Computer Network Security

COMP 178 | Spring 2025 | University of the Pacific | Jeff Shafer

Cryptography:
Intro, One Time Pads, Block Ciphers

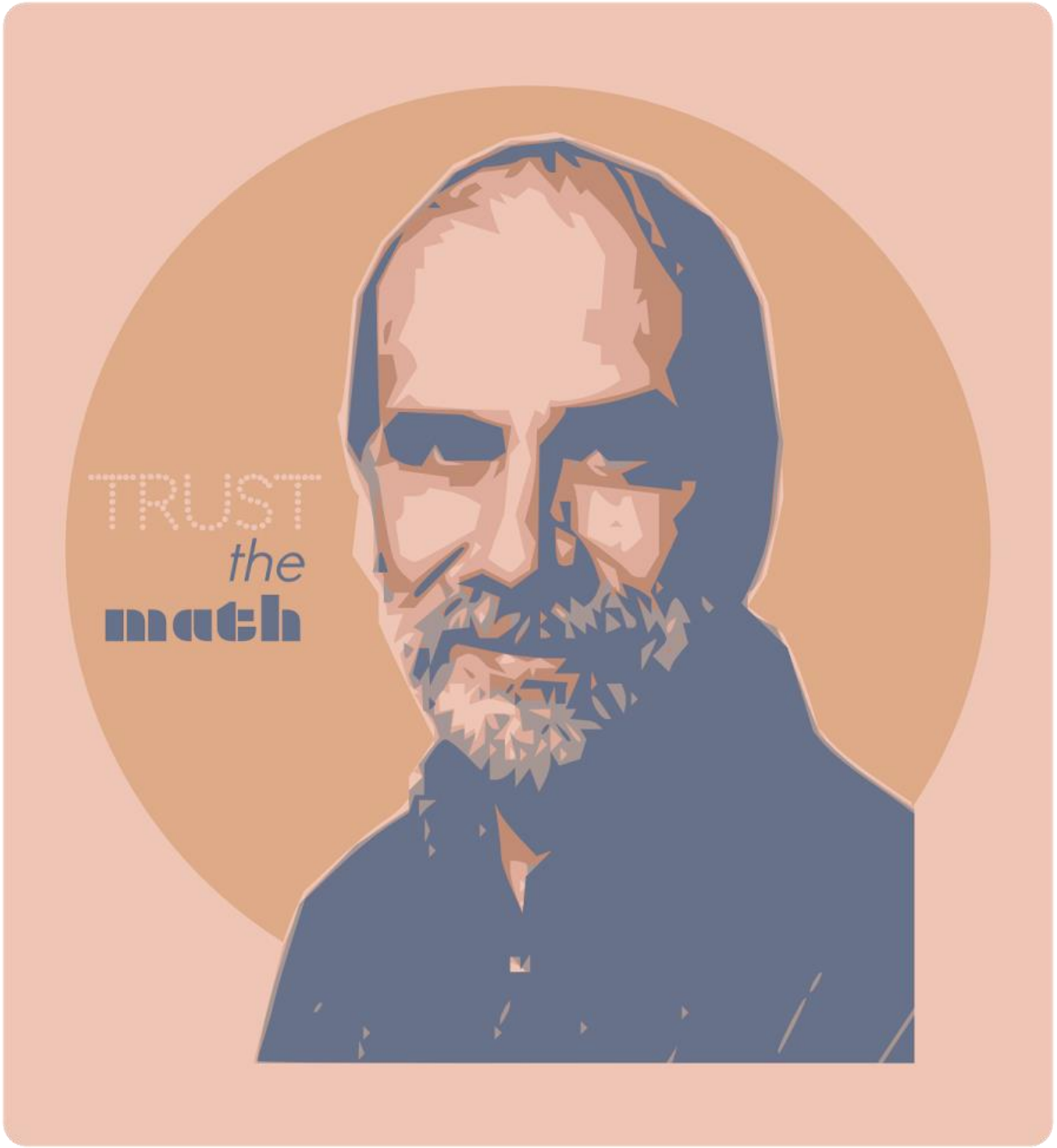
Let’s talk about cryptography

An hour ought to do it, right?


Cryptography is hard
Cryptography is hard
Cryptography is (very) hard

If it's **good encryption**,
it should look like **random noise**

But just because it **looks** like random noise doesn't mean it's **good** encryption

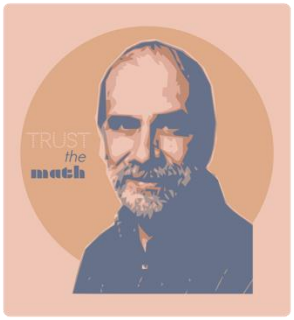


“Trust the Math”

~ Bruce Schneier

https://www.schneier.com/

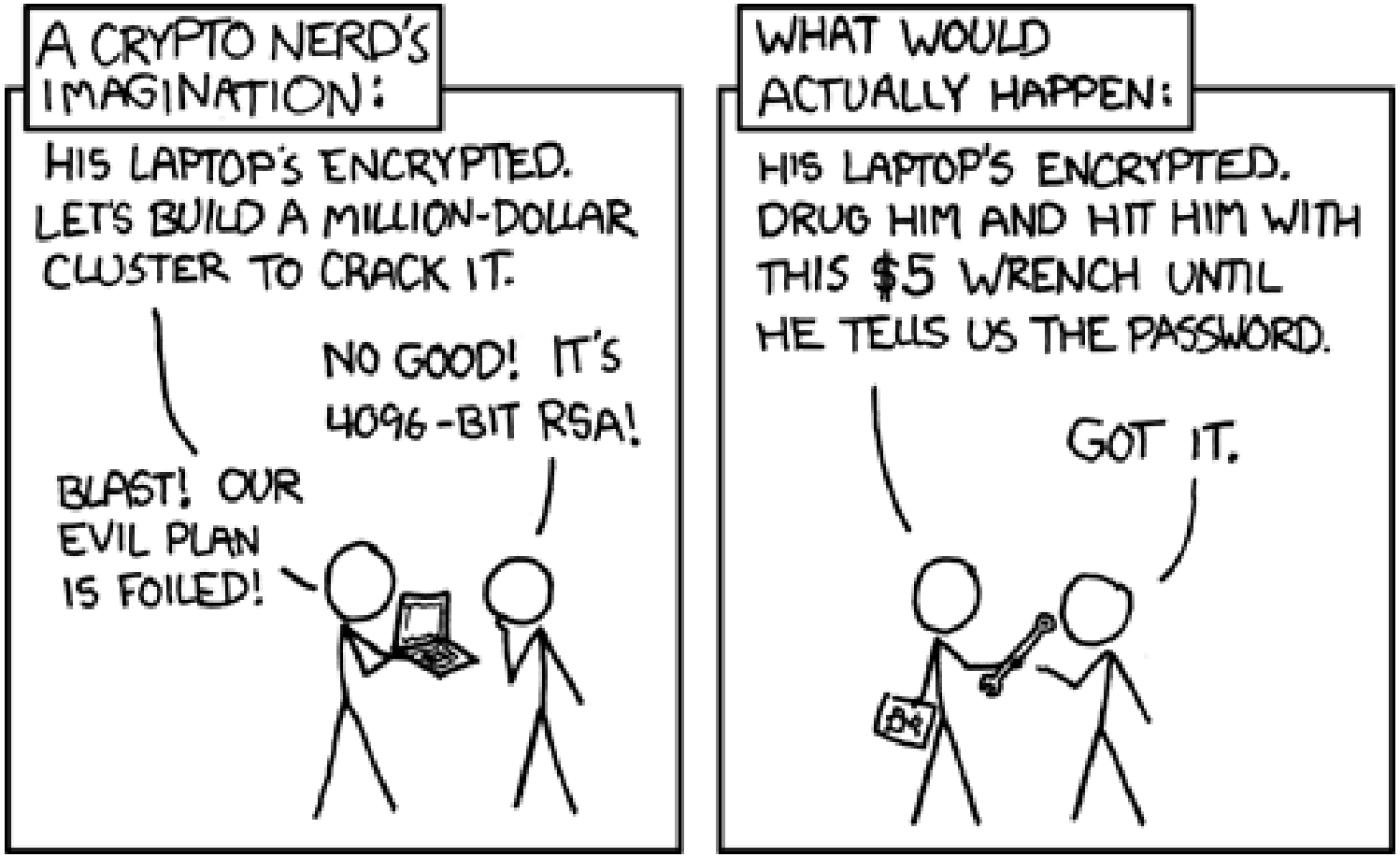
Cryptography is Harder Than It Looks



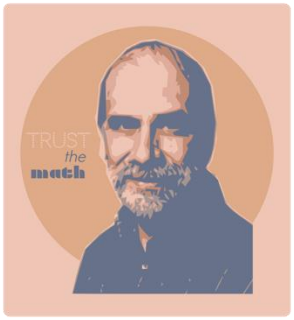
- ***"Cryptography is harder than it looks"***
  - It looks like math, and we have smart mathematicians, so problem solved, right?
  - Problem: Math equations can't secure anything
    - Write equations into software
    - Embed in a larger software system
    - Manage by an OS
    - Run on hardware
    - Connect to a network
    - Configure and operate by users
  - **Commonly find vulnerabilities *not* in underlying mathematics but in the implementation**

https://www.schneier.com/blog/archives/2016/03/cryptography_is.html

Cryptography is Harder Than It Looks



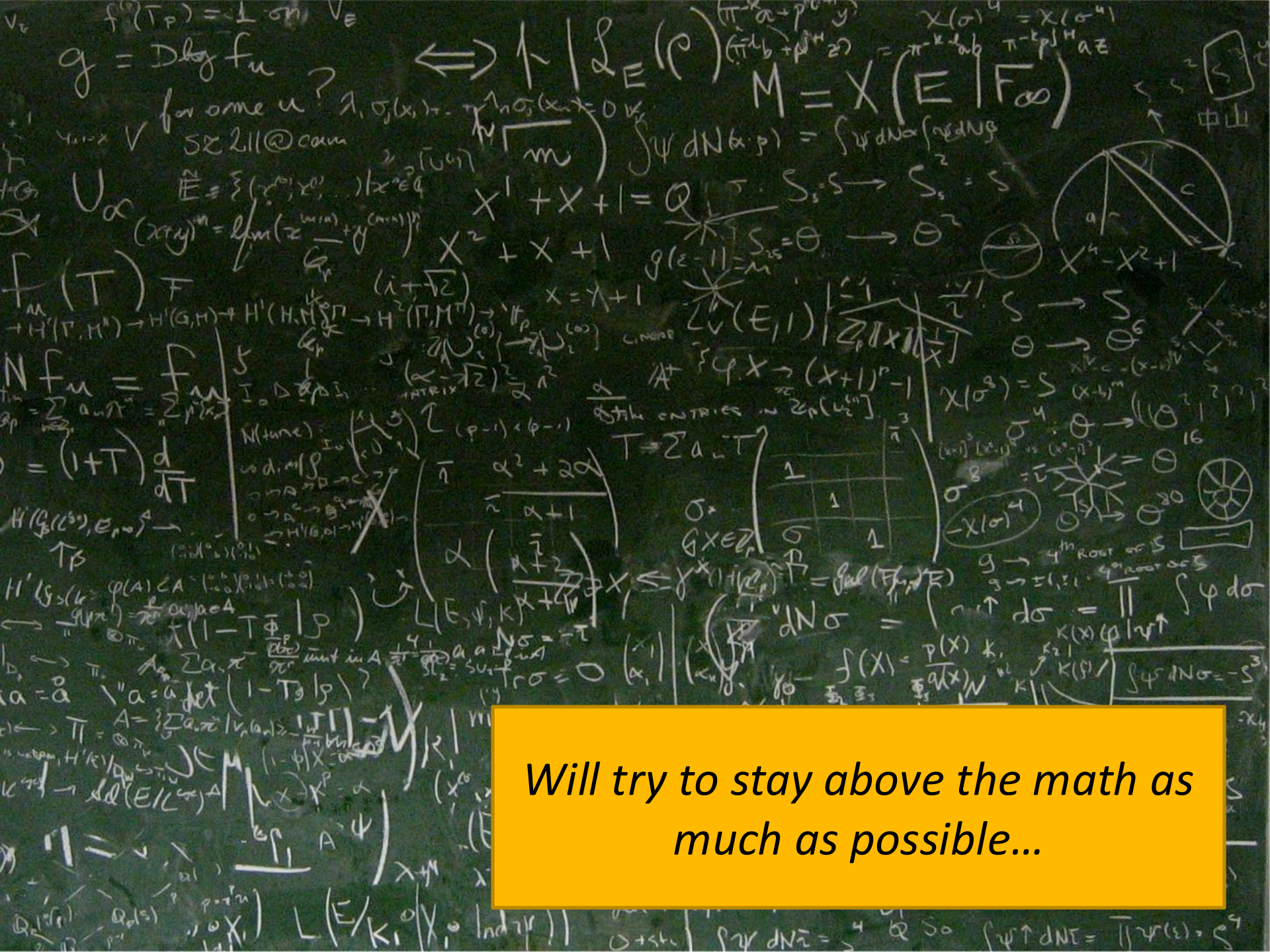
Cryptography is Harder Than It Looks



- ***"Complexity is the worst enemy of security"***
  - More lines of code
  - More interactions with other systems
  - More configuration options
  - **Result: More vulnerabilities!**

https://www.schneier.com/blog/archives/2016/03/cryptography_is.html

Cryptography


Will try to stay above the math as much as possible…

Useful References on Class Website

Cryptography

- **Crypto 101 e-book** [**Free/PDF**, (2017+)]
  - Written by Laurens Van Houtven and distributed at https://www.crypto101.io/. Updated directly at GitHub repo. (See also: Local mirror, may be out of date compared to github copy)
  - Big picture ideas + Commentary on breaking crypto. Light on the math! Incomplete / work in progress.
- **Cryptography: An Introduction (3rd Edition)** [**Free/PDF**, 2006]
  - Written by Nigel Smart, a cryptographer & computer science professor specializing in the area of elliptic curve cryptography.
  - Note: The free e-book is no longer being updated as-of 2016. See **Cryptography Made Simple** (2016, Springer Publishing) for purchase.
  - Heavy on the math!
- **Cryptography Engineering** [Paid, 2010]
  - Written by Niels Ferguson (Cryptographer, Microsoft), Bruce Schneier (Cryptographer), and Tadayoshi Kohno (Professor, University of Washington)

CRYPTO101

Crypto 101 is an introductory course on cryptography, freely available for programmers of all ages and skill levels.

Get current version (PDF) Tweet

Start to finish.

Comes with everything you need to understand complete systems such as SSL/TLS: block ciphers, stream ciphers, hash functions, message authentication codes, public key encryption, key agreement protocols, and signature algorithms.

Learn by doing.

Learn how to exploit common cryptographic flaws, armed with nothing but a little time and your favorite programming language.

Forge administrator cookies, recover passwords, and even backdoor your own random number generator.

Works everywhere.

DRM-free and available in all common formats:

- PDF (for Mac and PC)
- ~~EPUB (for most ebook readers, iPad and iPhone)~~
- ~~Mobi (for Kindle)~~

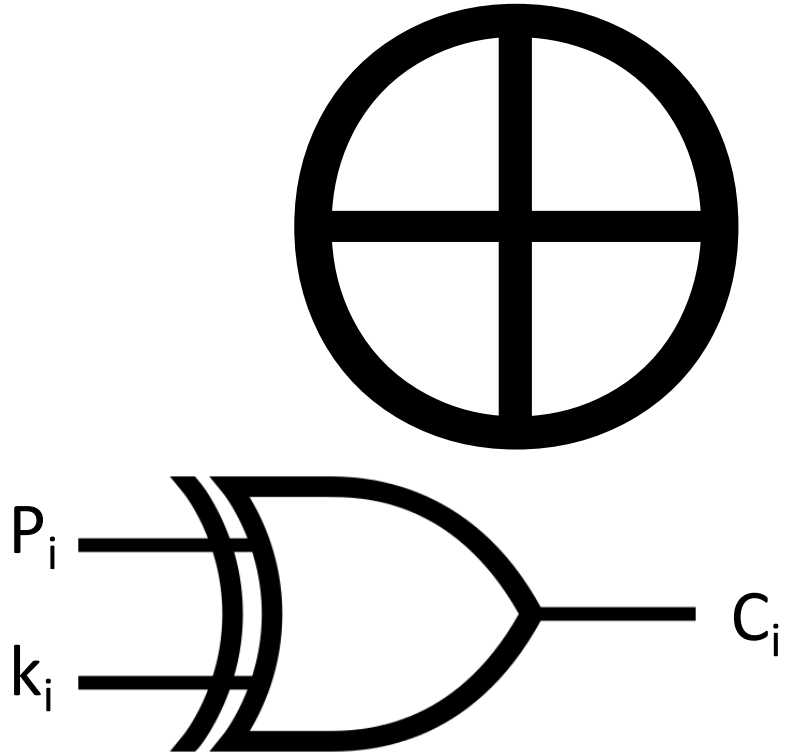
https://www.crypto101.io/

Has a video too!

https://www.youtube.com/watch?v=3rmCGsCYJF8

XOR & One Time Pads

XOR



P_i	k_i	C_i
0	0	**0**
0	1	**1**
1	0	**1**
1	1	**0**

P_i = Plaintext (bit i)

C_i = Ciphertext (bit i)

K_i = Key (bit i)

Useful XOR Properties

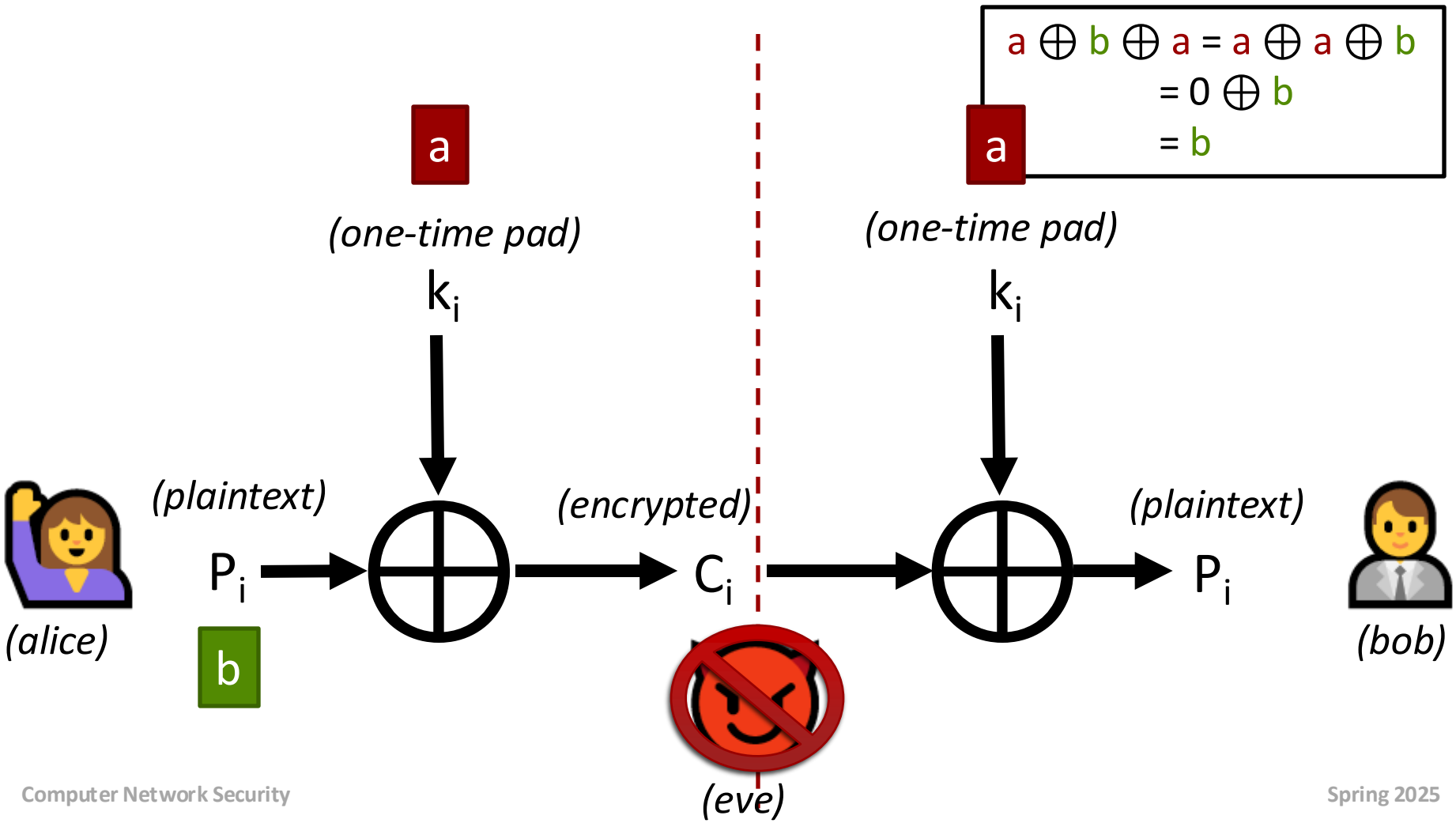
- Associative: Apply XOR in any order:
  $a \oplus (b \oplus c) = (a \oplus b) \oplus c$
- Commutative: Flip operands around:
  $a \oplus b = b \oplus a$
- Useful property for encryption:
  $a \oplus b \oplus a = a \oplus a \oplus b$
  $= 0 \oplus b$
  $= b$

One-Time Pad (OTP)

Pencil and paper technique

Dates back to 1882 (Frank Miller, designing for telegraph machines)

One-Time Pad (OTP)



One-Time Pad (OTP)

- Ciphertext provides no information about the original message (beyond length)
  - Thus, OTP has property of **Perfect Secrecy** ☺
  - Eve has no idea if P_i was 0 or 1 - Cannot be broken!
- But, Perfect Secrecy property is **only** true **if:** ☹
  - **….**

One-Time Pad Problems

- One-Time Pad must be used only **once**!
  - Problem: Repeated use allows statistical comparisons – easy to break
- One-Time Pad must be (cryptographically) random
  - Problem: Random number generation is hard/slow
- One-Time Pad must be shared between parties
  - Problem: Sharing must occur in advance of communication, out of band?
  - Both parties must keep pad secret
- One-Time Pad must equal length of plaintext
  - Problem: Length!

Beyond the One-Time Pad

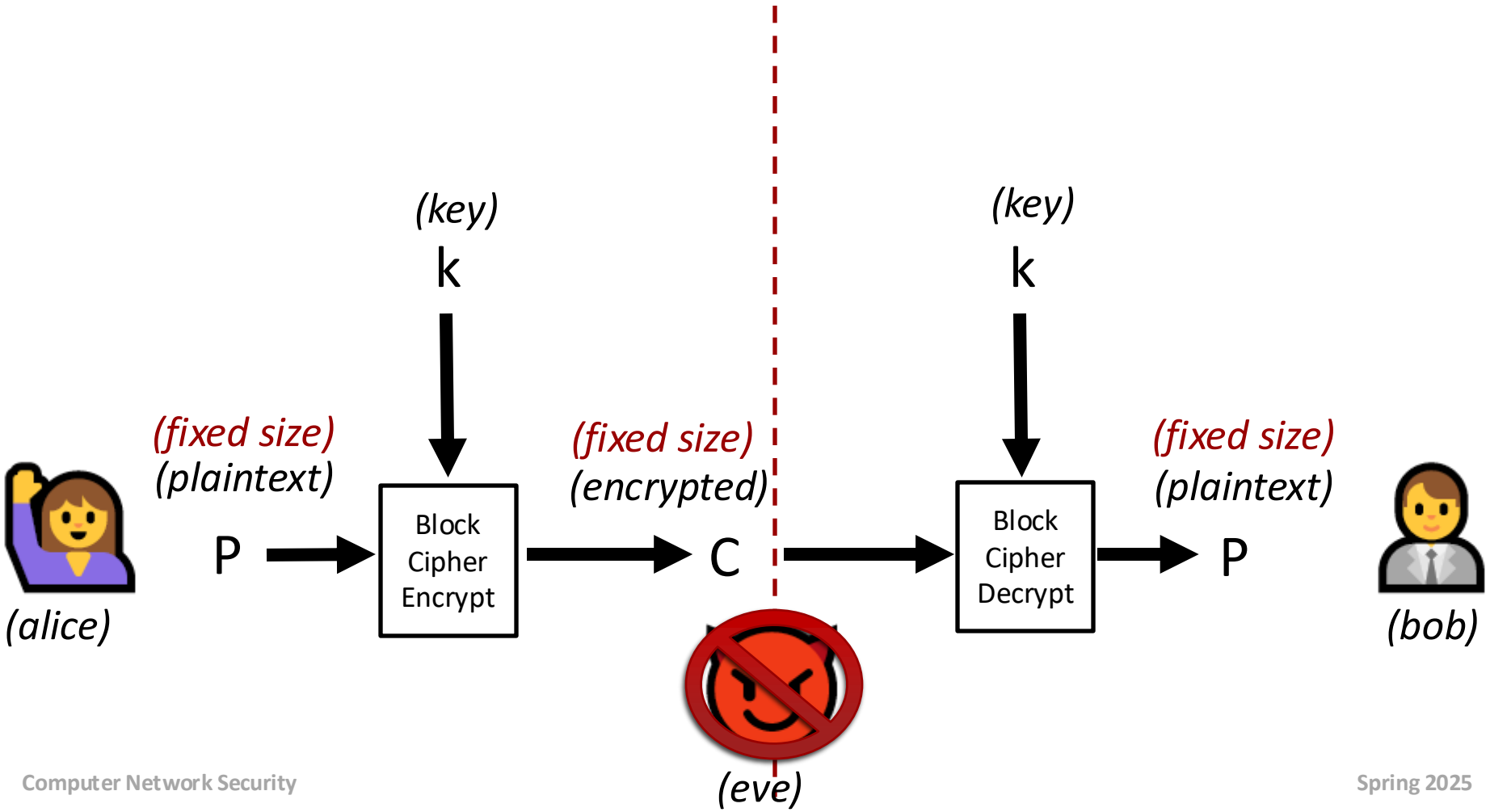
- What do we use instead? **Ciphers!**
  - Manageable key sizes ☺
  - Methods to negotiate keys over the Internet with parties we've never communicated with before ☺
  - Systems are more complex and lack the theoretical Perfect Secrecy property of OTPs ☹

Block Ciphers

Block Cipher

- Encryption function *E*
  C = E(k, P)
- Decryption function *D*
  P = D(k, C)
- **Symmetric-key encryption**
  - Same key is used for both encryption and decryption
- Operates not bit-by-bit but block-by-block
  - Block is a collection of bits
  - Block size is *fixed* by the cipher

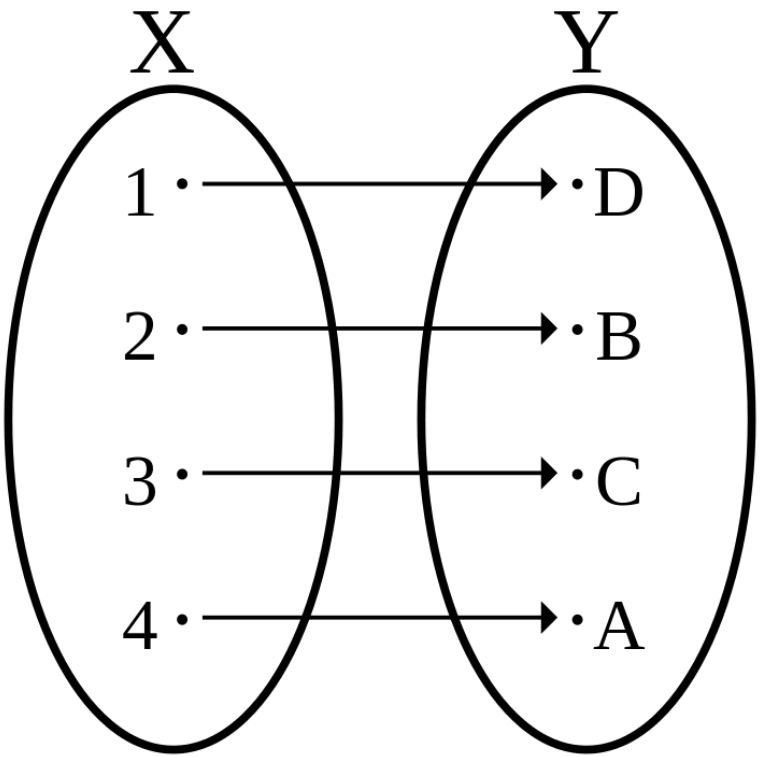
Block Cipher



Block Cipher

- What's in the box?
- **Keyed Permutation**
  - Permutation – every input block is mapped to a unique output block
    - Reversible!
  - Keyed – The key determines the exact mapping
    - *Mapping varies by the key*

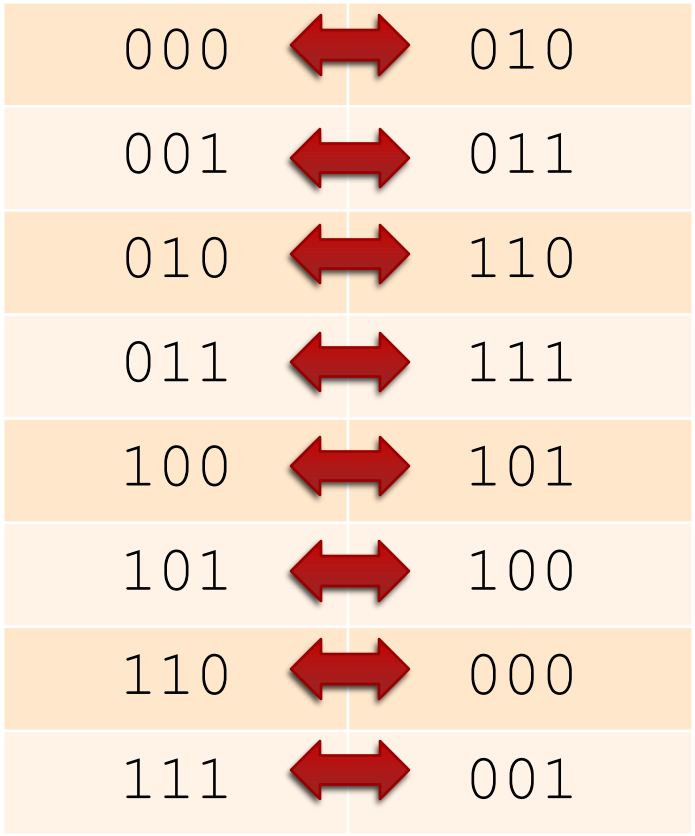
Bijection



- **Bijective function**
- Function between the elements of two sets
  - Each element of set X is paired with exactly one element of set Y
  - Each element of set Y is paired with exactly one element of set X
  - **There are no unpaired elements**

Block Cipher

- **Example Block Cipher**
  - 3 bit block size (tiny!)



Block Cipher

- Assuming no fatal flaws in mapping, brute forcing key is only path for attacker
  - Must try every potential key
  - $2^3 = 8$ possible blocks
  - $2^{128} = 3.4 \times 10^{38}$ possible blocks
- $n!$ permutations of an n element set
  - 128 bit key?
    $(2^{128})!$ permutations

Block Cipher

Pros

- Key are short / fixed length
  - 128-256 bits
- Much easier to communicate to recipient than a One-Time Pad!

Cons

- What happens if the **plaintext is larger than the block size**?
  - Example: AES block size is 128 bits (16 bytes)
- How do the sender and receive agree on the same key over an **insecure channel**?

Block Cipher Examples

- AES – Advanced Encryption Standard
- Blowfish/Twofish/Threefish
- DES/3DES

Block Cipher Examples - AES

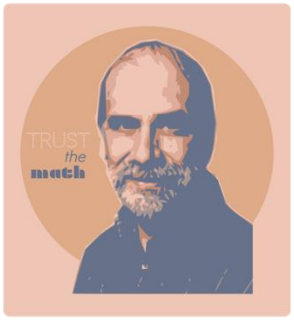
- Public peer-reviewed competition to select next-generation block cipher (AES)
  - Sponsored by NIST, 1997-2001
  - Evaluated on: security, licensing, implementation
- **Security**
  - "The extent to which the algorithm output is indistinguishable from a random permutation on the input block"
  - "Soundness of the mathematical basis for the algorithm's security"
  - "Other security factors **raised by the public** during the evaluation process, including any attacks which demonstrate that the actual security of the algorithm is less than the strength claimed by the submitter"
- **Licensing** (worldwide, non-exclusive, royalty-free)
- **Algorithm / Implementation**
  - Flexibility (e.g., additional key sizes, additional block sizes, stream cipher, …)
  - Computational efficiency / Memory requirements
  - Hardware and software implementation suitability
  - Simplicity ☺

https://competitions.cr.yp.to/aes.html

Block Cipher Examples - AES

- **AES – Advanced Encryption Standard**
  - Winning cipher: Rijndael (pronounced “Rhine Dahl”)
    - Developed by two Belgian cryptographers: Vincent Rijmen and Joan Daemen
    - Block size: 128 bits
    - Key size: 128, 192, 256 bits
  - Standardized: FIPS PUB 197 and ISO/IEC 18033-3

Block Cipher Examples - *Fish



- Three algorithms created by Bruce Schneier
- **Blowfish** (1993) – **Don't use**!
- **Twofish** (1998)
  - One of five finalists in the AES competition
  - "I have nothing but good things to say about NIST and the AES process."
    ~ Bruce Schneier
- **Threefish** (2008)

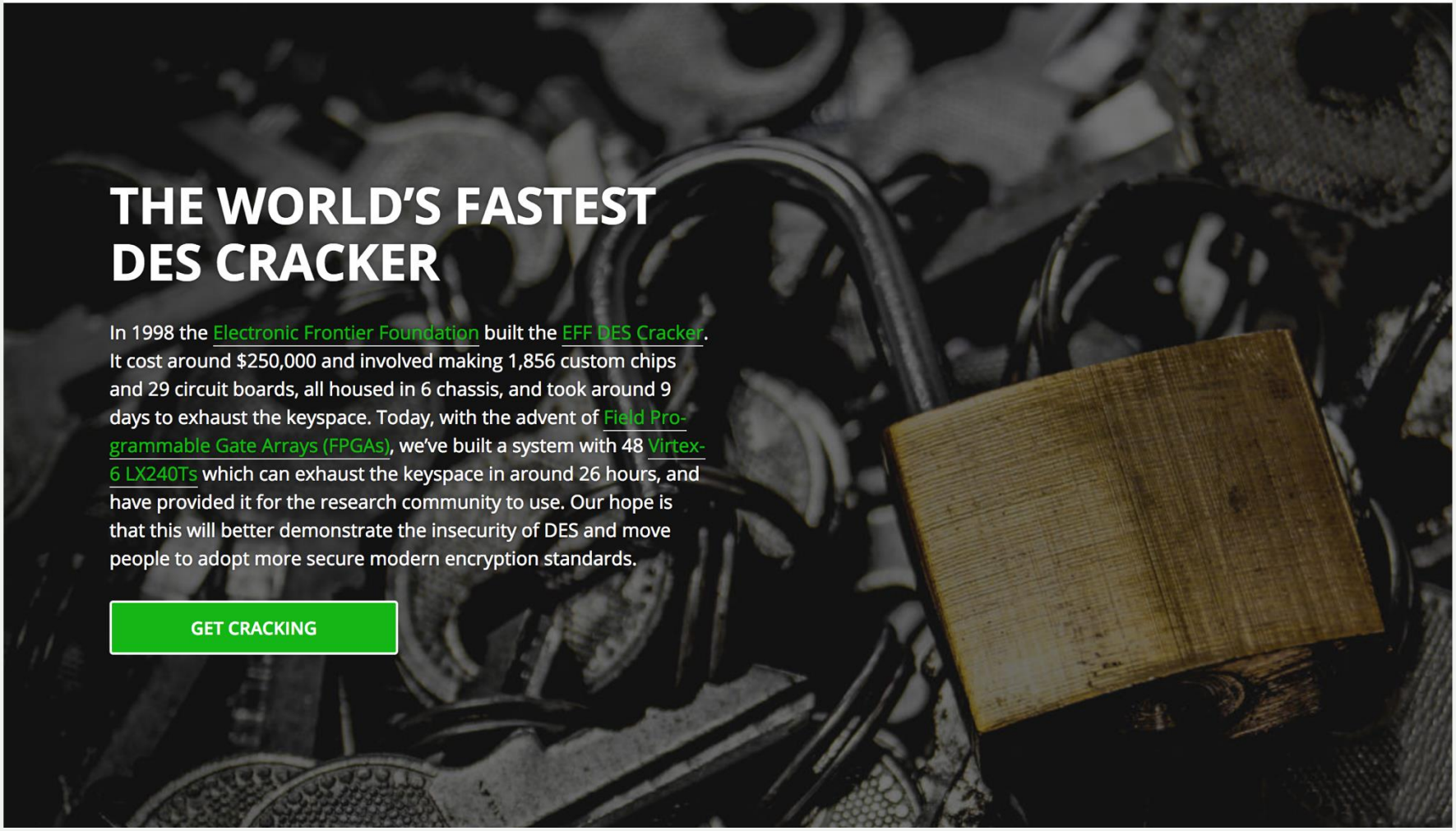
Block Cipher Examples – DES/3DES

- **DES** – Data Encryption Standard – **Don't use**!
  - Designed by IBM in 1975
  - Algorithm is "basically sound" (unless you have a large number of plaintexts to feed through system), but 56 bit key is too small on modern hardware
    - Brute forced in under a day using 1999 hardware
  - Withdrawn as standard by NIST (Replaced by AES)
- **3DES** ("triple DES") – **Don't use**!
  - Run DES three times….
  - Secure, but very very slow (10x worse than AES)
    - Only bother if dealing with legacy systems ☹

Breaking Cryptography (AES/DES)

Breaking Cryptography

- General methods
  - **Brute force**
    - Time / $$$
    - Cryptographic “break” is any attack *faster* than brute force
  - **Attack the cipher** (“Smart mathematicians”)
    - Reduce the search space for brute force attacks
    - Make repeated observations and intelligent guesses about keys
  - **Attack the implementation** (“Smart engineers”)
    - Side channel attacks (Cache timing, power usage, software-initiated attacks like rowhammer, …)



https://crack.sh/

Breaking Crypto – DES with crack.sh

- 48 Xilinx Virtex-6 LX240T FPGAs
  - 40 DES cores
  - 400MHz
  - 16×10^9 keys/sec/FPGA
  - 768×10^9 keys/sec total
- **Exhaustively search 56-bit DES key-space in:**
  2^{56} / 768,000,000,000
  = ~26 hours
- Formats supported (Single-DES)
  - NETLM/NETNTLMv1 Authentication
  - PPTP VPNs
  - WPA-Enterprise
  - des_crypt() Hashes
  - Kerberos5 DES
  - Known Plaintext DES
- Price: $20-$100

Breaking Crypto - AES

- AES cipher has been "broken"
  - Key recovery faster than brute force by a factor of 4
    - 128 bit key now offers security of 126 bits
    - 256 bit key now offers security of 254
  - Good math work, but *yawn…*
    - *Time to brute force before: ∞*
    - *Time to brute force now: ∞ / 4*

Research paper

A. Bogdanov, D. Khovratovich, C. Rechberger, "Biclique Cryptanalysis of the Full AES". IACR Cryptology ePrint Archive. 344-371. 10.1007/978-3-642-25385-0_19, **2011**

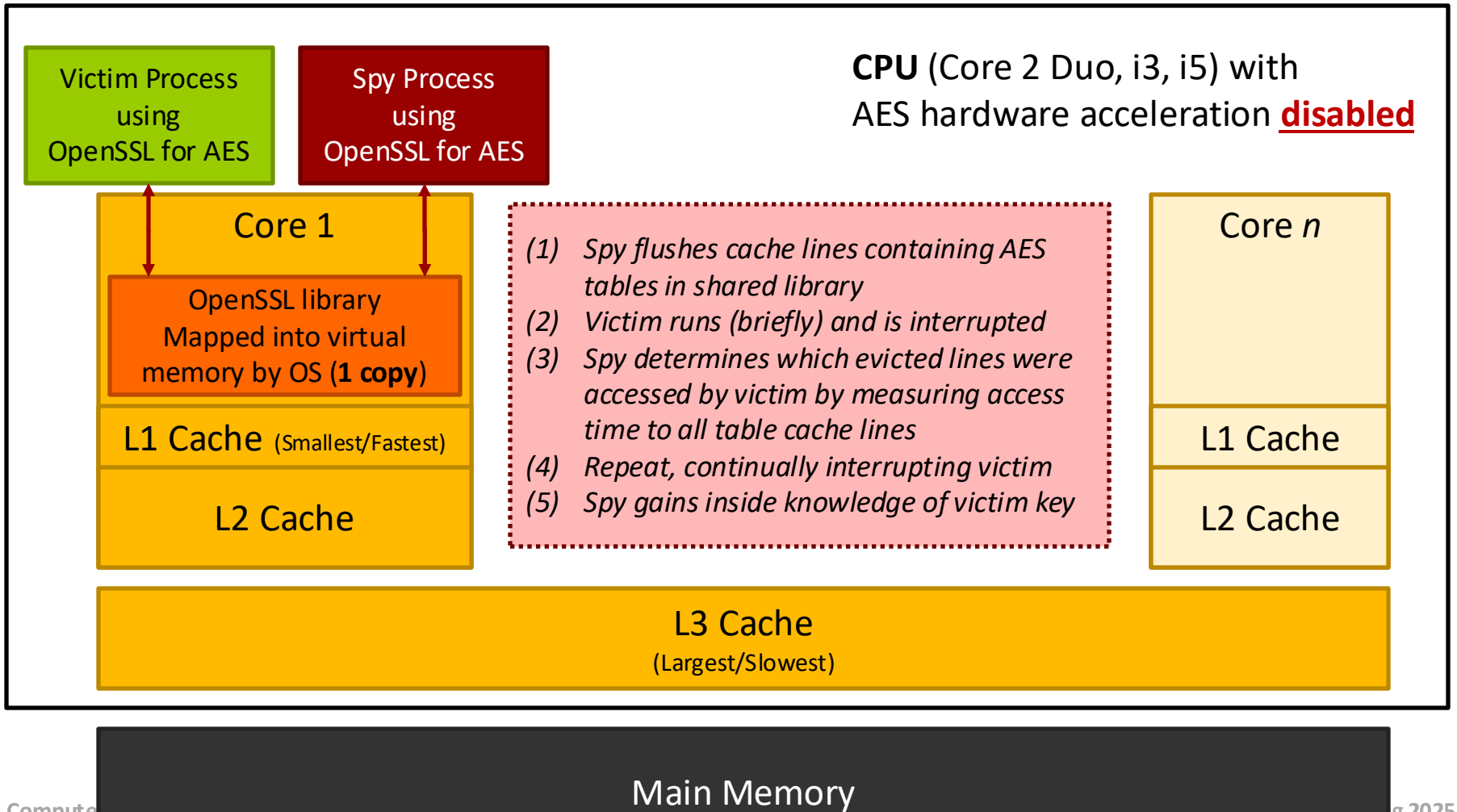
Breaking Crypto - AES

- AES implementation has been “broken”
  - Demonstrated recovery of the full AES key given only about 6 - 7 blocks of plaintext or ciphertext
  - Requires *spy process* running on same machine as AES software. No privilege escalation required.
    - **Side channel attack** via processor cache timing
    - Runs in Python in under a minute!

Research paper

A. C, R. P. Giri and B. Menezes, "Highly Efficient Algorithms for AES Key Retrieval in Cache Access Attacks," *2016 IEEE European Symposium on Security and Privacy (EuroS&P)*, Saarbrucken, **2016**

Side Channel Attack on AES



Countermeasures

- Use hardware AES acceleration
  - Hardware method does not use memory-resident tables for cipher
  - AES-NI instruction set in x86-64
- Use a cipher and implementation that is resistant to timing attacks
  - Idea: All operations should take constant time
    - No conditional branches with conditions based on secret information.
    - All loop counts are predictable in advance
    - No array lookups with indices based on secret information. *The pattern of memory access must be predictable in advance*
  - *Like NaCL library…*

Have I mentioned that writing robust crypto libraries is hard?