Design and Engineering of Computer Systems

Lecture 25:
Network Security

Mythili Vutukuru
IIT Bombay
Network Security: Overview

- Many security problems can occur in a networked system
  - Man-in-the-middle attack: fake website impersonates real website and cheats users
  - Denial of Service attack: a large volume of traffic sent to server causes it to crash, service no longer available
  - Malware: malicious software designed to damage a computer system
  - Port scanning: probing traffic sent to various ports to find open ports/services
  - Tampering with data in transit, eavesdropping on confidential information

- Some good security properties we desire
  - Identify and filter out malicious traffic, attack traffic, malware
  - Origin authentication: guarantee that the website we are accessing is genuine
  - Data integrity: no one has tampered with content during transit over network
  - Confidentiality: no one has snooped on our traffic to steal private information

- In this lecture: key ideas to secure networked applications (basics only)
Firewall and intrusion detection system

- **Firewall**: monitors incoming/outgoing traffic in a network and controls it according to various security rules
  - Can be a software program or hardware appliance
  - Filters packets based on rules (block incoming requests to some ports, disallow traffic from/to specific hosts)

- **IDS (intrusion detection system)**: inspect incoming packets for malicious traffic
  - More sophisticated analysis than simple rules in firewalls
  - Detects anomalies in network traffic (too much traffic coming to specific hosts could be denial of service attack, too many connection requests could be port scanning attack)
  - Looks for signatures of well-known malwares, viruses in incoming traffic
  - Can raise alerts (intrusion detection) or block suspicious traffic (intrusion prevention)
Background: Public key cryptography

- Cryptography algorithms provide security properties using keys (strings of bits)
  - Based on hardness assumptions that some computations are hard to do
- Public key cryptography: a pair of keys used, one public, one private
  - Public key can be distributed in open, private key is kept secret with entity
- Cryptographic signatures for authentication
  - Alice generates $signature = sign(message, \text{private key of Alice})$ and sends it with message
  - Bob verifies message came from Alice by executing $\text{verify}(signature, \text{public key of Alice})$
- Encryption for confidentiality
  - Alice sends $cipher = encrypt(message, \text{public key of Bob})$ to Bob
  - Bob can recover original message with $decrypt(cipher, \text{private key of Bob})$
Background: Symmetric key cryptography

- Symmetric key cryptography: a shared secret key is used
  - Symmetric key algorithms are faster than public key algorithms
  - Mechanisms exist to agree on secret keys using public keys
- Message authentication codes (MAC) for data integrity
  - Alice sends message along with tag = MAC(message, secret key) to Bob
  - Bob can verify that message was not tampered with by regenerating tag, and verifying that it matches with received tag
  - If anyone tampered with message, recomputed tag will not match received tag
- Encryption for confidentiality with shared secret key
  - Alice sends cipher = encrypt (message, secret key) to Bob
  - Bob can recover original message with decrypt (cipher, secret key)
HTTP = requests and responses sent over regular TCP sockets
  - Anyone can snoop, tamper, pretend to be server
HTTPS = requests and responses sent over TLS/SSL sockets
  - TLS (Transport Layer Security), also called Secure Sockets Layer (SSL), built over TCP, with additional messages exchanged for security during connection setup
  - Separate port number (443, not 80) for servers using secure sockets
  - If URL indicates “https” instead of http, request sent to secure HTTP server
HTTPS provides several security guarantees
  - Authentication: website is authenticated to be genuine
  - Data integrity: data sent inside HTTP messages is safe from tampering
  - Confidentiality: data is encrypted, no one can snoop on private information

Only application layer security, outer TCP/IP headers are still visible
HTTPS: Details

- When we access a website, we want to know that the domain name, and associated information (e.g., public key) are all genuine.

- How is this guaranteed? Using trusted certificate authorities (CA), whom we trust to verify information about websites:
  - CA physically verifies and then signs information about a website (domain name, public key, ...) with its private key.
  - Website shares this SSL certificate with its clients during HTTPS setup.
  - Client (browser) can verify certificate using CA’s well-known public key (built into browsers).

- Website’s public key is used to arrive at a temporary shared secret key:
  - All HTTP request/response data is encrypted with shared secret key for confidentiality, then MAC added for data integrity.
Cryptographic hash functions

- Cryptographic hash functions: functions that map a large message to a small fixed size value (e.g., SHA-1, SHA-2, MD5)
  - Given message, compute hash(message) using hash function
  - Cannot invert function: cannot guess message by just looking at hash
  - Cannot easily generate another message that hashes to the same value

- Example: passwords in a computer system are stored as hashes
  - Do not store actual user password (privacy), instead store only the hash
  - No one can guess actual password by only looking at hash
  - When user enters password during login, hash it and verify match

- Example: websites post digests (hash) of large files (e.g., OS images). Users download image and verify digest to confirm integrity of data
Summary

• In this lecture:
  • Firewall, IDS to detect and filter malicious network traffic
  • HTTPS for authentication, integrity, confidentiality of HTTP data
  • Cryptographic hash functions

• Inspect traffic exchanged with a HTTPS website using Wireshark. Find out which information about packets is visible and which information is encrypted for confidentiality.