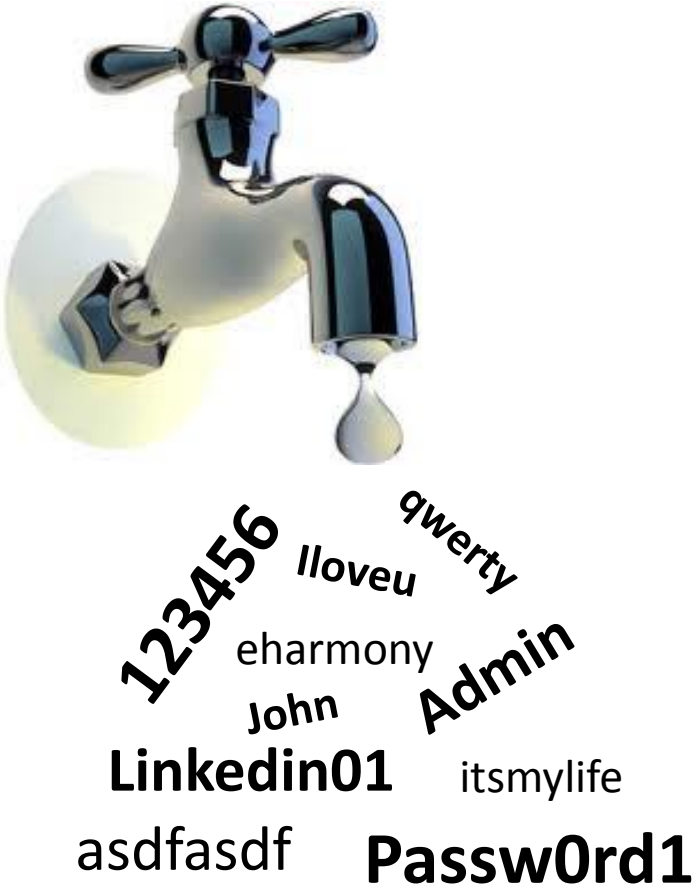


Lessons Learned From Recent Password Leaks…

CORE SECURITY

About Us

Flavio de Cristófaro (Core Security Technologies)

VP of Engineering for Professional Products

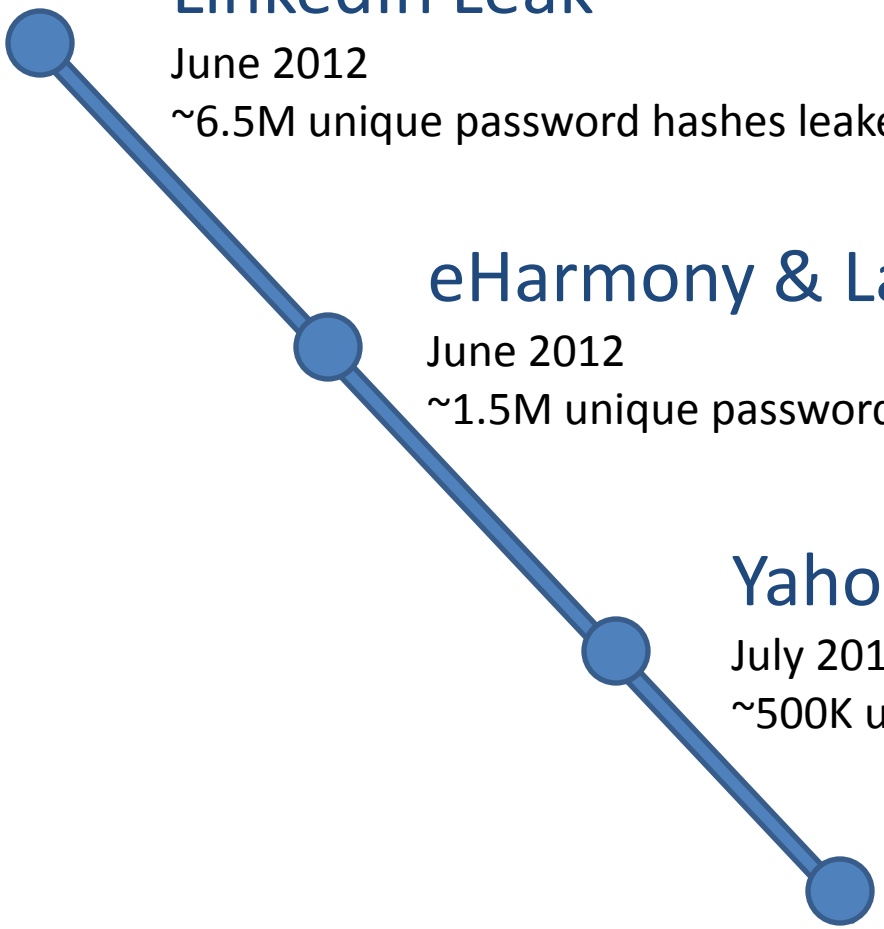
Brian Sutton (Core Security Technologies)

Security Consultant – Security Consulting Services

Ernesto Alvarez (Core Security Technologies)

Security Consultant – Security Consulting Services

Intro…Recent Password Leaks…



LinkedIn Leak

June 2012

~6.5M unique password hashes leaked (+80% cracked)

eHarmony & Last.fm Leaks

June 2012

~1.5M unique password hashes leaked (+95% cracked)

Yahoo Leak

July 2012

~500K unique passwords leaked (Plain)

IEEE Leak

September 2012

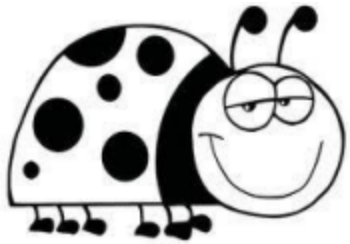
~100K unique passwords leaked (Plain)

CORE SECURITY

What's a Leak?

Most common ways password hashes are stolen

- SQL Injection
- Direct database/server compromise
- Unprotected backups
- Sniffers



Password Protection
Common Denominator

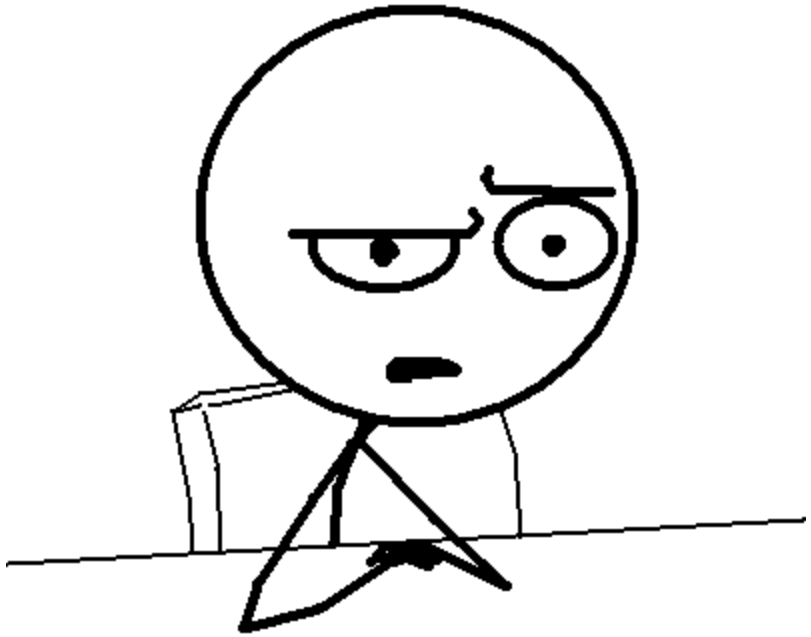
LinkedIn
SHA1
Min 6 characters

eHarmony
MD5 - All uppercase
Min 5 characters

Yahoo / IEEE
Plain text passwords

- **Passwords stored in plain**
- **or using unsuitable hashing functions**
- **No salt**
- **Short passwords allowed**
- **Complexity - Optional**

OK… So you already have my password hashes… Now what ?



What is password cracking?

Original password hash = “NG6glxWE69l”

Hash(“Pass”) = sTjGW7aAJoS  NG6glxWE69l

Hash(“pAss”) = 9h/pkn75pqV  NG6glxWE69l

Hash(“paSs”) = t13.KBlApJV  NG6glxWE69l

Hash(“pasS”) = NG6glxWE69l 

NG6glxWE69l

Original password = “pasS”

Cracking techniques…

Brute Force

- Basic
- Mask
- Markov

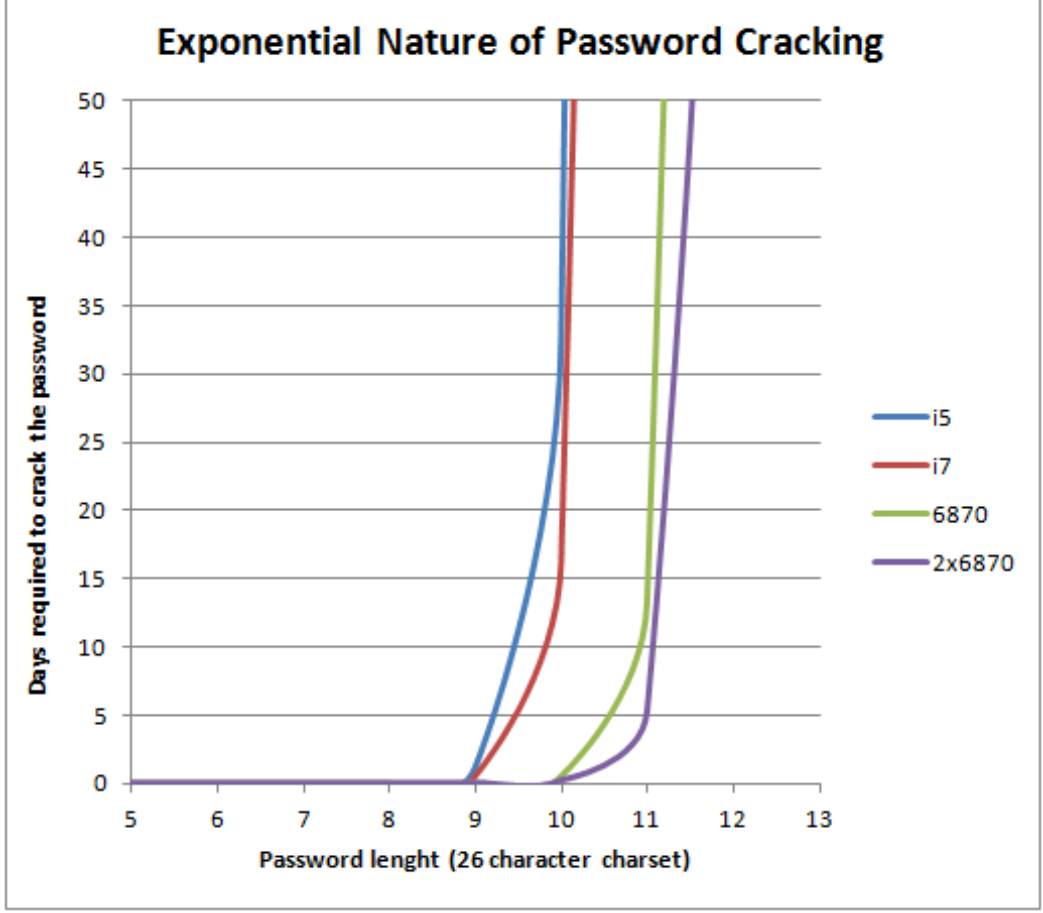
Dictionary Attack

- Basic
- Rule based
- Word combination
- Other variations

Rainbow Tables

CORE SECURITY

CPU vs GPU



34 x Intel i7

$ 34000

2 x Radeon 6870

$ 350 + $ 1000

Nice video - CPU vs GPU:
http://www.nvidia.com/object/nvision08_gpu_v_cpu.html

CORE SECURITY

Consequences

Passwords protected by…

- Weak hashing functions
- Unsalted
- Short length
- Limited complexity

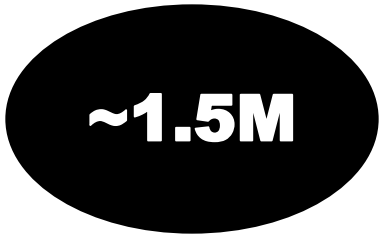
Can be easily translated into plain texts

(4 cases – More than 6M plain password exposed)

OK… What about a **REAL** life example?

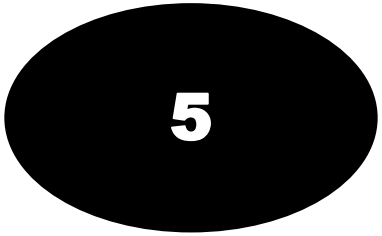
CORE SECURITY

Exploratory Phase Results…



Hashes leaked

Used to protect passwords

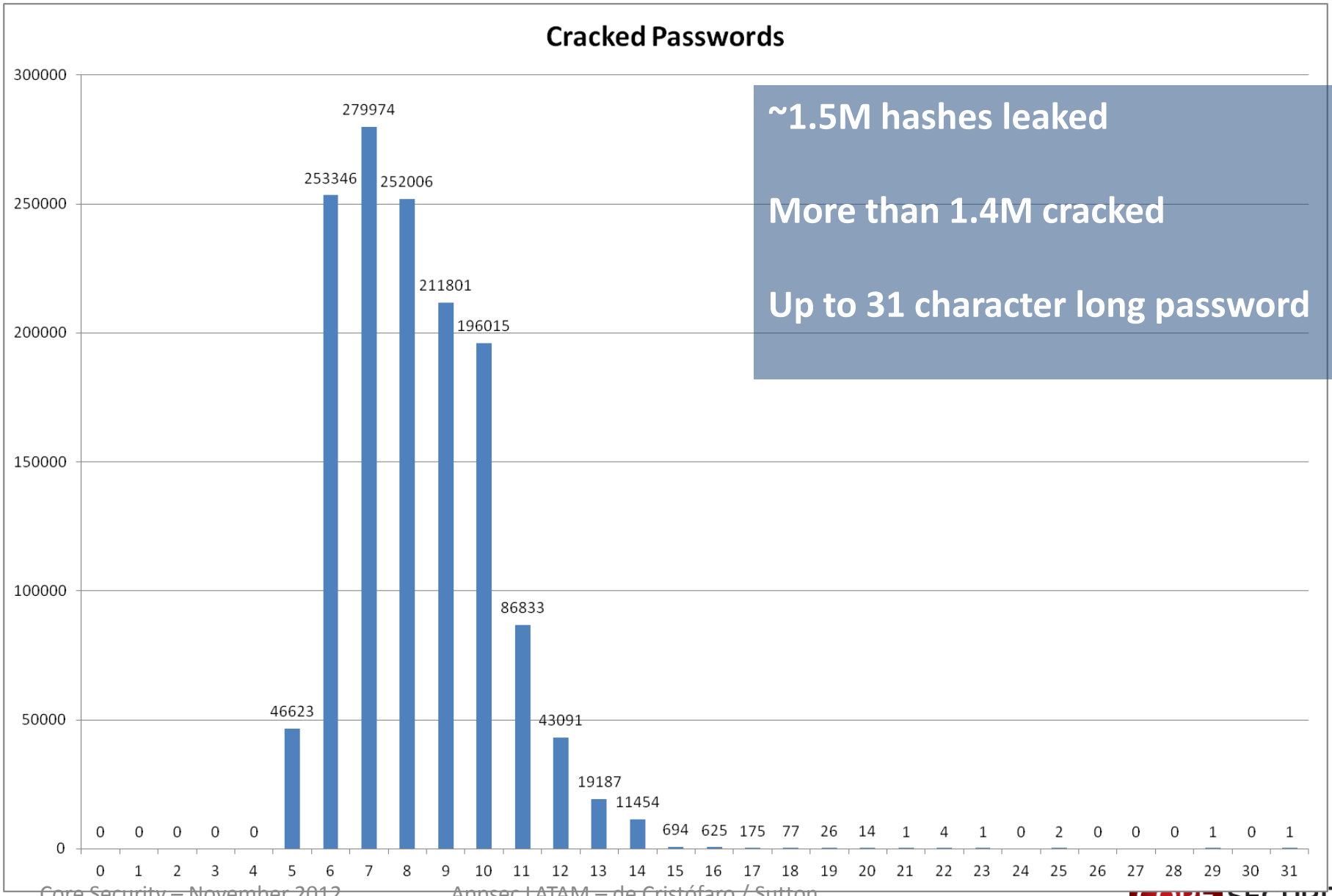


Minimum password length in use

All the passwords are uppercase

CORE SECURITY

Results – Cracked Passwords



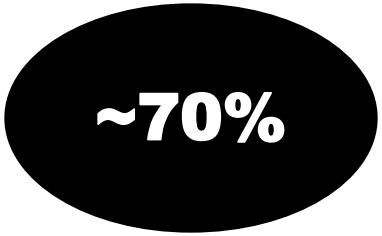
~1.5M hashes leaked

More than 1.4M cracked

Up to 31 character long password

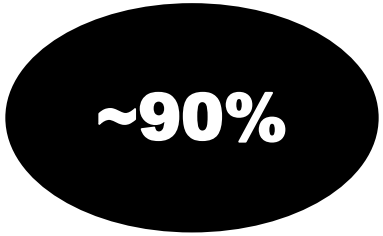
CORE SECURITY

Results – Time Required



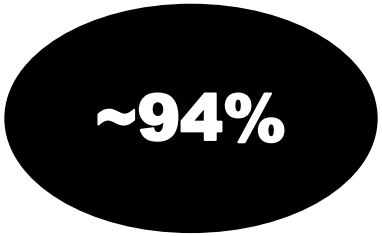
cracked in an hour or so

(Dictionary attack)



cracked in a few days

(Adding Markov and basic brute-force)

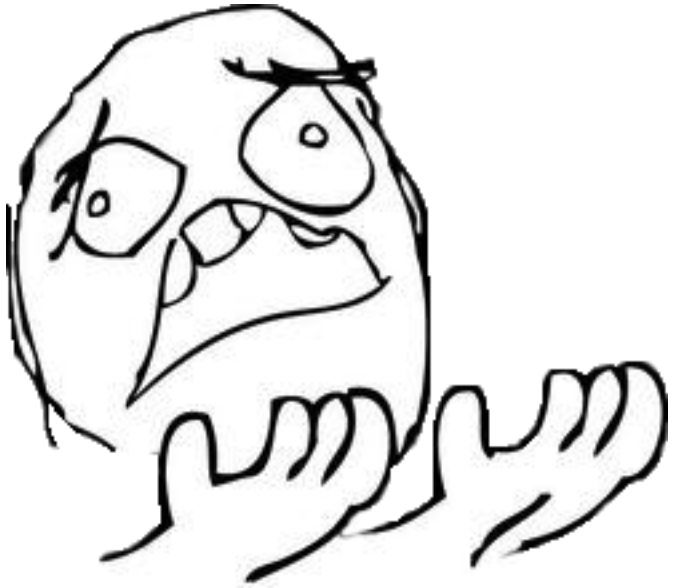


cracked in a few weeks

(Using brute-force)

CORE SECURITY

OMG!
Can we protect ourselves from this?



Password hash strength relies on 4 main rules…

1. **Password management**
   (Encryption algorithm adequacy and robustness, salt, etc)
2. **Password length**
3. **Charset in use**
4. **Password Randomness**
   (vs Human-generated)

Hashing Algorithm

Fast algorithms	Slow Algorithms
MD5	bcrypt
SHA-1	scrypt
DES	PBKDF2

Cryptographic hashing

Password hashing

TABLE 1. Estimated cost of hardware to crack a password in 1 year.

KDF	6 letters	8 letters	8 chars	10 chars	40-char text	80-char text
DES CRYPT	< $1	< $1	< $1	< $1	< $1	< $1
MD5	< $1	< $1	< $1	$1.1k	$1	$1.5T
MD5 CRYPT	< $1	< $1	$130	$1.1M	$1.4k	$1.5 × 10^{15}
PBKDF2 (100 ms)	< $1	< $1	$18k	$160M	$200k	$2.2 × 10^{17}
bcrypt (95 ms)	< $1	$4	$130k	$1.2B	$1.5M	$48B
scrypt (64 ms)	< $1	$150	$4.8M	$43B	$52M	$6 × 10^{19}
PBKDF2 (5.0 s)	< $1	$29	$920k	$8.3B	$10M	$11 × 10^{18}
bcrypt (3.0 s)	< $1	$130	$4.3M	$39B	$47M	$1.5T
scrypt (3.8 s)	$900	$610k	$19B	$175T	$210B	$2.3 × 10^{23}

By COLIN PERCIVAL, developer of scrypt algorithm

Salting

```
user@linux:~$ md5pass PAS$W0rD brian
      $1$brian$.ERov0Tp2BuBDLdC6cryv0

user@linux:~$ md5pass PAS$W0rD flavio
     $1$flavio$C0j87LRQWD8av3ib7hBIi1

user@linux:~$ md5pass PAS$W0rD someone
    $1$someone$2xLja3WP43LasMAjQLKSS.
```

Password Policies

Charset weight (for an ‘n’ characters password):

Charset	possible combinations
a-z or A-Z	26^n
a-z and A-Z	52^n
a-z and A-Z and 0-9	62^n
a-z and A-Z and 0-9 and symbols	95^n

Length weight (assuming 62 character charset):

Length (characters)	possible combinations
6	5,6E+10
7	3,5E+12
8	2,2E+14
10	8,4E+17
12	3,2E+21

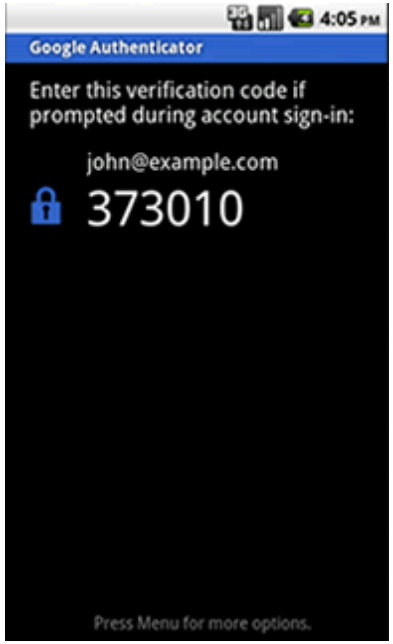
The Human Factor

Popular Password Managers:

- KeePass
- LastPass
- 1Password

Two-factor authentication:

- Google Authenticator
- SMS/email one-time passcodes
- Physical Tokens
- Code Cards



CORE SECURITY

1 - A crypto algorithm is necessary but…

2 - Slow algorithms slows password cracking

3 - Salting is not a silver bullet… but improves protection a lot

4 - Brute-force turns unfeasible with a few extra characters… Size matters.

Final Thoughts

CORE SECURITY