints are generated. Instead of unsalted SHA1, they are created with unsalted SHA256.

Although this is my screwup in my announcements, you should put away your torches and pitch forks because

1. The code used to generate the hints was made public (and attention was drawn to it)
2. If you tested the hints on the samples, you would have seen that SHA1 didn't work for the "governor washout beak" sample.

Wins

ID	Password	Hint	Date found
DOHB6DC7	mansard humpback unbutton	0b00	2018-10-14
SFELTO3W	faint bust perturb	0b00	2018-11-07
2SB5OP3G	befell car granary	0b00	2018-11-10
5BSLBTKR	minute judd obedient	0b10	2019-01-10

Winners

- 1st, 2nd, 3rd place: A somewhat fluid team that at times included s3inlc, winxp5421, blazer, hops, m33x, milzo, gearjunkie.

- 4th place: groozavu, ninjaslikecheese

Write-ups

https://github.com/agilebits/crackme/tree/master/write-ups

Costs

- “11,550 USD” for the price of the GPUs. Amortized (over three years) to 10.54 per day.
- “Rigs cost us approximately $16.24 per day to run”
- Computing their guess rate (from time on the project and portion of key space searched) they were would have taken about 4500 hours to exhaust half of the keyspace without hints. [computation]
- “Average rate of 209.85 kH/s”
- So between $4300 and $4860 to crack a three word password without hints [computation]

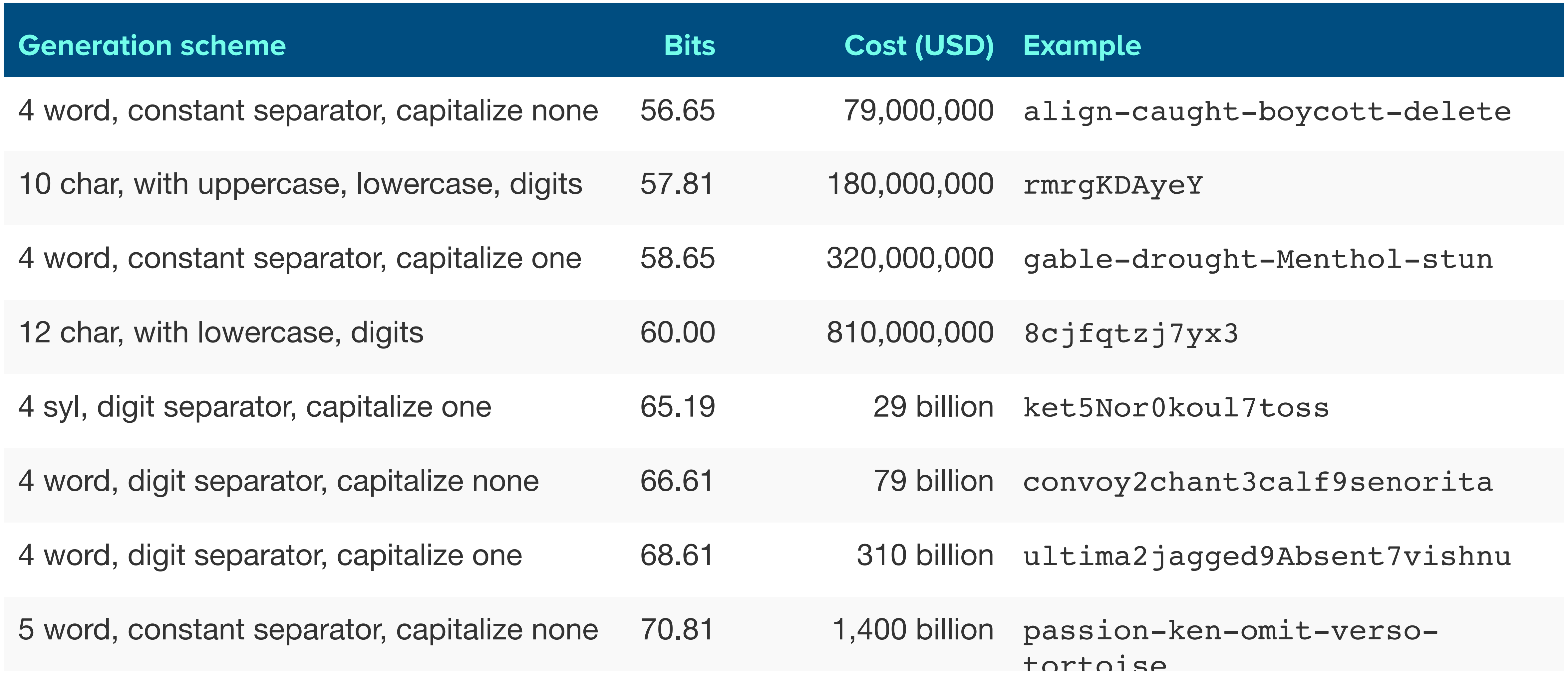
Costs

Assuming the worst (from the defender point of view) that was 6USD per 2^{32} guesses of PBKDF2-H256 100000 rounds in late 2018.

Costs for generated password (page 1 of 2)

Generation scheme	Bits	Cost (USD)	Example
3 word, constant separator, capitalize none	42.48	4,300	`prithee-insured-buoyant`
3 word, constant separator, capitalize one	44.07	13,000	`Dent-impanel-minority`
9 char, with lowercase, digits	45.00	25,000	`azdr3oqxc`
8 char, with uppercase, lowercase, digits	46.25	58,000	`8NhJqHPY`
3 syl, digit separator, capitalize one	48.15	220,000	`Best0jogh2gno`
3 word, digit separator, capitalize none	49.13	430,000	`swatch2forte1dill`
10 char, with lowercase, digits	50.00	790,000	`fovav9v6ot`
3 word, digit separator, capitalize one	50.71	1,300,000	`saute7docket3Bungalow`
9 char, with uppercase, lowercase, digits	52.03	3,200,000	`siFc96vGw`
11 char, with lowercase, digits	55.00	25,000,000	`aev7x9cgm3q`
4 syl, constant separator, capitalize one	55.22	29,000,000	`paghdeygibFrom`

Costs for generated password (page 2 of 2)



Generation scheme	Bits	Cost (USD)	Example
4 word, constant separator, capitalize none	56.65	79,000,000	`align-caught-boycott-delete`
10 char, with uppercase, lowercase, digits	57.81	180,000,000	`rmrgKDAyeY`
4 word, constant separator, capitalize one	58.65	320,000,000	`gable-drought-Menthol-stun`
12 char, with lowercase, digits	60.00	810,000,000	`8cjfqtzj7yx3`
4 syl, digit separator, capitalize one	65.19	29 billion	`ket5Nor0koul7toss`
4 word, digit separator, capitalize none	66.61	79 billion	`convoy2chant3calf9senorita`
4 word, digit separator, capitalize one	68.61	310 billion	`ultima2jagged9Absent7vishnu`
5 word, constant separator, capitalize none	70.81	1,400 billion	`passion-ken-omit-verso-tortoise`

See: https://github.com/agilebits/crackme/tree/master/doc/Costs

What I got wrong

Incentives: Price risk

Incentives: Price opportunity costs

Incentives: Attract more teams

Resources

- Challenge on Github, including write-ups and some of these calculations: https://github.com/agilebits/crackme

- Discussion on 1Password.community, including some of the other computations and discussing between us and participants as things were running https://1password.community/discussion/89318/world-password-day-cracking-challenge/p1