Cryptography (5830)

Jr qrirybc cvbarrevat yrnqref sbe gur qvtvgny ntr

Cryptography: “Hidden writing”

- Study and practice of building security protocols that resist adversarial behavior
- Blend of mathematics, engineering, computer science

Cryptography use cases

Crypto is foundational

Failures highlight dependency on good crypto:

- WWII
- Government sabotage
- Playstation 3 crack
- Wireless keys for cars
- Password cracking
- WEP attacks
- Many TLS vulnerabilities
- …

Google:
"report on smartphone encryption and public safety"

Google:
"keys under doormats"

Symmetric encryption

- Symmetric = secret key shared between sender and recipient
  - Functionality (correctness)
  - Security
    - Capabilities of attacker
    - Goals

Auguste Kerckhoffs' (Second) Principle

(circa 1883)

"The system must not require secrecy and can be stolen by the enemy without causing trouble"

A cryptosystem should be secure even if its algorithms, implementations, configuration, etc. is made public --- the only secret should be a key

Why?

Some attacker capabilities

- **Unknown plaintext**
  - attacker only sees ciphertext(s)
- **Known plaintext**
  - attacker knows some plaintext-ciphertext pairs
- **Chosen plaintext**
  - attacker can choose some plaintexts and receive encryptions of them
- **Chosen ciphertext**
  - Attacker can see encryptions of chosen plaintexts and decryptions of chosen ciphertexts
- **Side-channels** such as timing, length, partial bits of keys, knowledge of bad key distribution, etc.

Attacker goals?

Shannon's security notion (1949)

Def. A symmetric encryption scheme E is perfectly secure if for all messages M,M' and ciphertexts C

$$\Pr[\, E(K,M) = C \,] = \Pr[\, E(K,M') = C \,]$$

where probabilities are over choice of K

In words:
each message is equally likely to map to a given ciphertext

In other words:
seeing a ciphertext leaks nothing about what message was encrypted

Does a substitution cipher meet this definition? No!

One-time pads

Fix some message length L

Key generation: output random bit string K of length L

$E(K,M) = M \oplus K$ $D(K,C) = C \oplus K$

Shannon's security notion (1949)

Def. A symmetric encryption scheme is perfectly secure if for all messages M,M' and ciphertexts C

$$\Pr[\, E(K,M) = C \,] = \Pr[\, E(K,M') = C \,]$$

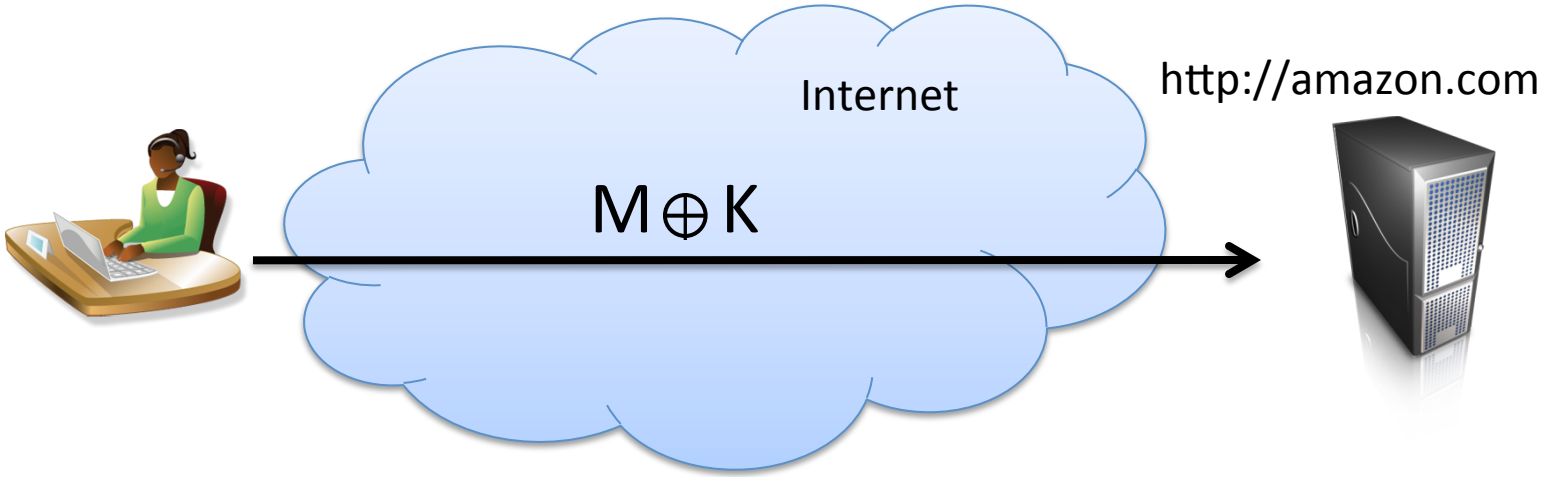
where probabilities are over choice of K

Thm. OTP is perfectly secure

For any C and M of length L bits

$$\Pr[\, K \oplus M = C \,] = 1 / 2^L$$

$$\Pr[\, K \oplus M = C \,] = \Pr[\, K \oplus M' = C \,]$$



Does OTP provide a secure channel?

Integrity easily violated

Reuse of K for messages M,M' leaks M ⊕ M'

Encrypting same message twice under K leaks the message equality

K must be as large as message

Message length revealed

Security notions are hard to get right

Some basic primitives

- Symmetric cryptography (shared key K)
  - encryption & decryption using K
  - message authentication using K
  - pseudorandom functions (PRF)
- Public-key cryptography (public key pk, secret key sk)
  - encrypt with pk and decrypt with sk
  - digitally sign using sk and verify with pk
- Hash functions (no keys)
  - used to “compress” messages in a secure way

The cryptography stack

Abstract tools

Length-preserving Encryption

Length-extending Encryption

Password hashing

TLS
SSH

PGP

Cryptanalysis

Constructions

FFX, EME

AES-GCM, OCB, CTR mode + CMAC

CBC mode, CTR mode, ECB mode

PBKDF2

RSA-OAEP

RSA-PKCS#1 v1.5

Signed Diffie-Helman key exchange

Cryptanalysis
Security reductions
"provable security"

Basic primitives

AES SHA256 RSA Elliptic curve groups

Cryptanalysis

Some goals for course

- **Learning to speak crypto**
  - What primitives are generally for
  - What are the security targets associated with them
- **Current best practices** for cryptographic constructions you are likely to encounter
  - Understand why they are considered best
  - Know how to break some inferior choices
- You should be able to **build crypto libraries** exposing clean, easy-to-use interfaces
  - Using someone else's implementations of primitives

Administrative

- Some details at https://github.com/cornelltech/CS5830-Spring16
- Big things:
  - Homework assignment (Python coding)
  - Final of some sort (Details TBA)
  - Participation
  - Projects
- TA: Rahul Chatterjee

Projects

- Non-trivial crypto feature implemented
  - For your startup project
  - Other group's startup project (need to sort IP)
  - Social good project (HRW, ACLU)

- Solo or in small teams (2 or 3)
  - Initial proposals will be due to me Feb 23, in-class pitches Feb 25

Homework for next week

- Give 1 or more examples of cryptography use in your company challenge projects from last semester
  - Could be code you touched, could not be
  - Extra credit points if I've never heard of the use case
  - Double extra credit points if you explain a meaningful, non-trivial attack
- Prepare a few sentence description, discussing application, security goals, pointer to (open source) code
  - Tuesday we will have some people present