CPSC 317
COMPUTER NETWORKING

Module 8: Security - Introduction

Some slides based on Kurose/Ross original slides, found at https://gaia.cs.umass.edu/kurose_ross/ppt.htm

WHAT SECURITY ISSUES HAVE WE SEEN THIS TERM?

- SMTP spoofing
- False BGP routing advertisements
- DNS hijacking
- No privacy in Ethernet, WiFi, IP, UDP, TCP
- IP (source) address spoofing
- How many times have I given the excuse:
  - “When <whatever> was designed, there were only a few players and they all knew and trusted each other”?

LEARNING GOALS: BASICS AND ENCRYPTION

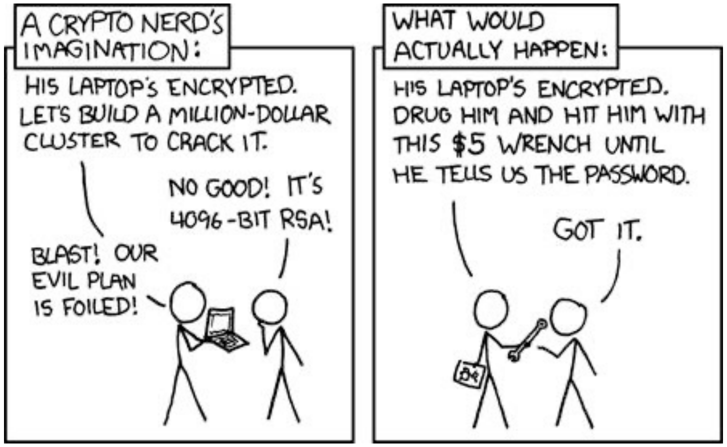
- Explain the principles of security:
  - Confidentiality, authentication, integrity, availability
- Evaluate the tradeoffs of providing security at different levels
  - Application, session, transport, network
- Demonstrate familiarity with security terminology
  - Alice, Bob, Trudy, plain text, cipher text, etc.
- Explain the role of encryption in confidentiality
- Explain the basics of an encryption scheme

READING

- Reading: 8.1, 8.2

SECURITY

- *"Secure" web servers are the equivalent of heavy armored cars. The problem is, they are being used to transfer rolls of coins and checks written in crayon by people on park benches to merchants doing business in cardboard boxes from beneath highway bridges. Further, the roads are subject to random detours, anyone with a screwdriver can control the traffic lights, and there are no police.* – Garfinkel, Spafford, "Web Security and Commerce"


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.

PRINCIPLES OF SECURITY

- **confidentiality**: only the sender and the intended receiver should "understand" message contents
  - sender *encrypts* message
  - receiver *decrypts* message
- **authentication**: the sender and receiver want to confirm each other's identity
- **message integrity**: the sender and receiver want to ensure that the message is not altered (in transit, or afterwards) without detection
- **access and availability**: services must be accessible and available to users

PRIVACY AND SECURITY

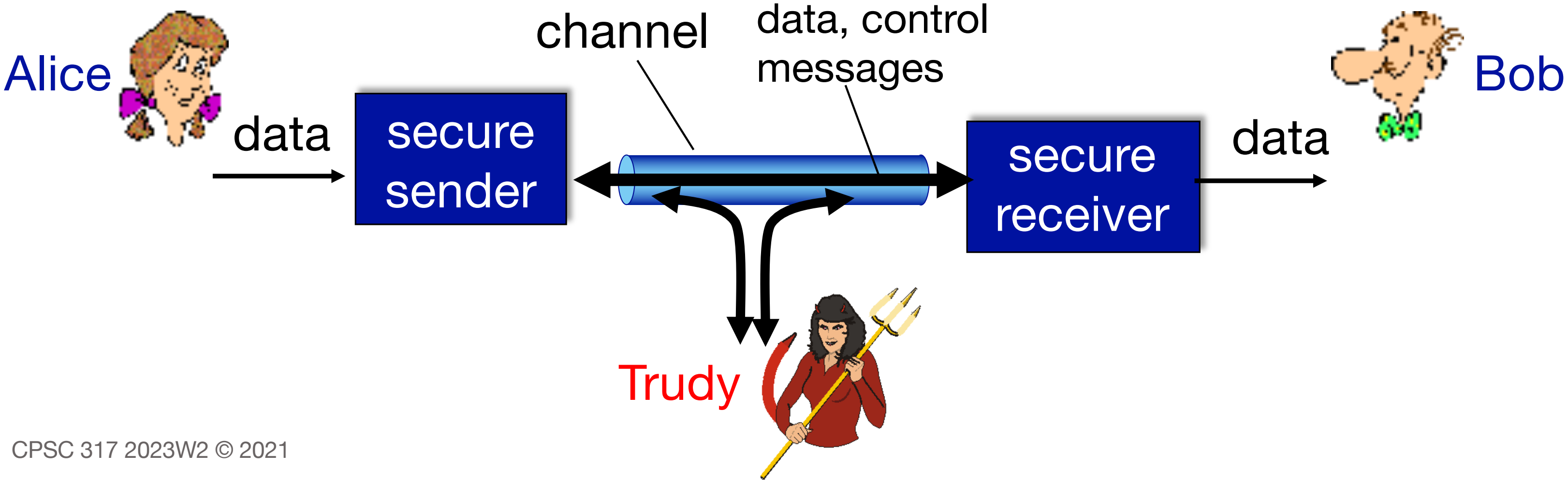
- TCP provides reliable, in-order delivery of data
- It does not provide security or privacy
  - Neither does UDP, for that matter
- How can we secure data?

ALTERNATIVES FOR SECURITY

- What level should be responsible for security?
- Options:
  - A new network-layer protocol (IPsec)
  - A new transport-layer protocol (QUIC)
  - A new “pseudo-layer” between transport and application (SSL, TLS)
  - Responsibility of the application (SSH)

THE ACTORS

- Most network security literature uses these names:
  - Alice, Bob want to communicate “securely”
  - Trudy (intruder) is an adversary who “attacks” the communication between Alice and Bob



THE ACTORS – ALICE AND BOB

“Alice” and “Bob” can be:

- Real-life people that want to communicate
- A browser and server for electronic transactions (e.g., web stores)
- A bank client and online banking service
- A DNS client and server
- BGP routers exchanging routing table updates

THE ACTORS – TRUDY

Threat model: What can “Trudy” do?

- ***eavesdrop***: sniff and record messages on the channel
- actively ***insert*** messages in the connection
- ***impersonation***: can fake (spoof) an entity, e.g., by spoofing source address in packet (or any field)
- ***hijacking***: “take over” ongoing connection by removing sender or receiver, inserting themselves in place
- ***denial of service***: prevent service from being used by others (e.g., by overloading resources, dropping packets)

CONFIDENTIALITY

- No one but the sender and the receiver should be able to interpret the content of the message
- Thought experiment: Where in “real life” do we have or expect to have confidentiality?
  - Phone call?
  - Conversation while outside?
  - Conversation while inside our home?
- Given that Trudy can see all the data, how do we provide confidentiality?

ENCRYPTION: THE BASICS

An **encryption algorithm** comprises:

- A method for *encrypting* the data
- A method for *decrypting* the data
- A *secret key* used in the decryption/encryption method
  - May be a pair of keys – one for encryption, one for decryption

ENCRYPTION: THE BASICS

Example: Caesar cipher

- Encryption: replace all letters by the corresponding letter K positions ahead in the alphabet
- Decryption: repeat process with $(-K)$
- Secret key is K
- Rot13 is the Caesar cipher with a key of 13

CLICKER QUESTION

Using the Caesar cipher with K = 13 (also known as rot13)

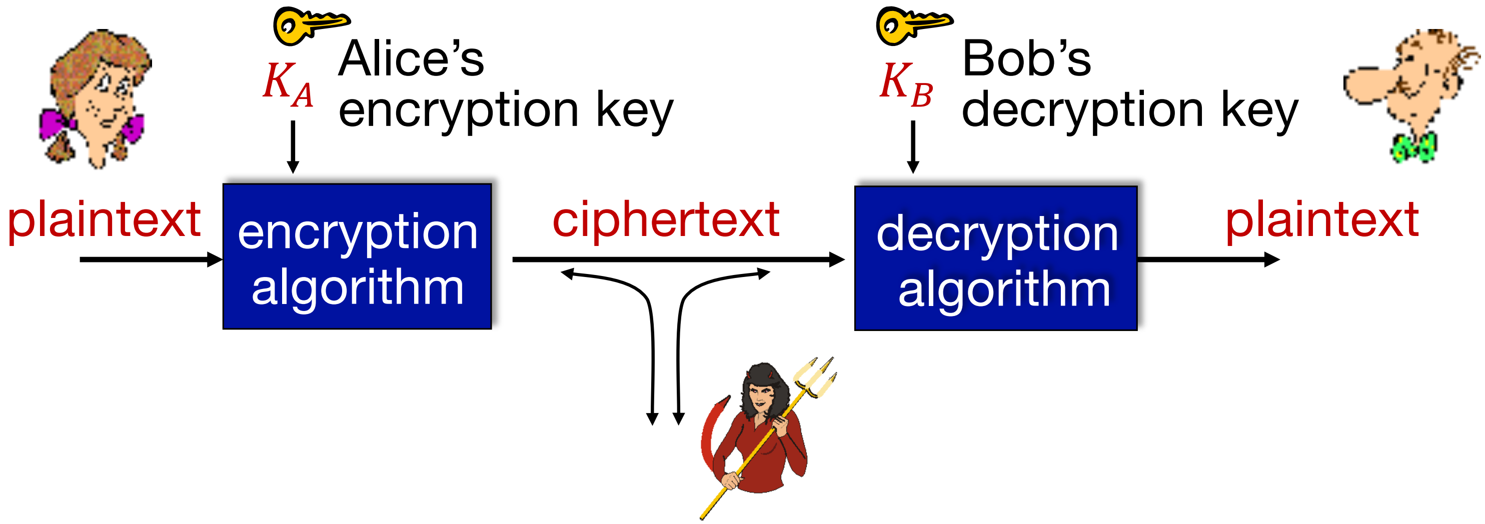
Decrypt the following message:

GHEAVC

ENCRYPTION APPROACH

- Naïve approach: security by obscurity
  - assume attacker is unaware of the decryption method
  - unsafe: good encryption algorithms are eventually made public
  - attackers can “pick a lock”
- Better approach: secret key
  - assume attacker knows the method, but not the key
  - decryption is impossible without the key

THE LANGUAGE OF CRYPTOGRAPHY



m: plaintext (a.k.a. clear text) message

$K_A(m)$: ciphertext, encrypted with key K_A

$K_B\left(K_A(m)\right) = m$

BREAKING AN ENCRYPTION SCHEME

- Ciphertext-only attack:
  - Trudy has ciphertext, but not plaintext (e.g., knows $K_A(m)$ but not m)
  - Option 1: brute force, search through all keys
  - Option 2: statistical analysis (look for patterns)
- Known-plaintext attack:
  - Trudy has some ciphertext with its plaintext (e.g., for some m it knows $K_A(m)$), wants to break other ciphertexts
- Chosen-plaintext attack:
  - Trudy has the ability to encrypt any plaintext (e.g., knows K_A, or can trick Alice into encrypting any message), but doesn't have key for decryption

IN CLASS ACTIVITY

- ICA81