Seminar on Cryptography and Security

City College of San Francisco

Jack

CryptoHack

October 16, 2021

Good Morning

Today’s Plan

- Would love to have this talk be student led
- Happy to keep this informal, so if at any point a question pops into your head, then feel free to ask[1]
- I’ll take a few minutes to introduce myself to you all
- We can then jump straight into a Q&A, else I have some slides to share with you all

[1] *Disclaimer*: I don’t promise to always have a good answer!

A Random Walk to Cryptography

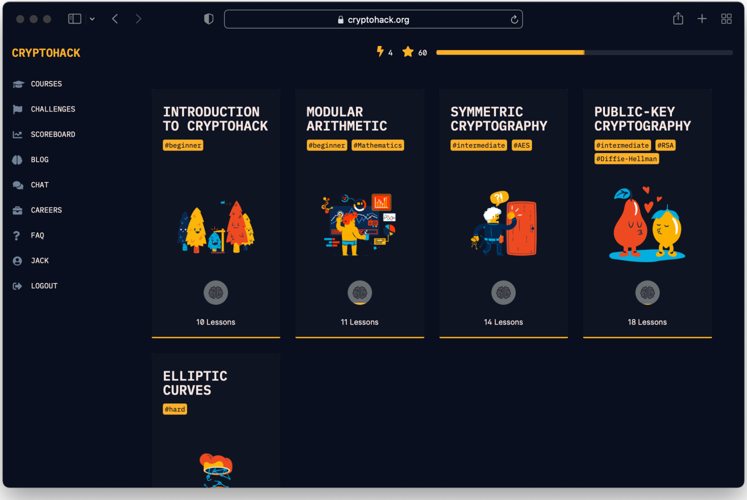
About Me

- Art School Dropout
- PhD in Theoretical Physics
- Discovered cryptography via CTFs
- Co-founder of CryptoHack
- Previously: Security Engineer with Northrop Grumman
- Coming soon: Consultant with the NCC Group Cryptography Services team

`https://cryptohack.org`

`https://www.nccgroup.com/uk/assessment-advisory/cryptography/`

CryptoHack



cryptohack.org

CRYPTOHACK

⚡ 4 ★ 60

- COURSES
- CHALLENGES
- SCOREBOARD
- BLOG
- CHAT
- CAREERS
- FAQ
- JACK
- LOGOUT

INTRODUCTION TO CRYPTOHACK
#beginner
10 Lessons

MODULAR ARITHMETIC
#beginner #Mathematics
11 Lessons

SYMMETRIC CRYPTOGRAPHY
#intermediate #AES
14 Lessons

PUBLIC-KEY CRYPTOGRAPHY
#intermediate #RSA #Diffie-Hellman
18 Lessons

ELLIPTIC CURVES
#hard

CryptoHack

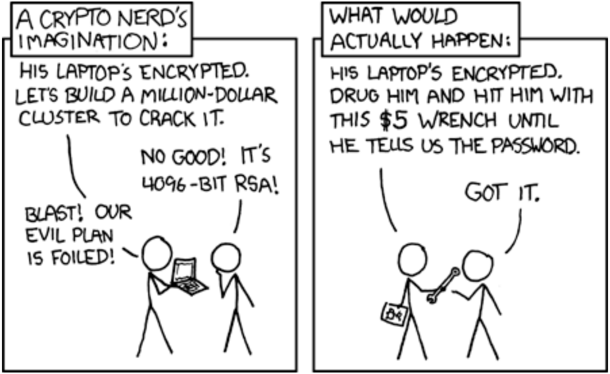
- Learn cryptography by breaking it
- 150+ puzzles and interactive challenges
- Fundamentals, AES, RSA, Diffie-Hellman, Elliptic Curves, hash functions, …
- Active community on Discord

Most of what I know about cryptography, I learnt from creating CryptoHack challenges and talking with my CryptoHack friends.

Go and play CTFs!

- A good CTF covers all cyber security areas:
  - ⋆ **Cryptography** (My favourite)
  - ⋆ Pwn (Binary exploitation)
  - ⋆ Web
  - ⋆ Reverse engineering
  - ⋆ Forensics
- Chasing flags in teams is a great way to learn from each other and see other ways of problem solving
- Hard CTFs get you to the cutting edge of research
- For the competitive people here, CTFs are a great motivator to learn new topics!

Cryptography and Security


A CRYPTO NERD'S IMAGINATION:
HIS LAPTOP'S ENCRYPTED. LET'S BUILD A MILLION-DOLLAR CLUSTER TO CRACK IT.
NO GOOD! IT'S 4096-BIT RSA!
BLAST! OUR EVIL PLAN IS FOILED!
WHAT WOULD ACTUALLY HAPPEN:
HIS LAPTOP'S ENCRYPTED. DRUG HIM AND HIT HIM WITH THIS $5 WRENCH UNTIL HE TELLS US THE PASSWORD.
GOT IT.

https://xkcd.com/538/

Three brands of failure

Very roughly, we see security vulnerabilities associated with *bad cryptography* in the following three scenarios:

Common mistakes

- Engineers have created their own cipher suite
- Engineers have taken secure cipher suites but incorrectly implemented (part of) the code
- Secret information has somehow been leaked

Don’t Roll your Own Crypto

Mistake One

Engineers have created their own cipher suite

- Cryptography is very hard to design
- It’s easy to list snake-oil and bizarre cryptosystems, but even giants fail
- What cryptosystems do you know of which have been broken / retired

Don’t Roll your Own Crypto

Mistake One

Engineers have created their own cipher suite

- Secret attacks: Lucifer / DES

Don’t Roll your Own Crypto

Mistake One

Engineers have created their own cipher suite

- Secret attacks: Lucifer / DES
- Outdated hash functions: MD5, SHA1

Don’t Roll your Own Crypto

Mistake One

Engineers have created their own cipher suite

- Secret attacks: Lucifer / DES
- Outdated hash functions: MD5, SHA1
- Retired ciphers: RC4

Don’t Roll your Own Crypto

Mistake One

Engineers have created their own cipher suite

- Secret attacks: Lucifer / DES
- Outdated hash functions: MD5, SHA1
- Retired ciphers: RC4
- Original RSA suggested $N \simeq 2^{266}$, current NIST recommendation: $N \simeq 2^{4096}$

Don’t Roll your Own Crypto

Mistake One

Engineers have created their own cipher suite

- Secret attacks: Lucifer / DES
- Outdated hash functions: MD5, SHA1
- Retired ciphers: RC4
- Original RSA suggested $N \simeq 2^{266}$, current NIST recommendation: $N \simeq 2^{4096}$
- Original Diffie-Hellman suggested $p \simeq 2^{200}$, current NIST recommendation: $p \simeq 2^{2048}$.

Don’t Roll your Own Crypto

Mistake One

Engineers have created their own cipher suite

- Secret attacks: Lucifer / DES
- Outdated hash functions: MD5, SHA1
- Retired ciphers: RC4
- Original RSA suggested $N \simeq 2^{266}$, current NIST recommendation: $N \simeq 2^{4096}$
- Original Diffie-Hellman suggested $p \simeq 2^{200}$, current NIST recommendation: $p \simeq 2^{2048}$.
- DES can now be exhaustively cracked: `https://crack.sh`

Mistake Two

Engineers have taken secure cipher suites but incorrectly implemented (part of) the code

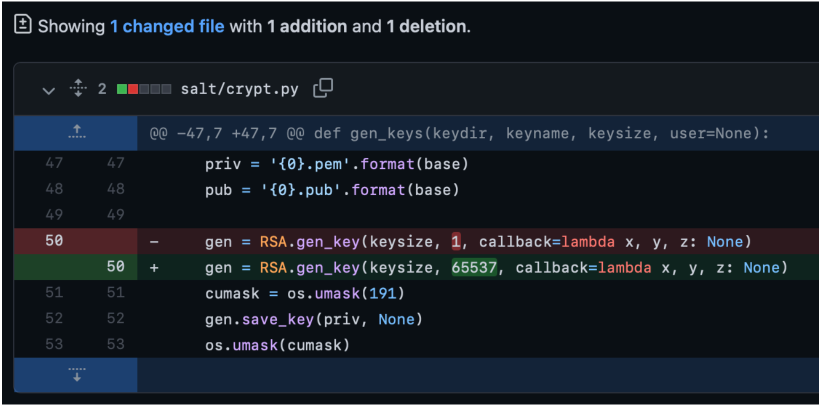
- This is a far more common security flaw
- What kind of problems do you imagine may have happened?

Mistake Two

Engineers have taken secure cipher suites but incorrectly implemented (part of) the code

- Bad public-key parameters chosen

Saltstack



Showing 1 changed file with 1 addition and 1 deletion.

2 🟩🟥□□□ salt/crypt.py

```
@@ -47,7 +47,7 @@ def gen_keys(keydir, keyname, keysize, user=None):
         priv = '{0}.pem'.format(base)
         pub = '{0}.pub'.format(base)

-        gen = RSA.gen_key(keysize, 1, callback=lambda x, y, z: None)
+        gen = RSA.gen_key(keysize, 65537, callback=lambda x, y, z: None)
         cumask = os.umask(191)
         gen.save_key(priv, None)
         os.umask(cumask)
```

https://github.com/saltstack/salt/commit/5dd304276ba5745ec21fc1e6686a0b28da29e6fc

Return of the Coppersmith Attack (ROCA)

- Estonian ID cards were protected with RSA
- Millions of cards were needed to be created, so engineers came up with a *fast* way to generate large primes:

$$p = k \cdot M + (65537^a \mod M)$$

- Here k, a are secret integers, but M is the product of the first n primes.
- Too much of these primes are known, and we can use mathematics to recover p, q from N.

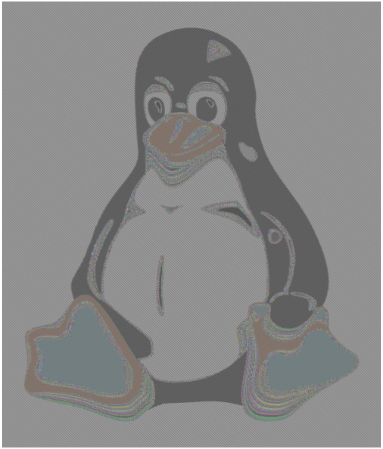
https://en.wikipedia.org/wiki/ROCA_vulnerability

Mistake Two

Engineers have taken secure cipher suites but incorrectly implemented (part of) the code

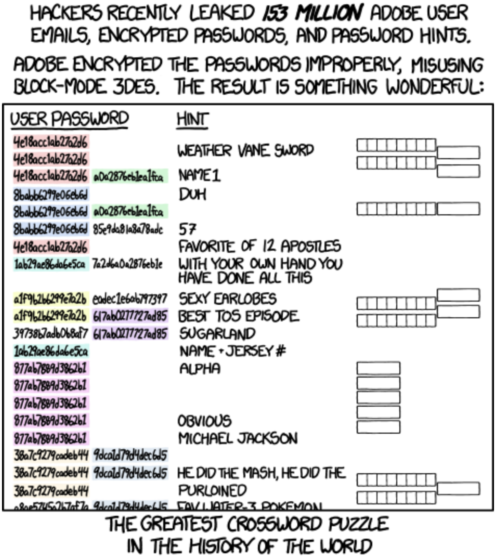
- Bad public-key parameters chosen
- The wrong block cipher modes chosen

ECB Penguin



https://blog.filippo.io/the-ecb-penguin/

Abode Crossword


HACKERS RECENTLY LEAKED *153 MILLION* ADOBE USER EMAILS, ENCRYPTED PASSWORDS, AND PASSWORD HINTS.
ADOBE ENCRYPTED THE PASSWORDS IMPROPERLY, MISUSING BLOCK-MODE 3DES. THE RESULT IS SOMETHING WONDERFUL:
THE GREATEST CROSSWORD PUZZLE IN THE HISTORY OF THE WORLD

USER PASSWORD		HINT
4e18acc1ab27a2d6		WEATHER VANE SWORD
4e18acc1ab27a2d6		
4e18acc1ab27a2d6	a0a2876eb1ea1fca	NAME1
8babb6299e06eb6d		DUH
8babb6299e06eb6d	a0a2876eb1ea1fca	
8babb6299e06eb6d	85e9da81a8a78adc	57
4e18acc1ab27a2d6		FAVORITE OF 12 APOSTLES
1ab29ae86da6e5ca	7a2d6a0a2876eb1e	WITH YOUR OWN HAND YOU HAVE DONE ALL THIS
a1f96266299e7a2b	eadec1e6ab797397	SEXY EARLOBES
a1f96266299e7a2b	617ab0277727ad85	BEST TOS EPISODE
3973867adb068af7	617ab0277727ad85	SUGARLAND
1ab29ae86da6e5ca		NAME + JERSEY #
877ab7889d3862b1		ALPHA
877ab7889d3862b1		
877ab7889d3862b1		
877ab7889d3862b1		OBVIOUS
877ab7889d3862b1		MICHAEL JACKSON
38a7c9279cadeb44	9dca1d79d4dec6d5	
38a7c9279cadeb44	9dca1d79d4dec6d5	HE DID THE MASH, HE DID THE
38a7c9279cadeb44		PURLOINED
a8ae5745a2b7af7a	9dca1d79d4dec6d5	FAV WATER-3 POKEMON

https://xkcd.com/1286/

Mistake Two

Engineers have taken secure cipher suites but incorrectly implemented (part of) the code

- Bad public-key parameters chosen
- The wrong block cipher modes chosen
- Uncounting counters
- Reused nonces (Hint: nonce = n_{once})
- Returning private keys as public
- Allowing users too much control in parameters
- ECC is secure, but not all curves are!

Secrets need to be *secret*

This leaves us with the most subtle of the three mistakes

Mistake Three

Secret information has somehow been leaked

- Post-it notes!
- Bad randomness
- Secrets left in Git repos (or in HTML source code[2])
- Side-channel attacks

[2] https://twitter.com/GovParsonMO/status/1448697768311132160

Questions?

Thank you for listening

Questions?

CTF Resources

- CTFtime lists most upcoming CTFs and keeps track of scores. The more you win, the higher your teams global rank is! `https://ctftime.org`
- Huge list of resources: `https://zaratec.github.io/ctf-practice/`

My favourites:

- PicoCTF is a beginner's CTF which has a bunch of permanent challenges, as well as yearly competitions `https://picoctf.org`
- Subject specific:
  - ⋆ Cryptography `https://cryptohack.org`
  - ⋆ Pwn `https://pwn.college`
  - ⋆ Web `www.pentesterlab.com`
  - ⋆ Reversing `http://reversing.kr/`