Instructor: David Wie (dweit@cs.utexas.edu) TA: TBD

Overarching goal of cryptography: securing communication over untrusted networks

Alice ----> Bob

third porty should not be able to 1) eavesdrop of communication (confidentiality) 2) tamper with the communication (integrity)

<u>Today</u>: secure communication on web (https://...) TLS protocol (transport layer security) two components : handshake (key exchange) record layer (confidentiality + integrity)

protecting data at rest: disk encryption

Most of this course: study mechanics for protecting confidentiality + data

- Encryption schemes for confidentiality
- Signature schemes for message integrity
- Key exchange for setting up shared secrets

End of this course: protecting communication => protecting computation

- Two users want to learn a joint function of their private inputs
 - Lis training models on private (hidden) dota
 - is comparing two DNA sequences privately
 - is private auction to determine winner without revealing bids
 - individual votes)
- We can show the following remorkable theorem:

"Anything that can be computed with a trusted party can be computed without !"

Logistics and administrivia: - Course website: https://www.cs.utexas.edu/~dwulf/courses/fa23

- See Ed Discussion for announcements, notes will be posted to course website (1-2 days after lecture)
- Homework submission via Gradescope (enroll via Coavos) _ one of there is programming assignment (Python)
- Course consists of 5 homework assignments (worth 70%) and two in-class exams (worth 30%)
- Five late days for the semaster: use in 24-hour increments, max 72 hours (3 late days) for any single assignment
- Some office hours will also be autilable on Zoom

<u>This semester</u>: Lectures will be recorded using Lectures Online Please participate virtually if you are feeling unwell

A brief history of cryptography: Original goal was to protect communication (in times of war) Basic idea: Alice and Bob have a shared key k Alice computes $C \in Encript(k, m)$ *i j j j j ciphertext key* message (plaintext) Bob computes m < Decrypt (k, c) to recover the message This tuple (Encrypt, Decrypt) is called a cipher K, M, C are sets (eg., K= M= C = {0,1323) Definition. A cipher is defined over (K, M, C) where K is a key-space, M is a message space and C is a ciphertext space, and consists of two absorithms (Encrypt, Decrypt): Encrypt: $K \times M \rightarrow C$ } functions should be "efficiently-computable" Decrypt: $K \times C \rightarrow M$ } theory: runs in probabilistic <u>polynomial</u> time [algorithm can be <u>randomized</u>] practice: fast on an actual computer (e.g., < 10ms on my laptop) Correctness: YKEK, YmEM: Decrypt (k, Encrypt(k, m)) = m "decrypting a ciphertext recovers the original message" Early ciphers: - Caesar cipher: "shift by 3" AHD Not a cipher! There is no key! Anyone can decrypt! L> Algorithm to encrypt is assumed to be public. <u>NEVER RELY ON SECURITY BY OBSCURITY!</u> - Harder to change system than a key BH⇒E $C \mapsto F$; ХнэА YHDB - Less scrutiny for secret algorithms $Z \mapsto C$ - Caesar cipher ++ : "shift by k" (k=13: ROT-13) k is the key > Still totally broken since there are only 26 possible keys (simply via brute force guessing) - Substitution cipter: the key defines a permutation of the alphabet (i.e., substitution) $\begin{array}{c|c} A \mapsto C \\ B \mapsto X \\ C \mapsto T \\ \hline \end{array}$ $Z \mapsto T \leftarrow$ substitution table is the key How many keys? For English alphabet, 26! $\approx 2^{88}$ possible keys very large value, <u>cannot</u> brate force the key

- Still broken by frequency analysis e is the most frequent character (~12%) g is the least frequent character (~0.10%)
- Can also look at digram, trigram frequencies
- Vigener apper (late 1500s) "polyalphabetic substitution" key is short phrase (used to determine substitution table): m = HELLO
 - k = CAT
 - k = un, Encrypt (k, m): HELLO + <u>CATCA</u> < repeat the key "EEPP

L'interpret letters as number between 1 and 26 addition is modulo 26

if we know the key length, can break using frequency analysis otherwise, can try all possible key lengths l=1,2,...

L> general assumption: keys will be much shorter than the message lotherwise if we have a good mechanism to deliver long keys securely, then can use that mechanis to share messages directly

Francier substitution ciphers: Enigma (based on rator machines) but ... still breakable by frequency analysis

Today: encryption done using computers, lots of different ciphers - AES (advanced encryption standard; 2000) "block cipher" "stream cipher"