Carnegie Mellon University

Security

Outline

- **Introduction**
- Protecting data
- Protecting resources

Security

- Computer security is the protection of computer systems and information from harm, theft, and unauthorized use.
- This includes physical security (outside of our scope) as well as cyber security.

CIA definition of security

- Confidentiality – only authorized users can access data and resources.
- Integrity – data is not corrupted or modified
- Availability – information and resources are available to authorized users.

Verifying identity

- Authentication is the process of verifying that you are who you say you are.

On the internet, nobody knows you're a dog

Self identification

- You register your identity with some service. provider.
- You provide some means for future verification of your identify.
  - What you know – password, secret question, etc
  - What you have – smart card, phone number, etc
  - What you are – biometrics
- When you wish to use this service, you provide the verification.

Attested verification

- A third party verifies that you are who you say you are and registers you.
- The registration is documented by
  - Entering you Into a database
  - Providing you with credentials
    - Physical – smart card, credit card, ID from third party
    - Digital – key, certificate, external access to third party registration database

Authorization

- You are authorized to access resources if you have privileges enabling access.
- Privileges are usually associated with authentication mechanisms.
- Resources include individual files or an item's data, computer programs, computer devices and functionality provided by computer application.

Authorization server

- An authorization server
  - Assumes you have been authenticated
  - Provides a client a token that encodes privileges
- Tokens are usually ephemeral (expire after a specified interval).
- Examples:
  - OAuth
  - Kerberos

Security principles

- Least privilege: Users and programs should only have the necessary privileges to complete their tasks.
- Least functionality: Information systems are configured to provide only essential capabilities and to prohibit or restrict the use of non-essential functions, such as ports, protocols, and/or services that are not integral to the operation of that information system.

More principles

- Limiting access
  - Restrict number of access points. E.g. require access through a gateway
  - Restrict traffic. E.g. Firewalls can restrict access based on port numbers.

Outline

- Introduction
- **Protecting data**
- Protecting resources

Data

- Data should be protected from unauthorized reading or modification.
- Data can be
  - At rest – stored on a permanent storage medium.
    In transit – being transmitted over the internet
  - In use – in memory of a process

Outline - protecting data

- **Hashing**
- TLS

Hashing

- "Hello" → SHA-3 → 256 bit number

- A hash is a one way transformation based on a public algorithm with no key
- Not possible (very difficult) to decrypt
- NIST recommends the SHA-3 family of algorithms
- SHA-3 is notably faster than other hashing algorithms.
  - Measured in cycles per byte of value being hashed.
  - 12.6 cpb on a typical x86-64-based machine

Use of Hashing

- Used to verify integrity of data
  - Passwords: save hash of password but not password. When user enters password, compare to hash to verify.
  - Downloads: publish hash of software available for download. Compare hash of downloaded software. Verifies that software has not been modified.

Outline protecting data

- Hashing
- **TLS**

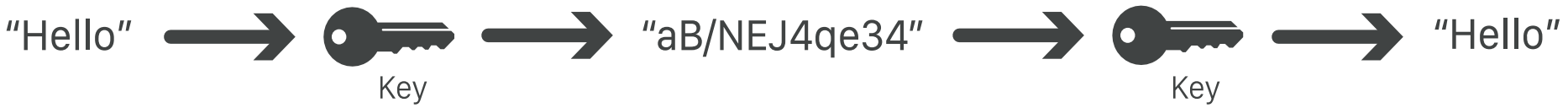
TLS

- TLS (Transport Layer Security) is designed to provide communications security over a computer network and to avoid eavesdropping and tampering.
- Broadly used.
- TLS depends on
  - Symmetric encryption
  - Asymmetric encryption
  - Public Key infrastructure

Encryption

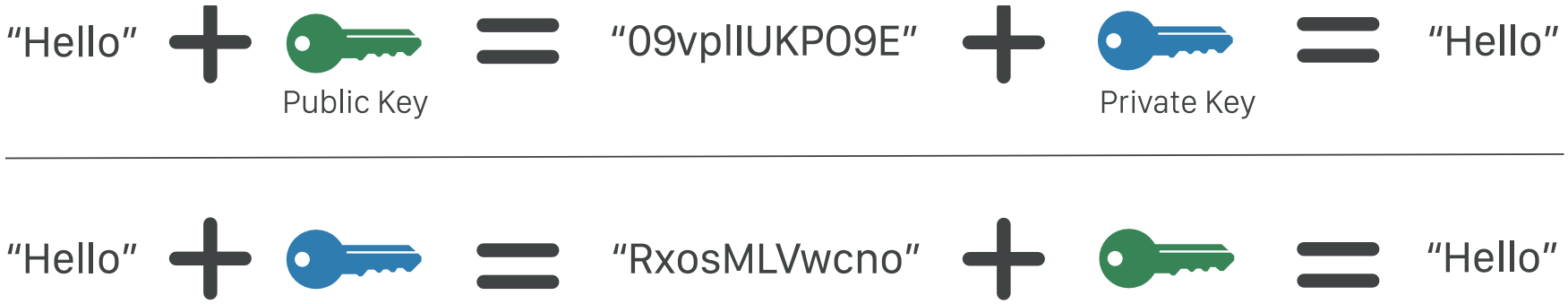
- Encoding data so that it is not readable without key.
- Two forms of encryption
- Symmetric – the same key is used for encryption and decryption.
- Asymmetric- one key is used for encryption and a separate key is used for decryption.

Symmetric encryption



- Use same key for encrypting and decrypting
- Suitable for data at rest
- A portion of solution for data in transit.
- NIST (National Institute of Science and Technology) approved algorithm is AES with key lengths of >128 bits

Asymmetric encryption



- Also known as public/private key encryption
- Messages encrypted with public key can be decrypted by private key (and vice versa)
- NIST approved algorithms: DSA, RSA, ECDSA >1024 bits

Performance comparison

- Symmetric encryption is ~4000x faster than asymmetric encryption.
- TLS uses asymmetric encryption to verify identity and symmetric encryption to transfer data

Public/private keys

- Public keys are available to the world.
- Private keys are kept private by the owner.
- Public and private keys are mathematically related but distinct.
- Knowing the public key of an individual does not allow you to deduce the private key of that individual.

Guaranteeing only recipient can read a message

- Suppose Bob wishes to send a message that only Alice can read.
- Bob encrypts the message with Alice's public key (known to the world).
- It can only be decrypted by Alice's private key (known only to Alice)

Guaranteeing a sender of a message

- Suppose Alice wishes to send a message that can only have come from her.
- She encrypts the message with her private key.
- It can be decrypted with her public key so anyone can read it.
- But only Alice knows her private key so only she can have sent it.

Digital Signature

- A digital signature is a means for sending an open signed letter.
- Anyone can read it but it is guaranteed to come from a particular party.
- You wish to send “text”.
- Hash “text” to get a hash value
- Encrypt the hash value with your private key
- The message consists of “text”+encrypted hash value.

The message cannot be altered

- Message cannot be altered.
- The hash value guarantees the integrity of the message, and the senders private key encrypts the hash value.
- Changing the message would require knowing the sender's private key.

Why encrypt just the hash value?

- Efficiency The hash value is much shorter than the full message. The time to decrypt depends on the length of the message. Shorter is faster.
- Compatibility. Different signing schemes exist and just encrypting the hash provides compatibility with multiple schemes.

Public Key Infrastructure

- A public key infrastructure (PKI) is a set of roles, policies, hardware, software and procedures needed to create, manage, distribute, use, store and revoke digital certificates
- A Certificate Authority (CA) is an independent organization that will issue a certificate only to a party that can verify its identity.

Digital Certificate

- A digital certificate is an electronic document used to prove the validity of a public key. The certificate includes information about the key, information about the identity of its owner and the digital signature of a CA.
- Two important elements of a certificate
  - URL of web site that has been certified.
  - Digital signature of a CA.

Getting a certificate

Bob

Registers web site

Trusted Certificate Authority

Trusted Certificate Authority

Certificate digitally signed by CA

S3D

Accessing Web Site

Man in the middle attack

- You are in the airport scanning for an available ISP
- You find “freewifi” and get an IP address from them.
- “freewifi” may be an attacker
- “freewifi” can
  - modify messages to spoof the web site and steal your credentials
  - eavesdrop on your communication with a web site

TLS Overview

- TLS begins with a handshake to establish identity and create symmetric key.
- Symmetric key is used to encrypt actual messages
- Discarded after session completes
- Another session will generate a different key

TLS handshake

- Establish identify
- Uses certificates which depend on public/private keys
- Because certificates are digitally signed, they can neither be modified or spoofed
- Use Diffie-Hellman algorithm to create session key for symmetric encryption

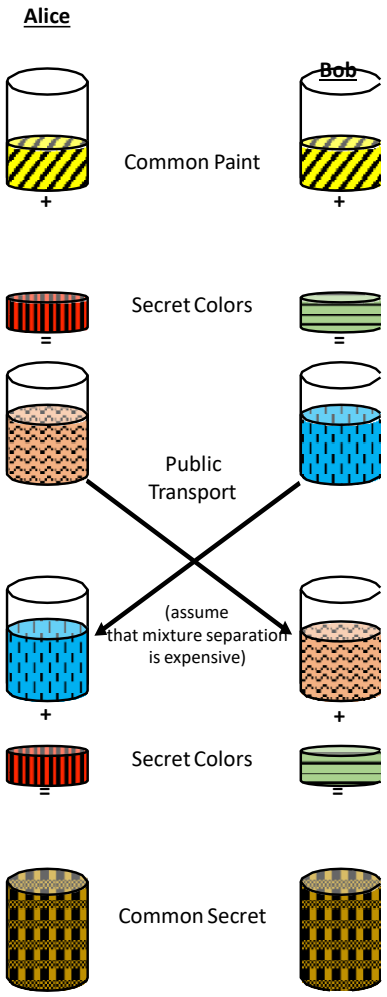
Creating symmetric key

- Suppose Alice and Bob wish to communicate securely.
- Communication over the internet is open to eavesdropping.
- Step 1 is to develop a shared symmetric key.

Diffie-Hellman

- The Diffie-Hellman algorithm is a means for Alice and Bob to generate a shared symmetric key even if there is an eavesdropper on their communication.
- Security of the algorithm is dependent on the difficulty of factoring large numbers.
- We present a more intuitive description using colors

Intuitive explanation of Diffie-Hellman



- Alice and Bob agree on a shared color.
- Alice and Bob both independently choose a secret color.
- Alice mixes the shared color with her secret color and sends the mixture to Bob
- Bob mixes the sharedshared color with his secret color and sends to Alice.
- Alice adds her secret color to the mixture she got from Bob. Bob does the same.
- The resulting color, on both sides, has the same components

S3D

Explanation of algorithm

- Determining the components of a color mixture is hard.
- A common large prime number and secret large prime numbers take the place of the colors in the example.
- Finding the prime factors of a number is NP hard.

Additional implementation issues

- The shared key is ephemeral. Once a session is over, it is cleared from memory. Even if it is leaked, the damage is limited to a single session.
- Generating the secret large primes depends on finding large random numbers.
- This in turn depends on physical phenomenon. E.g. electrical noise from computer components.

Thwarting man in the middle

- Man in the middle may see all messages but
  - Credential is digitally signed so it cannot be modified
  - Diffie-Hellman protects against eavesdropper (the man in the middle)
- Your communication with web site is encrypted using key unknown to man in the middle

TLS steps (again)

- Handshake to
  - establish identify
  - Create symmetric key
- Use symmetric key to encrypt all communications
- After session is complete, discard symmetric key.

Data in use

- Within an application
- Within a cookie
- Within cache
- Within log
- Typically decrypted

Protectting data in use

- Require authorization to access data
- Restrict sensitive data in use. Do not store sensitive data in cookies or logs
- Purge caches
  - Periodically or
  - On specific events. E.g. session end

Outline

- Introduction and general security activities
- Protecting data
- **Protecting resources**

Protecting resources

- Perimeter security
  - Restricting access
    - Authentication
    - Authorization
  - Assumption is that if user can get through perimeter, they are not malicious
- Zero trust security
  - Perimeter + authorization tokens. Do not trust even users who get through perimeter authorization checks.

Authorization tokens

- The authorization method issues tokens to users.
- Tokens embody access privileges.
- Tokens are ephemeral (expire after specified time).
- Resource manager – possibly API – verifies tokens prior to allowing access.

Denial of Service attack

- A Distributed Denial of Service (DDoS) is an attempt to limit availability by overloading the networks or servers hosting an application.
- To mitigate a DDoS attack
  - Limit attack surface
  - Ensure adequate network capacity
  - Ensure adequate server capacity
  - Maintain current back ups with different access privileges.

Limit attack surface

- The attack surface is the total number of all possible entry points for unauthorized access into any system.
- Techniques include
  - Disabling access to most ports. Firewalls only forward messages to specific ports.
  - Segmenting networks. Use principle of least privilege to control access

Use a Content Distribution Network

- A CDN is a network of servers linked together with the goal of delivering content as quickly, cheaply, reliably, and securely as possible.
- A CDN will place servers at the exchange points between different networks.

Backups

- Maintaining current backups in distinct datacenters with distinct access controls will guard against a ransomware attack.

Summary

- Security involves protecting both data and resources
- Data at rest can be protected by symmetric encryption
- Data in transit can be protcted by TLS\
- The integrity of data can be verified by hash function.
- Protecting resources involves
  - Authentication
  - Authorization
  - Access controls at resources