Section 5
Public Key Crypto Topics

Calculating RSA, Cryptanalysis, and Crypto Ethics

Eric Zeng, Keanu Vestil
October 29, 2020

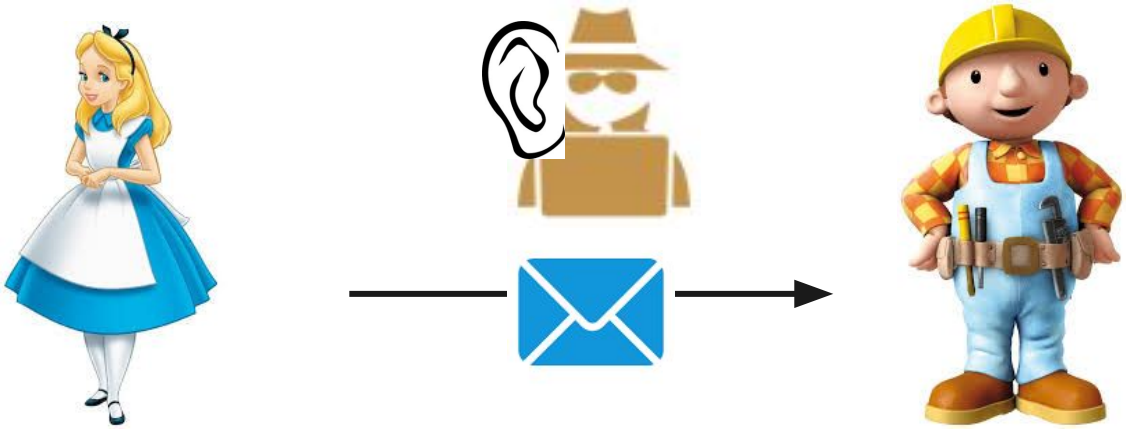
Administrivia

- Lab 1 due date extended to Friday, October 30 @ 11:59pm
  - Please do not make any further changes to files after submitting
- Homework 2 out now - due Friday, November 6 @ 11:59 pm
  - Written exercises about cryptography concepts
- Final project checkpoint #1 - due Friday, November 13 @ 11:59pm
  - Group members' names + NetIDs and brief description of the presentation topic

Calculating RSA:
Key generation, encryption, and decryption

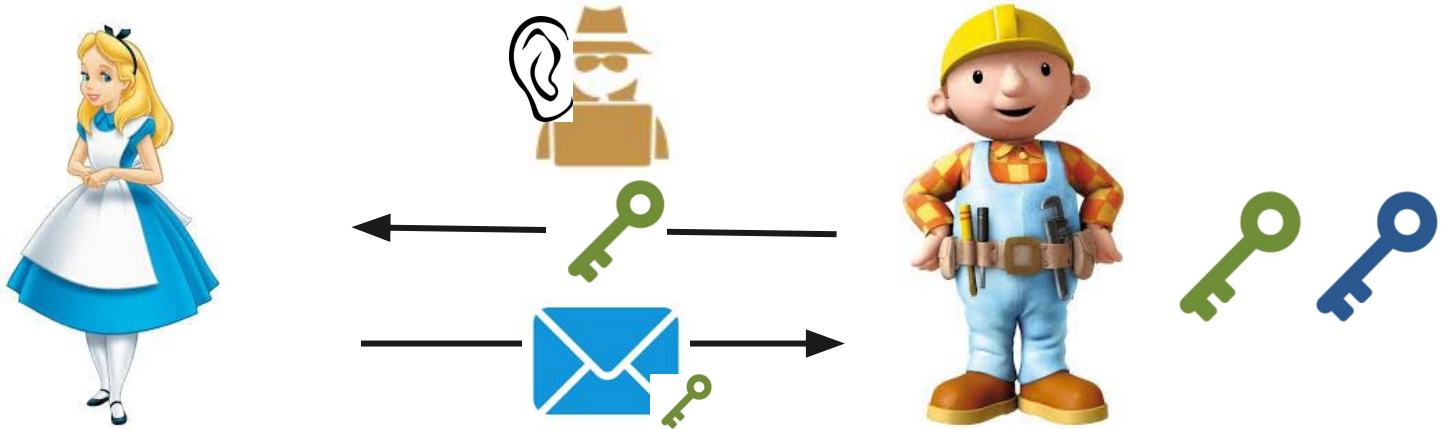
Public Key Cryptography Review

- Scenario: Alice wants to send Bob a message on the internet
  - Goal: confidentiality of data
  - Problem: people eavesdropping on network → can't share **symmetric** keys secretly



Public Key Cryptography Review

- Solution: public key cryptography (aka **asymmetric** cryptography)
  - Bob generates a key pair: one private key (secret), and one public key that is safe to share with anyone
  - Alice encrypts a message using Bob's public key
  - Bob decrypts with Bob's private key



RSA - a public key cryptosystem

RSA can:

- Generate public/private key pairs
- Encrypt plaintext
- Decrypt ciphertext

RSA is based on computing modular exponentiation with large primes

- Easy to compute, hard to reverse (without the private key)

RSA Algorithm Review

Notation:
Public Key: (e, n)
Private Key: d
Message: m

Key Generation

Select two large primes, **p** and **q**

Let **n** = p⋅q

Let **ϕ(n)** = (p -1)(q - 1)

Select a random prime **e** such that **e** and **ϕ(n)** are *relatively prime* (no common factors other than 1)

Compute **d** such that

e⋅d ≡ 1 mod ϕ(n) (equivalent to solving 1 = (e⋅d) mod ϕ(n))

Encrypting m: $c = m^e \bmod n$

***Decrypting c*:** $c^d \bmod n = (m^e)^d \bmod n = m$

RSA Activity (Canvas Quiz)

Q1. Given these RSA parameters: p = 5, q = 7, e = 5

n = p⋅q
ϕ(n) = (p -1)(q - 1)
e⋅d ≡ 1 mod ϕ(n)
1 = (e⋅d) mod ϕ(n)

Encrypt:
$c = m^e \bmod n$

Decrypt:
$c^d \bmod n$
$= (m^e)^d \bmod n$
$= m$

What is n?

What is ϕ(n)?

What is d?

Encrypt 16

Decrypt 12

(https://canvas.uw.edu/courses/1396608/quizzes/1331770)

RSA Activity (Canvas Quiz)

Given these RSA parameters: p = 5, q = 7, e = 5

n = p ▪ q
ϕ(n) = (p -1)(q - 1)
e ▪ d ≡ 1 mod ϕ(n)
1 = (e ▪ d) mod ϕ(n)

Encrypt:
$c = m^e \bmod n$

Decrypt:
$c^d \bmod n$
$= (m^e)^d \bmod n$
$= m$

What is n?

n = 5 ▪ 7 = 35

What is ϕ(n)?

ϕ(n) = (5 - 1)(7 - 1) = 24

What is d?

5 ▪ d ≡ 1 mod 24
1 = (5 ▪ d) mod 24
d = 5

Encrypt 16

$16^5 \bmod 35 = 11$

Decrypt 12

$12^5 \bmod 35 = 17$

How to calculate decryption key on homework without trial/error?

Extended Euclidean Algorithm….

Or WolframAlpha :)

RSA Primitive versus Real Life

- Plain RSA also does not provide integrity
  - Can tamper with encrypted messages

In practice, OAEP is used: instead of encrypting M, encrypt $M \oplus G(r)$; $r \oplus H(M \oplus G(r))$

  - r is random and fresh, G and H are hash functions

Demonstration: finding vulnerabilities in CBC-MAC with cryptanalysis

Is encryption (confidentiality) enough?

Scenario: Franzi wants to send out an email about exam times - and a hacker has learned the encryption key

franzi@cs

“Final!!!
KNE 110
Monday
2:30PM”

AES 128-bit key,
CBC mode

ok

In this case, an adversary doesn’t gain anything important by learning the content of this message.

Is encryption (confidentiality) enough?

But, the attacker could tamper with the message during transmission, and the recipient would not know - so we need to ensure **integrity**

Tampers with message in transit

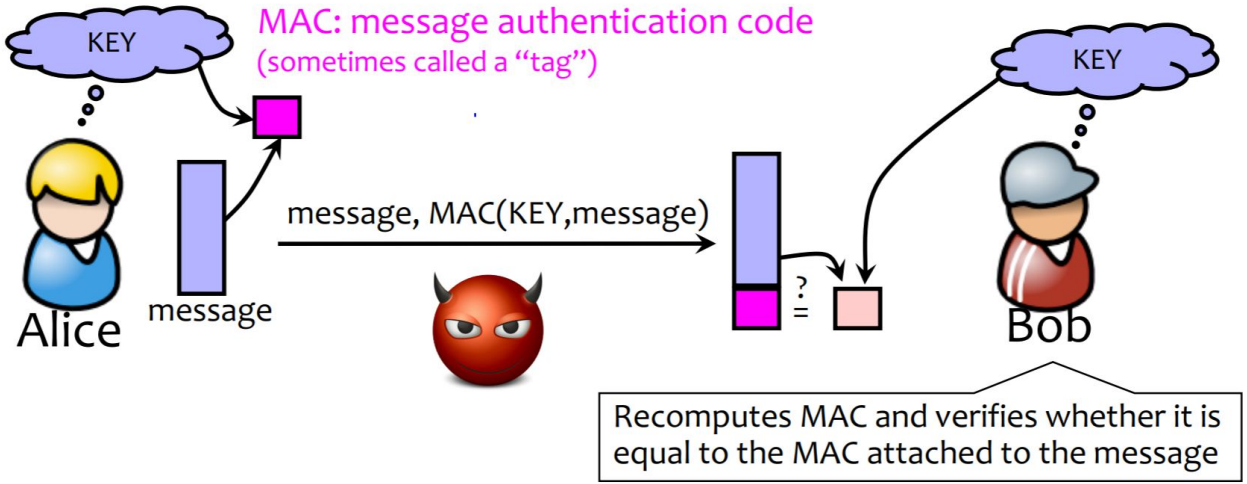
“Final!!!
KNE 110
Saturday
12:30AM”

????

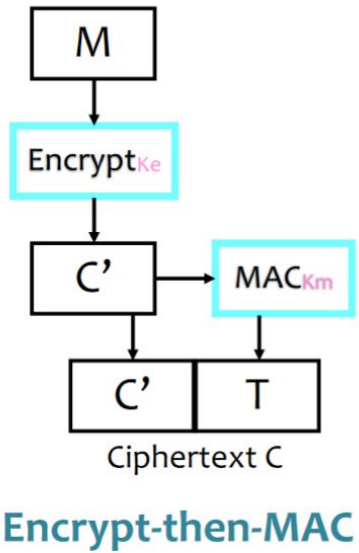
484 Student

MAC (Message Authentication Code)

Provides integrity and authentication: only someone who knows the KEY can compute correct MAC for a given message.



When do we MAC?



Encrypt-then-MAC

The good:

Encrypt-then-MAC

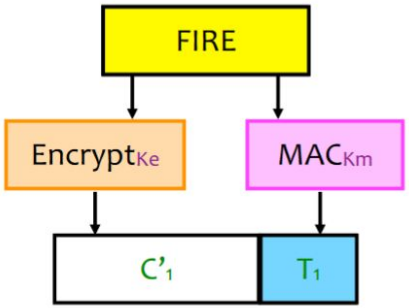
MAC-then-encrypt

Not as good as Encrypt-then-MAC

The bad (& ugly):

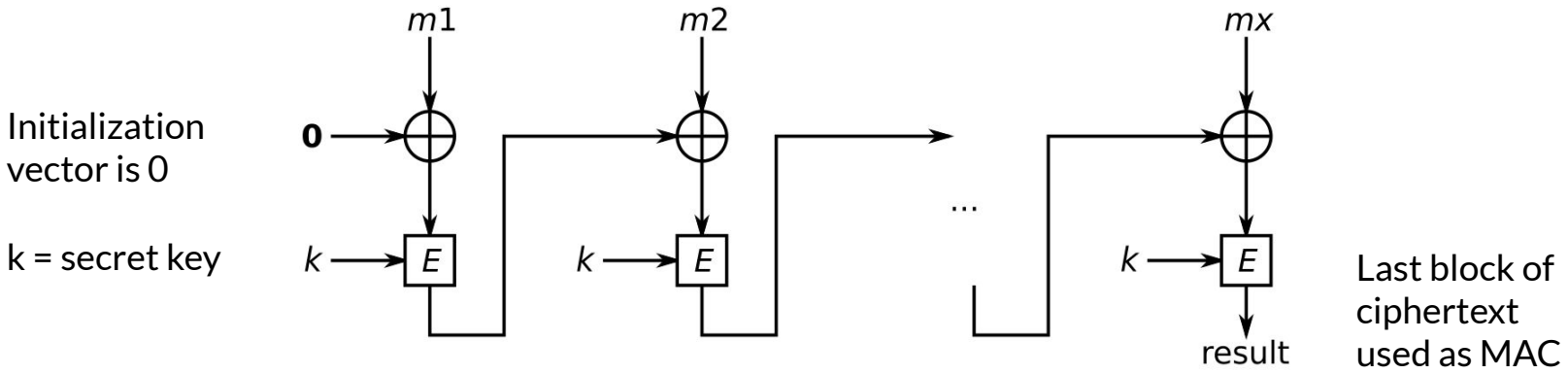
Encrypt-and-MAC

MAC is deterministic! Same plaintext → same MAC



How do we create a MAC?

CBC-MAC: Encrypt the message in CBC mode, use the last block as the MAC



*CBC-MAC is not the only MAC algorithm - today most use HMAC; we'll show why next

Is CBC-MAC vulnerable?

- How could we find out?
  - Cryptanalysis: using mathematical analysis to rigorously reason about a cryptographic system

- Let’s use cryptanalysis to find a collision
  - two different inputs leading to the same MAC tag
  - (violating collision resistance)

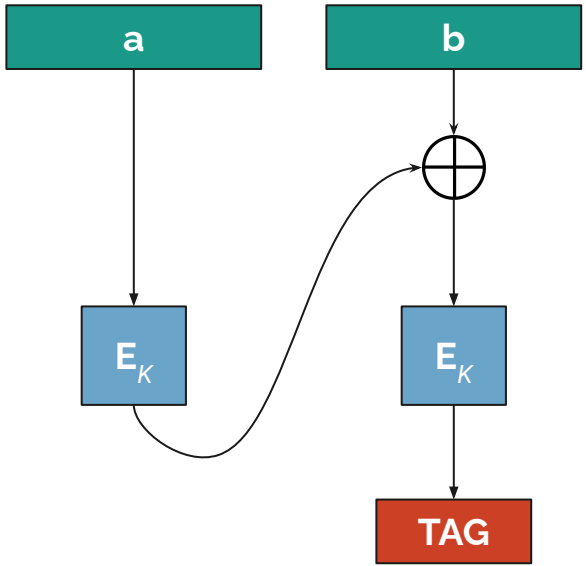
Exercise: CBC-MAC collision vulnerability

Suppose *a* and *b* are both one block long, and suppose the sender MACs *a*, *b*, and *a* || *b* with CBC-MAC.

An attacker who intercepts the MAC tags for these messages can now forge the MAC for the message

$$b \,||\, (M_K(b) \oplus M_K(a) \oplus b)$$

which the sender never sent. The forged tag for this message is equal to $M_K(a \,||\, b)$, the tag for *a* || *b*. Justify mathematically why this is true.



(Ferguson, Schneier, & Kohno. *Cryptography Engineering: Design Principles and Practical Applications*. Wiley Publishing 2010. Exercise 6.3 p. 97)

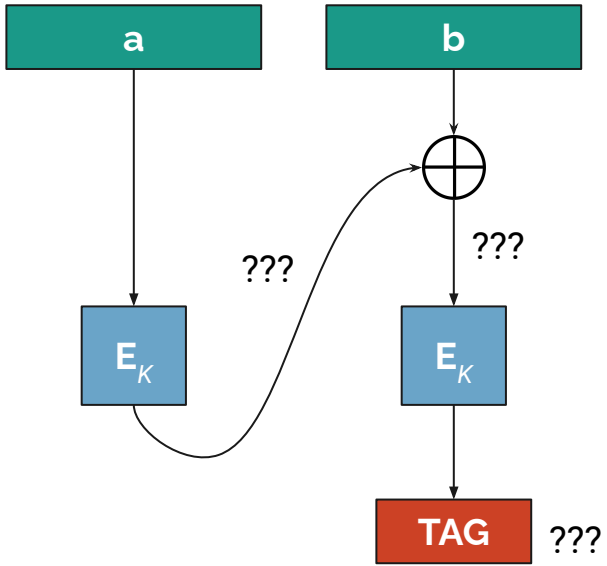
Exercise: CBC-MAC collision vulnerability

Prove:

$$M_K(b \,||\, (M_K(b) \oplus M_K(a) \oplus b)) = M_K(a \,||\, b)$$

Step 1: Figure out what $M_K(a)$, $M_K(b)$, and $M_K(a \,||\, b)$ in terms of the encryption key.

Annotate sketch with the sender's messages and MACs.



(Ferguson, Schneier, & Kohno. *Cryptography Engineering: Design Principles and Practical Applications*. Wiley Publishing 2010. Exercise 6.3 p. 97)

Exercise: CBC-MAC collision vulnerability

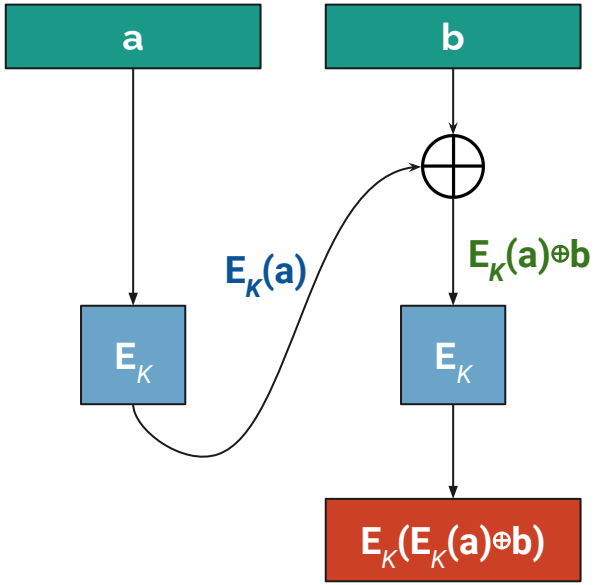
Prove:

$$M_K(b \,||\, (M_K(b) \oplus M_K(a) \oplus b)) = M_K(a \,||\, b)$$

$M_K(a) = E_K(a)$

$M_K(b) = E_K(b)$ (not shown)

$M_K(a \,||\, b) = E_K(E_K(a) \oplus b)$



(Ferguson, Schneier, & Kohno. *Cryptography Engineering: Design Principles and Practical Applications*. Wiley Publishing 2010. Exercise 6.3 p. 97)

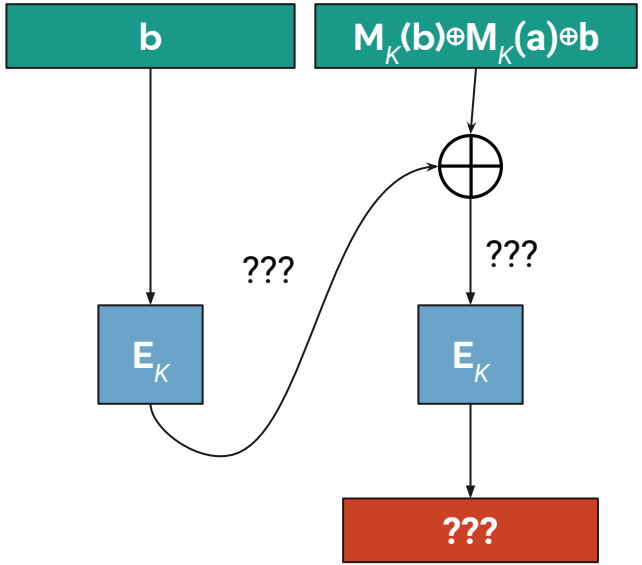
Exercise: CBC-MAC collision vulnerability

Prove:

$$\boldsymbol{M_K(b \,||\, (M_K(b) \oplus M_K(a) \oplus b)) = M_K(a \,||\, b)}$$

Step 2: Figure out $M_K(b \,||\, (M_K(b) \oplus M_K(a) \oplus b))$.

For the MAC of the attacker's message
$b \,||\, (M_K(b) \oplus M_K(a) \oplus b)$, what are the values of the ???'s?



(Ferguson, Schneier, & Kohno. *Cryptography Engineering: Design Principles and Practical Applications*. Wiley Publishing 2010. Exercise 6.3 p. 97)

Exercise: CBC-MAC collision vulnerability

Prove:

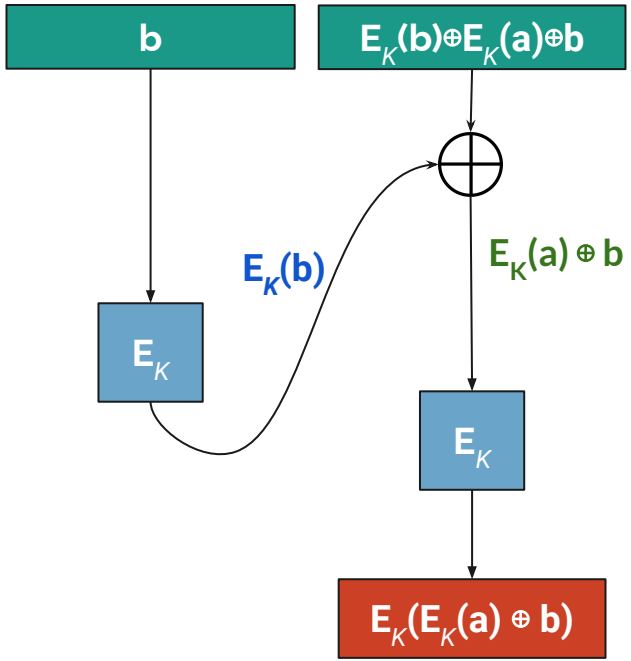
$$M_K(b \,||\, (M_K(b) \oplus M_K(a) \oplus b)) = M_K(a \,||\, b)$$

$M_K(b \,||\, (M_K(b) \oplus M_K(a) \oplus b))$

$= M_K(b \,||\, (E_K(b) \oplus E_K(a) \oplus b))$

$= E_K(E_K(b) \oplus E_K(b) \oplus E_K(a) \oplus b)$ ← These terms cancel out

$= E_K(E_K(a) \oplus b)$ ← *This is the same as $M_K(a \,||\, b)$!*



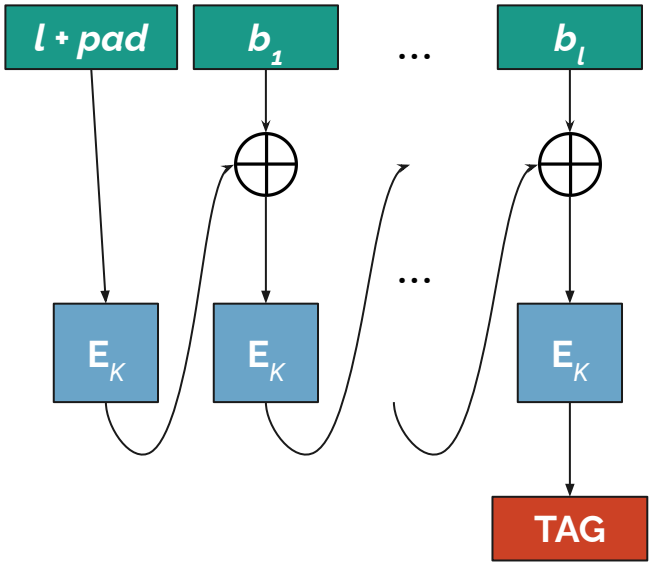
So what?

- We can prove, just using the specification of CBC-MAC, that the messages $b \,||\, (M(b) \oplus M(a) \oplus b)$ and $a \,||\, b$ share the same tag. This approach is a common method used in cryptanalysis.
- We broke the *theoretical* guarantee that no two different messages will never share a tag.
- If you were to use CBC-MAC in a protocol, it provides information about specific weaknesses and how not to use it.

Safer CBC-MAC for variable length messages

For a message *m* of length *l*:

1. Construct *s* by prepending the length of *m* to the message: *s = concat(l, m)*
2. Pad *s* until the length is a multiple of the block size
3. Apply CBC-MAC to the padded string *s*.
4. Output the last ciphertext block, or a part of it. Don't output intermediates.



- *Warning*: Appending to end is just as broken as what we showed!
- Or encrypt output with another block cipher under a different key (CMAC). Or use HMAC, UMAC, GMAC.
- Follow latest guidance very carefully!

Crypto and Ethics

What role does cryptography play in society?

What is the purpose of cryptography?

- What do we use tools like public key encryption, symmetric encryption, secure hashes for?

- Securing computer systems and communications
  - Network security: HTTPS, SSH
  - Authentication: passwords, U2F security keys

Who benefits from cryptography?

Whose problems and threat models are these cryptographic tools addressing?


Industry

- Protecting infrastructure, data, and customers
- Threats and adversaries: unauthorized access, by “hackers” or otherwise

Government

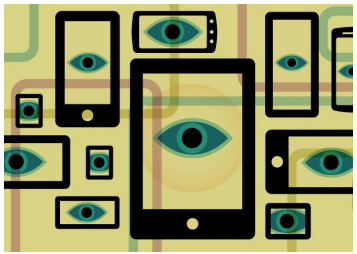
- National security: military, intelligence, diplomatic, intergovernmental communications and operations
- Threats and adversaries: hacking, spying, and cyberattacks by unfriendly countries, other “threats to national security”

What about ordinary people?

- Ordinary people can indirectly benefit when the products we use are secure - e.g. our credit cards on Amazon
- But is cryptography addressing our threat models? Does it benefit us?

Mass Surveillance?

Online Privacy?



What about marginalized people?

- Does cryptography address the needs and threat models of marginalized people?

Computer-amplified racial profiling of African Americans by police


You may be in California's gang database and not even know it

By Ali Winston / March 23, 2016

Harassment and assault (online and physical) of women and LGBTQ people


Tennessee teen's suicide highlights dangers of anti-LGBTQ bullying

Channing Smith's death by suicide after being cyberbullied highlights the decade-old movement to stop bullying online, where it flourishes.

Ethics Activity - Who Benefits from Crypto?

Q2. Pick a system, piece of software, or tool that uses cryptography, and reflect on the following questions:

- Who benefits from this software?
- Who might be harmed by this software?
- Who might be excluded or underserved by this software?

Possible systems:

- Bitcoin
- Signal (private messenger)
- iPhone encryption backdoor
- Tor
- Google Password Checkup tool

(https://canvas.uw.edu/courses/1396608/quizzes/1331770)

Cryptography is not neutral

- Where governments, companies, cryptographers, and software engineers choose to prioritize the development and deployment of cryptography is both a *political* and *moral* choice

As individuals, we have ***agency***

- Academic cryptographers - Who are you building new crypto for? Industry? Governments? People?
- Software engineers - What systems (with crypto) are you willing to work on? Who benefits? How will it affect society?

Further Reading on Crypto and Ethics

Crypto for the People - Seny Kamara

- Talk: https://www.youtube.com/watch?v=Ygq9ci0GFhA
- https://www.wired.com/story/seny-kamara-crypto-encryption-underserved-communities/
- Seny will be guest lecturing in 484 on Dec 2

The Moral Character of Cryptographic Work - Philip Rogaway

- https://web.cs.ucdavis.edu/~rogaway/papers/moral.pdf