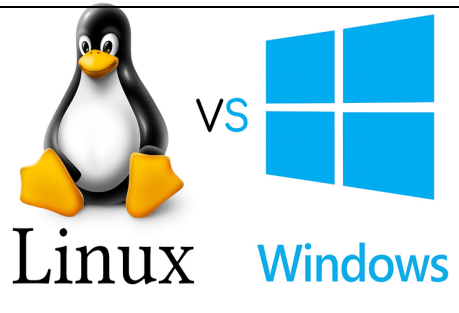
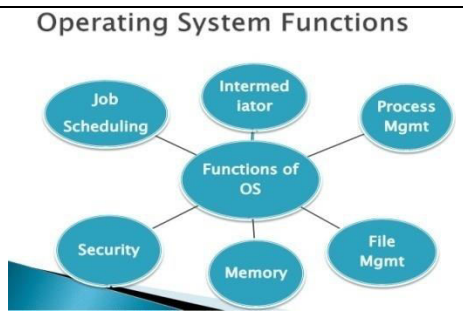
CIT210 OP. SYS. MANAG.

CSUN
COLLEGE OF ENGINEERING AND COMPUTER SCIENCE

Week 11 Day 2: Implementing Security in Cloud Environment, Attack Scenario, Hacking Process, Phishing, Cloud Security Risks and Countermeasures, (Encryption, Symmetric, Asymmetric, Digital Certificates, SSL), PGP, AES, RSA, S/MIME





ATTACK SCENARIO

Malicious Attacker

- Malicious attackers hide in Internet and try to
  - Steal the important or sensitive information
  - Collect user' data and sell these
  - Control or destroy the compromised system
  - Raise awareness
- Focus on the attackers' motivation.
  - Money
  - Philosophy
  - Show off

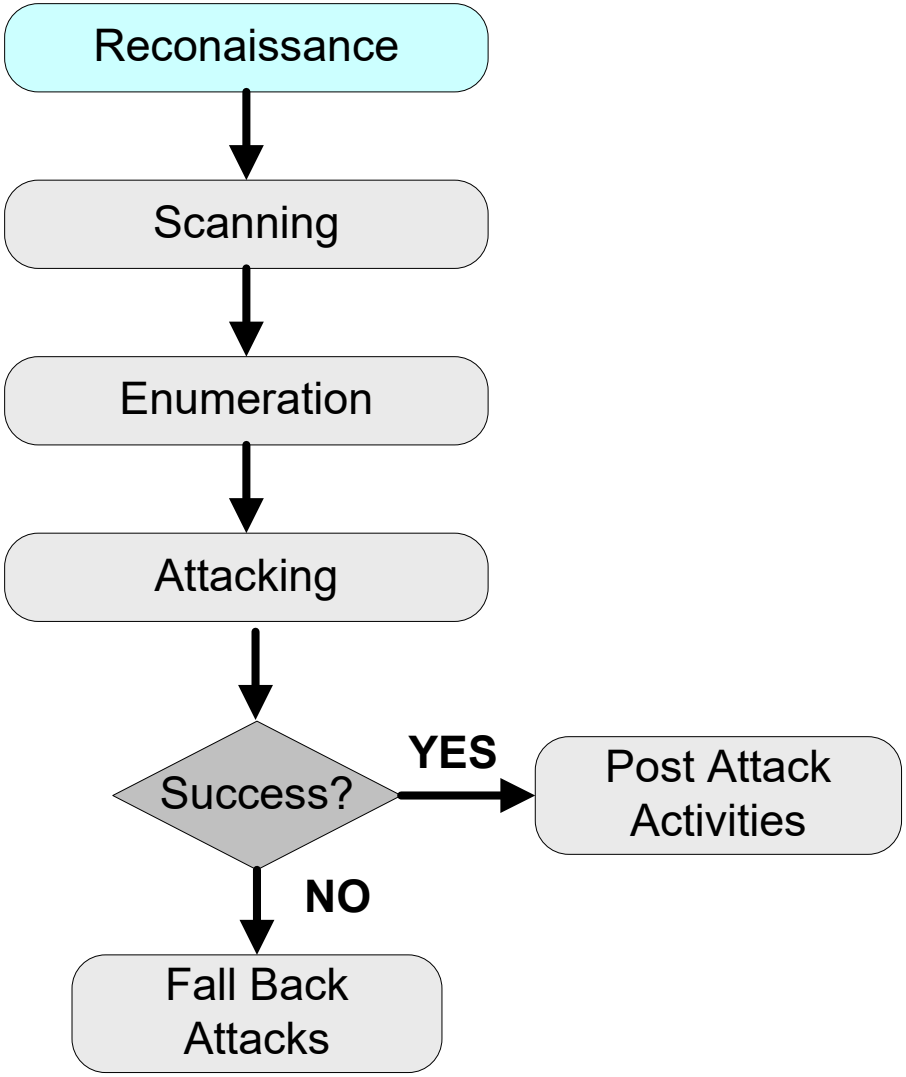
Hacker Economic

- How much of your personal information
  - Name, phone number and address
  - ID number
  - Credit card number
  - …etc
- But the price of data have been reduced day of day
  - Hacker must work hard
  - Automation
  - Large deployment
  - …etc

How to Attack?

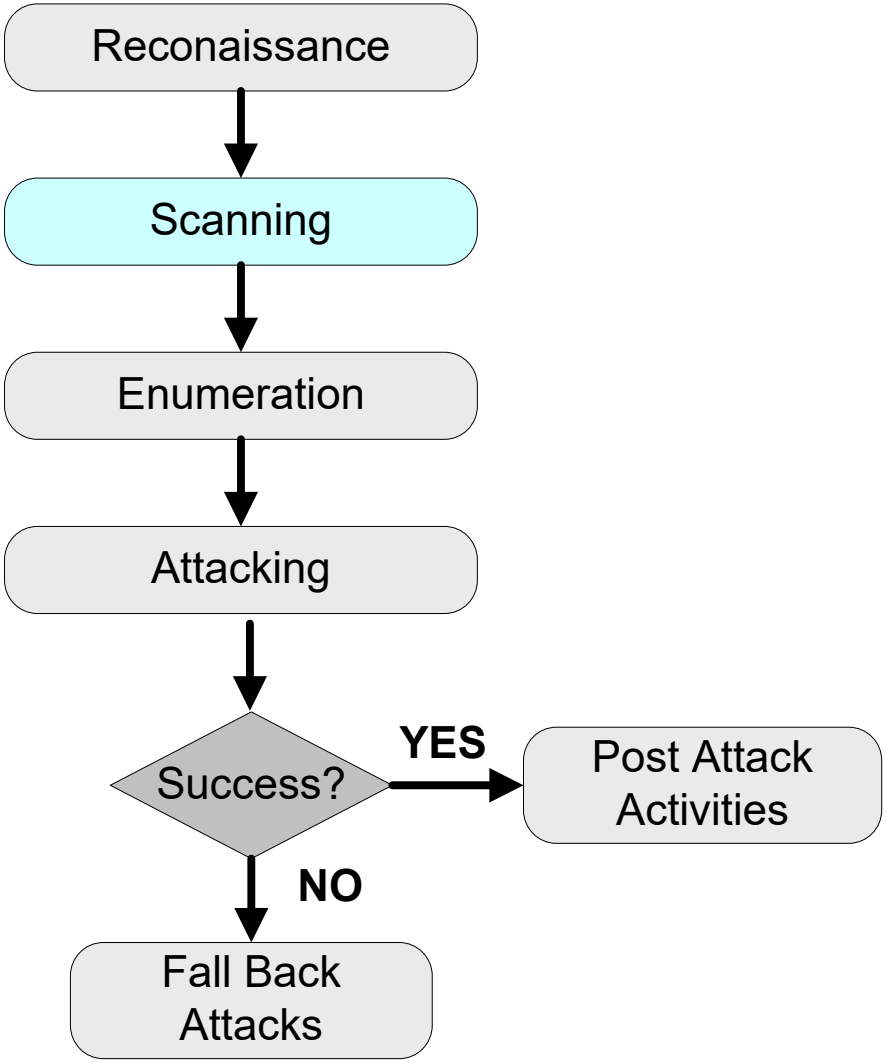
- How does hacker attacks the system or steal some information?
- Hacker is not the God
  - Still need some background and practices.
  - Still need to prepare some preprocess work.
  - Still need some tools.
- There are two parts of attack behavior
  - Penetration
  - Attack and destroy

The Hacking Process: Reconnaissance



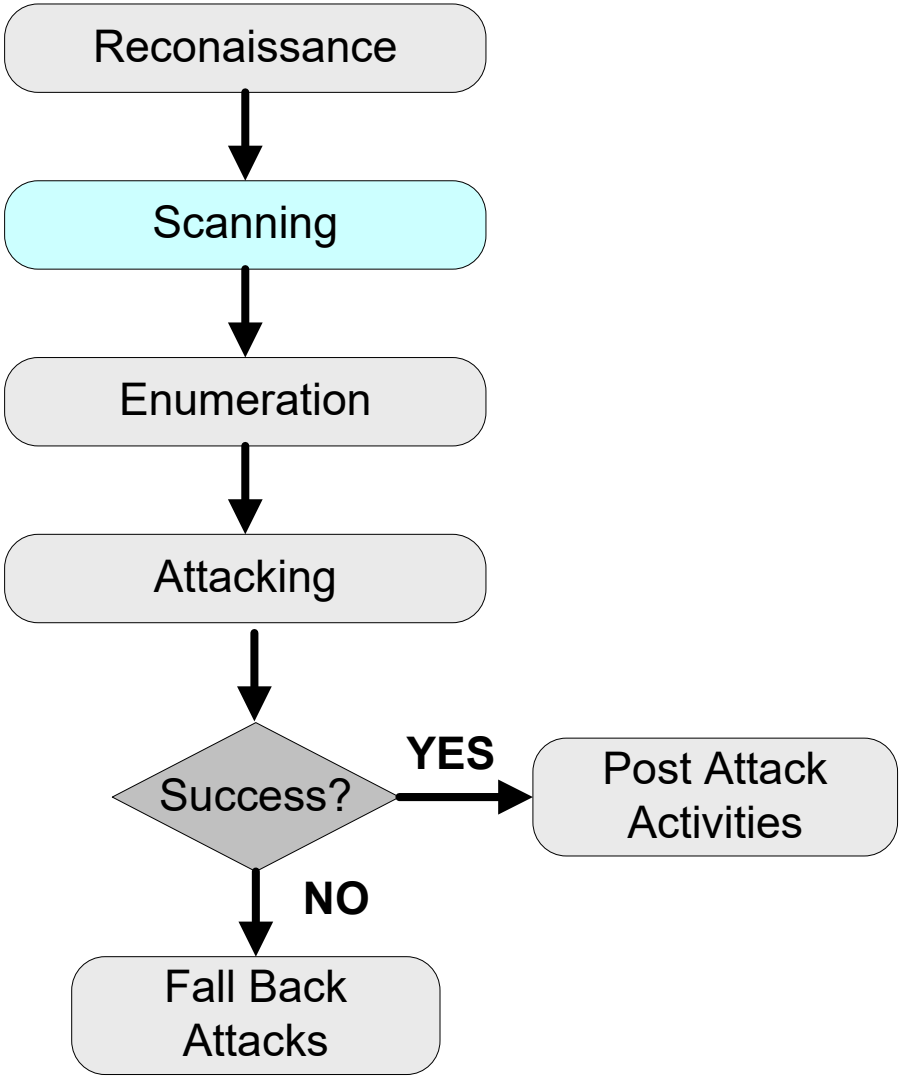
- Web research
- Search engines
- Background checks
- Newspaper searches
- Press releases
- Social networking
- Public records
- Eavesdropping
- Physical surveillance

The Hacking Process: Scanning



War dialing
using a modem to automatically scan a list of telephone numbers,
War driving
act of searching for Wi-Fi wireless networks by a person in a moving vehicle

The Hacking Process: Scanning



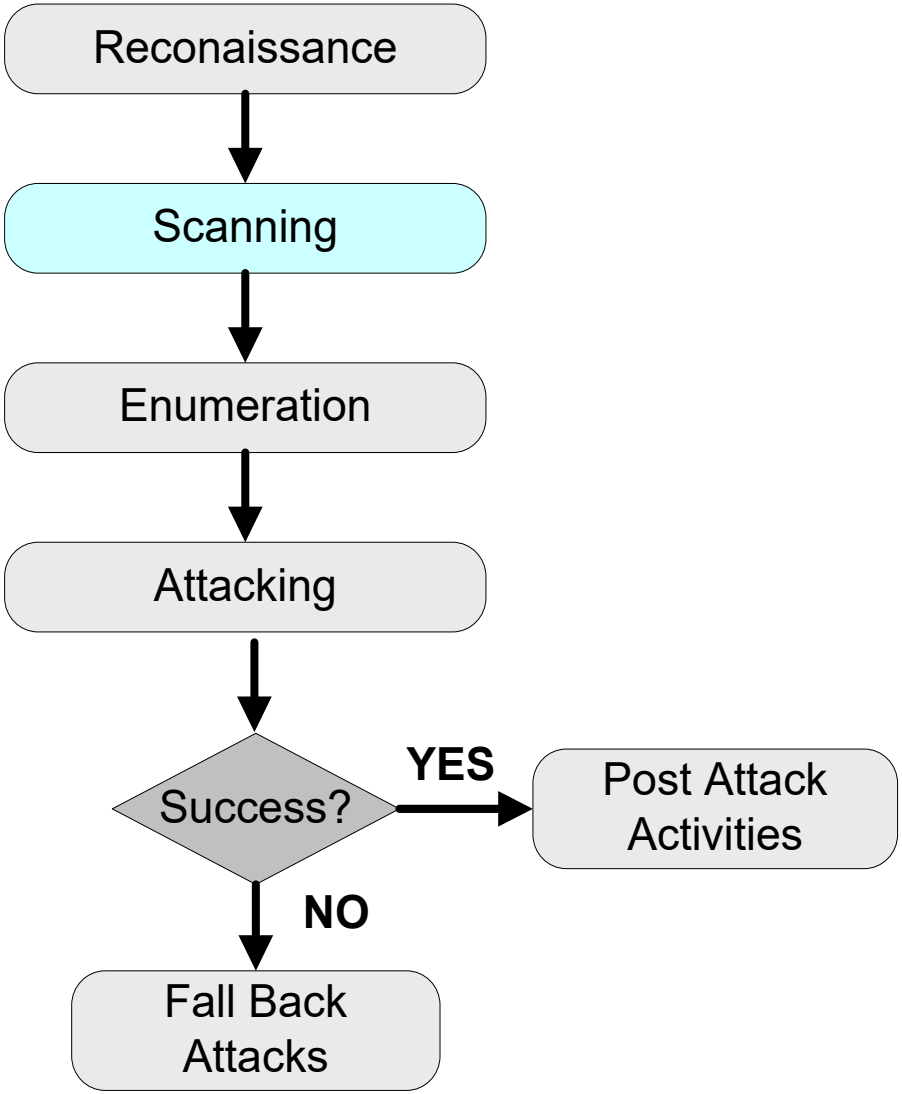
Netstumbling

is a tool for Windows that facilitates detection of Wireless LANs using the 802.11b, 802.11a and 802.11g WLAN standards
The program is commonly used for:

Wardriving

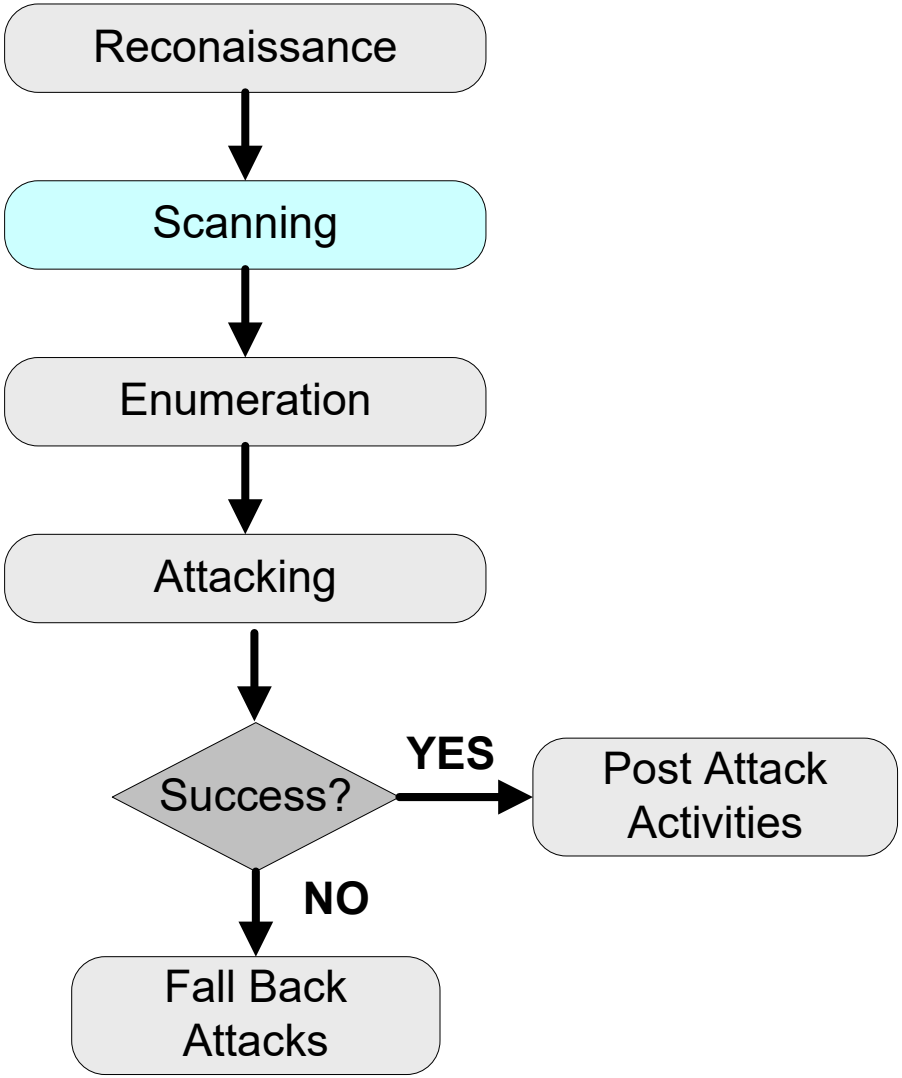
Verifying network configurations
Finding locations with poor coverage in a WLAN
Detecting causes of wireless interference
Detecting unauthorized ("rogue") access points
Aiming directional antennas for long-haul WLAN links

The Hacking Process: Scanning



Ping sweeps
is a technique used to determine which of a range of IP addresses map to live hosts.
It consists of ICMP ECHO requests sent to multiple hosts.
The classic tool used for ping sweeps is fping which traditionally was accompanied by gping to generate the list of hosts for large subnets

The Hacking Process: Scanning

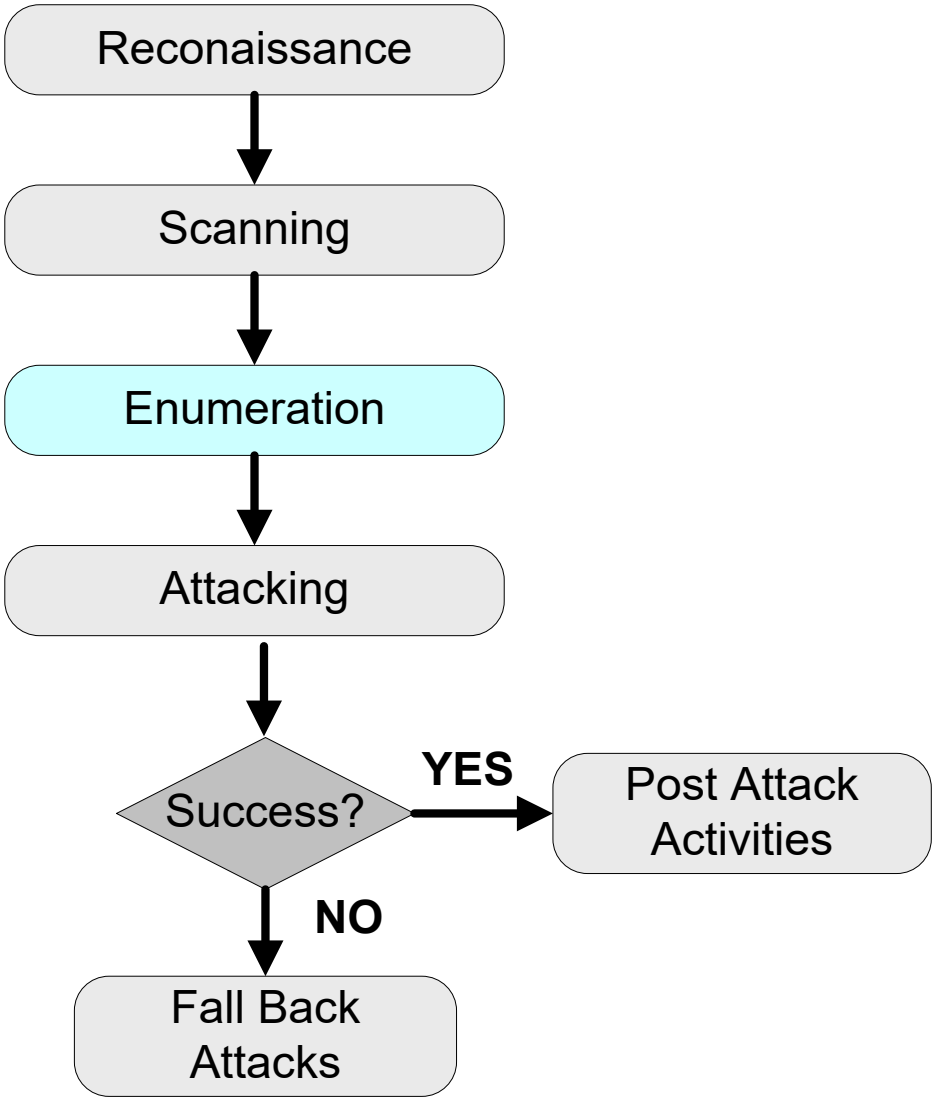


Port scanning

A **port scanner** is a software application designed to probe a server or host for open ports. This is often used by administrators to verify security policies of their networks and by attackers to identify running services on a host with the view to compromise it.

portsweep is to scan multiple hosts for a specific listening port. The latter is typically used in searching for a specific service,

The Hacking Process: Enumeration



Discover target details

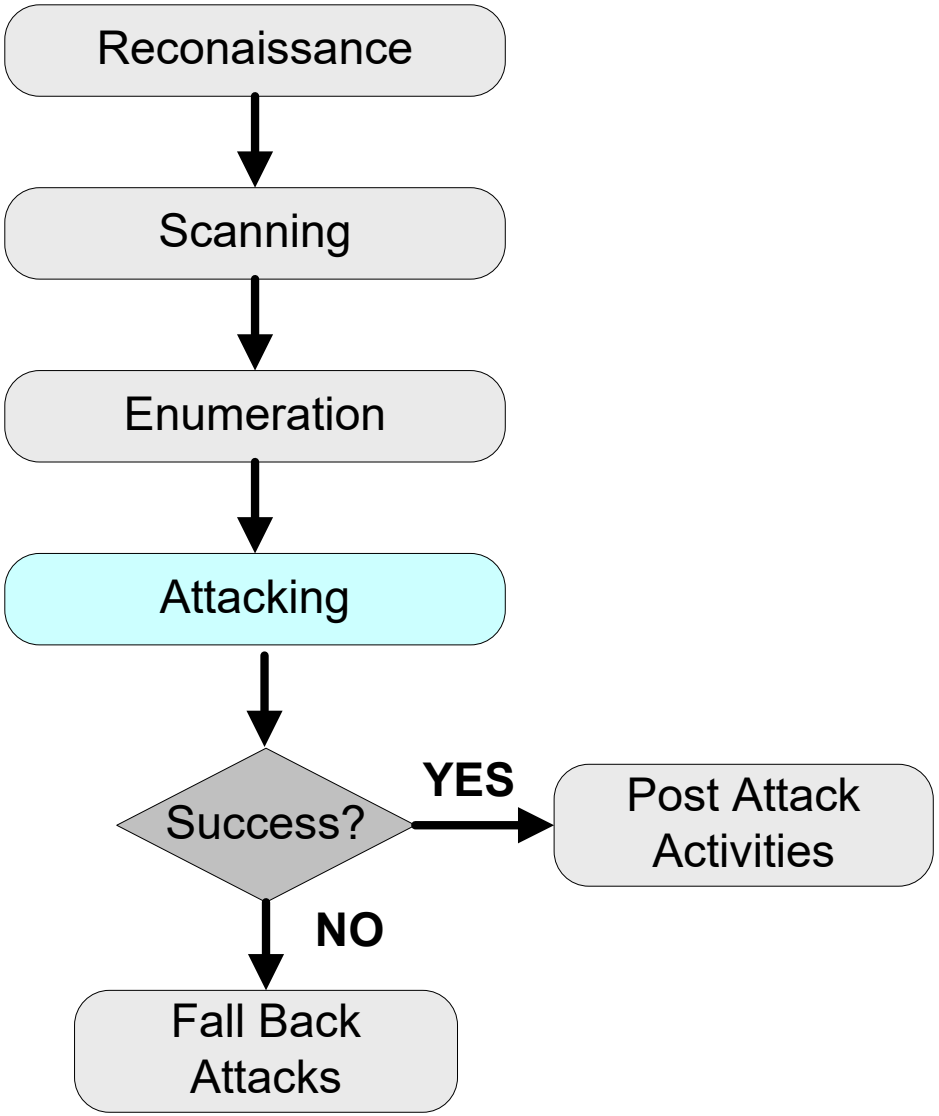
OS identification

Nmap

Service identification

Identify vulnerabilities

The Hacking Process: Attacking



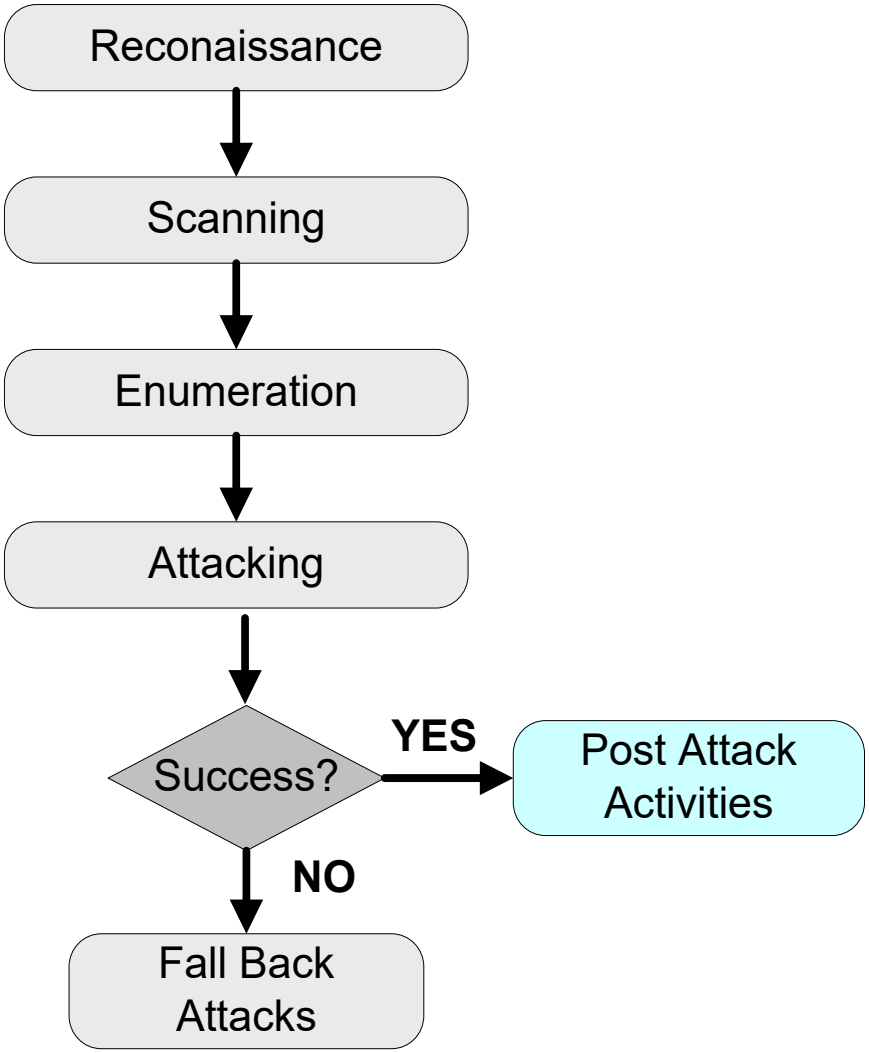
Exploit vulnerabilities

Network Nodes

Applications

Gain access

The Hacking Process: Post-Attack Activities



Attack is successful

Escalate privilege

Install additional tools

Accomplish objective (e.g., data theft)

Remove evidence

Workflow

- Penetration
  - Collect all public information.
  - Try to find the weakness or vulnerability.
  - Try to get the access right or administrator privileges.
- Attack and destroy
  - Steal or delete data
  - Crash the system
  - Monkey business

Penetration

- Hacker finds the weak point to access the target server and leave without trace
  - Weak password or SQL injection
  - Program vulnerability
  - Erase all log file
- Penetration test (also called pentest) is a method of evaluating the security
  - Simulate an attack from a malicious source.
  - Analysis the feasibility of an attack and guide improving the system security.

Attack and Destroy

- Hackers try to steal data, block service and destroy system
  - Compared with penetration, attacks do not need to be hidden.
  - Paralysis of the operational capabilities of the host and the firewall.
  - Install malicious code to destroy the system and data.

Password

- In general, password is the basic method for authentication
  - Length of password and characters determine the strength of security.
  - Only lowercase with 4 characters has 45,000 possibilities.
  - Common words can be crack in few minuets by dictionary attack.

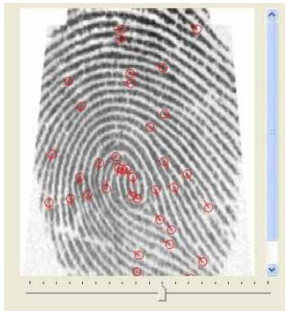
- In Cloud
- Numbers of users is grown more than million to billion
  - Hacker can use the few possibilities to try all of users.
  - Based on birthday attack, hacker has a high chance to get the correct password.
  - Always some people use unsafe passwords.

Solution

- Basically, in the registration phase
  - Force user to use alphanumeric passwords.
  - Force user to use more than 8 char passwords.
  - Limit user to use the command words in dictionary.
- Set security password policy
  - Limit the numbers of incorrect attempts.
  - Ban the attacker's IP with few minutes.
- But in some case, burse force could block the user's normal use
  - A kind of deny of service.

Solution (cont.)

- Service vendors can use other security technology to provide a high degree of security protection
  - Communication lock – unlock the account by user dialing to a special phone number.
  - Smart card - contain volatile memory and microprocessor components that provides strong security authentication.
  - Fingerprint – each one has his own fingerprint that cannot easily be forged.



Phishing

- Phishing is a way of attempting to acquire sensitive information by a fake web page
  - Cloned the normal web pages.
  - Steal password or credit card account
- There are three way of phishing
  - URL Shortening
  - A confusing link
  - SEO (Search Engine Optimization) poisoning

Phishing (cont.)

- URL shortening is a technique on WWW in which URL may be made substantially shorter in length and still direct to the required page.
  - Long URL cannot post in some forum, like Twitter or BBS.
  - Trying to type long URL by hand will be time-consuming and result in errors.
  - But short URL cannot determine the actual one or fake until link to.
- A confusing link is a URL that replace the similar characters or use the same meaning words
  - example.com vs examp1e.com
  - www.candy.com vs www.candies.com
- SEO (Search Engine Optimization) is the process of improving the visibility of a website in search engines.
  - User usually find a website by search engine
  - Hacker may poison SEO to increase the rank of phishing pages

In Cloud

- Cloud vendor provides many services and many service webpages
  - Users cannot remembers all of URLs
  - Users may fall into phishing web pages
- User can log in once and gain access to all services by SSO
  - Reduce phishing success, and users are not trained to enter password everywhere.
  - Each service can redirect to other service.

In User

- User should be careful on account needing to be verified or any other topic used by phishing
- User should improve concentration on detail of email
  - All legitimate e-mail messages contain an item of information that is not readily available to phishers.
- User can use plug-in or software to help identify legitimate sites

Communication

- A packet is a formatted unit of data
  - Computer communications links that simply transmit data as a series of bytes.
  - A message would separate into several segment packets and each packet transmits to target computer.
  - Receiver reconstruct message from these packets.
- Each packets would pass through several computers and switches.

Sniffer

- Sniffer means someone intercept and log traffic pass over a digit network or part of a network.
- NIC(network interface card) drops any packet which receiver is not his MAC address
  - NIC in promiscuous mode would not drop any packet.
  - Hacker can get any packet pass through his NIC by setting promiscuous mode.

ARP Spoofing

- Switch uses ARP protocol to record the pair of MAC address and IP address
  - Hacker uses ARP spoofing to attack a LAN (local-area network)
  - Hacker send the ARP request and register his MAC address before the legal user.
  - Hacker also can send lots of useless ARP request for DoS (denial-of-service) attack

VLAN switch

- Using VLAN switch, user only can access switch and cannot connect to other users
- In high-end device, switches can detect the ARP spoofing/attack and alert administrator the malicious behavior.
- Switches can also set the static route table
  - But in many case, the IP address would be dynamic decided

In Cloud

- Cloud service venders limit the environment setting that can reduce the probability of sniffer.
- Using SSL/TSL cryptographic protocols that provide communication security over the internet
  - Even hacker getting the packet also cannot get the sensitive message.

DoS

- Denial-of-Service (DoS) is an attempt to make a computer resource (like storage service) unavailable
  - Hacker send lots of request to server such that server cannot response to legal users.
- Server can only provide services to fixed users
  - Hackers account for all the resource as much as possible.
  - Other users wait until server is available.

Attack

- In general, a server can handle hundreds of requests at once
  - A desktop can send hundreds of small request to attack a server.
  - But in large-scale server, host can easily handle thousands to hundreds of thousands request.
  - Hacker must need more resource.
- DDoS (distributed denial of service) occurs when multiple systems flood the bandwidth or resources of a target system.

Botnet

- Hackers want more resource to attack
  - Hackers using Trojans and worms control lots of computers.
- A botnet is a collection of compromised computers that are used for malicious purposes
  - The average size of a botnet is estimated at 20,000 computers
  - Hackers control compromised computers to send spam and DoS target.

In Cloud

- Cloud computing provides service to many users
  - One of the cloud properties is availability.
- User access cloud service anytime and anywhere
  - This means cloud platform must can service thousands of users.
  - Cloud platform can distribute workload across multiple computers and services.
- The general DDoS attack cannot affect the cloud environment.

In Cloud (cont.)

- Cloud vendors offer pay-per-usage or pay-as-you-go access to computers and services.
- Hacker may burn your quota
  - DDoS may not affect the end user, but can increase the quota of usage of bandwidth.
  - Hackers can send lots of query to increase spending of company

Cloud Security Risk and Countermeasure

- Abuse and nefarious use of cloud computing
  - Countermeasures include:
    - Stricter initial registration and validation processes
    - Enhanced credit card fraud monitoring and coordination
    - Comprehensive inspection of customer network traffic
    - Monitoring public blacklists for one's own network blocks
- Insecure interfaces and APIs
  - Countermeasures include:
    - Analyzing the security model of CSP interfaces
    - Ensuring that strong authentication and access controls are implemented in concert with encrypted transmission
    - Understanding the dependency chain associated with the API

Cloud Security Risk and Countermeasure

- Malicious insiders
  - Countermeasures include:
    - Enforce strict supply chain management and conduct a comprehensive supplier assessment
    - Specify human resource requirements as part of legal contract
    - Require transparency into overall information security and management practices, as well as compliance reporting
    - Determine security breach notification processes

Cloud Security Risk and Countermeasure

- Shared technology issues
  - Countermeasures include:
    - Implement security best practices for installation/configuration
    - Monitor environment for unauthorized changes/activity
    - Promote strong authentication and access control for administrative access and operations
    - Enforce SLAs for patching and vulnerability remediation
    - Conduct vulnerability scanning and configuration audits
- Data loss or leakage
  - Countermeasures include:
    - Implement strong API access control
    - Encrypt and protect integrity of data in transit and at rest
    - Analyze data protection at both design and run time
    - Implement strong key generation, storage and management, and destruction practices

Cloud Security Risk and Countermeasure

- Account or service hijacking
  - Countermeasures include:
    - Prohibit the sharing of account credentials between users and services
    - Leverage strong two-factor authentication techniques where possible
    - Employ proactive monitoring to detect unauthorized activity
    - Understand CSP security policies and SLAs

- Unknown risk profile
  - Countermeasures include:
    - Disclosure of applicable logs and data
    - Partial/full disclosure of infrastructure details
    - Monitoring and alerting on necessary information

Important Commercial Algorithms

- The most important symmetric algorithms from a commercial point of view are:
  - DES (Data Encryption Standard)
  - 3DES
  - AES (Advanced Encryption Standard)
  - RC5
  - Blowfish

DES

- This has been a US government standard for many years (although recently complimented with AES).
- Adopted in 1977 by National Bureau of Standards (now NIST)
- Divides message into blocks of 64 bits, and uses a key of 56 bits
- Key idea for this: XOR the data with the key
- It uses a 64-bit key (actually, only 56 bits are used for the encryption, the other 8 bits are parity bits), so it is no longer viable.
- Increased processing speeds (in recent years) are making brute force attacks on DES more viable.

DES

- In July 1998, DES was officially cracked by a machine built by the EFF
  - Total cost: under $250,000
  - Total time: 6-8 months
- They then published the details of their approach, which essentially was a brute force attack
- Note: 56 bits means 2^{56} keys to try

3DES

- Then, came the idea of using DES twice on a given message.
- A subtle form of attack was discovered which made 2DES no better than DES.
- 3DES proved to have the properties that 2DES was supposed to have.
- 3DES is a viable and popular symmetric block algorithm.
- Main algorithm: repeat DES 3 times with different keys (so key size is now 168 bits)
- Still very secure - brute force attacks would take too long, and that is the only way to attack this algorithm
- Main problem: it is inefficient , SLOW

Advanced Encryption Standard (AES)

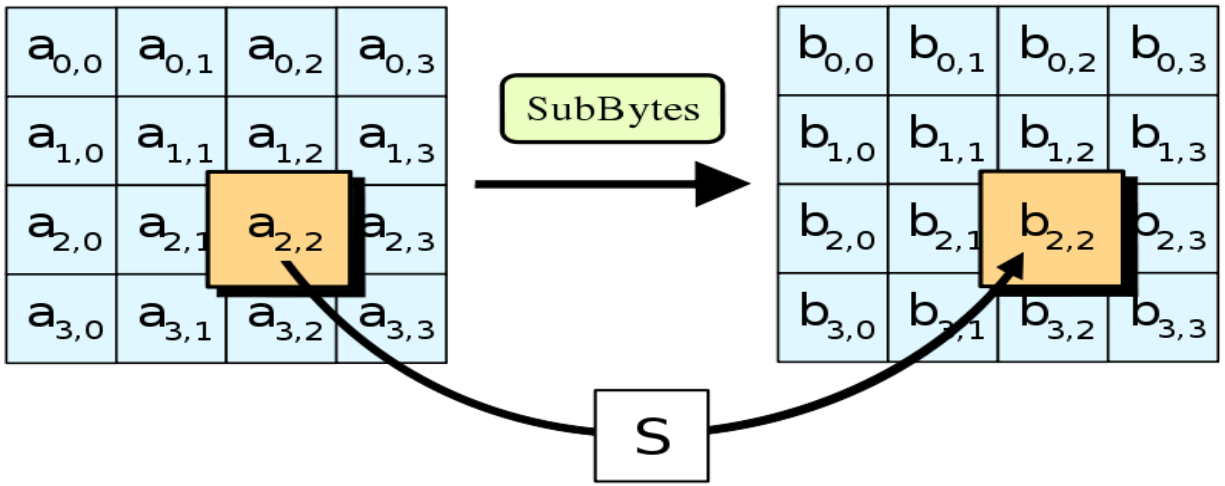
- Designed in response to a call by NIST in 1998, and officially adopted in 2001
- Block length is 128 bits, and keys can be 128, 192, or 256 bits.
- Essentially, proceeds in 4 rounds (which are repeated):
  - Substitute bytes
  - Permute
  - Mix columns
  - Add round key

AES

- AES is now an accepted federal standard and is widely available in open source form. Implementations are available in C++ and Java.
- 3DES still has the advantage that it has been studied (in DES) form for many years.
- The guestimate is that AES will be a viable encryption standard for the next 50 years, but there could be some surprises down the raod.

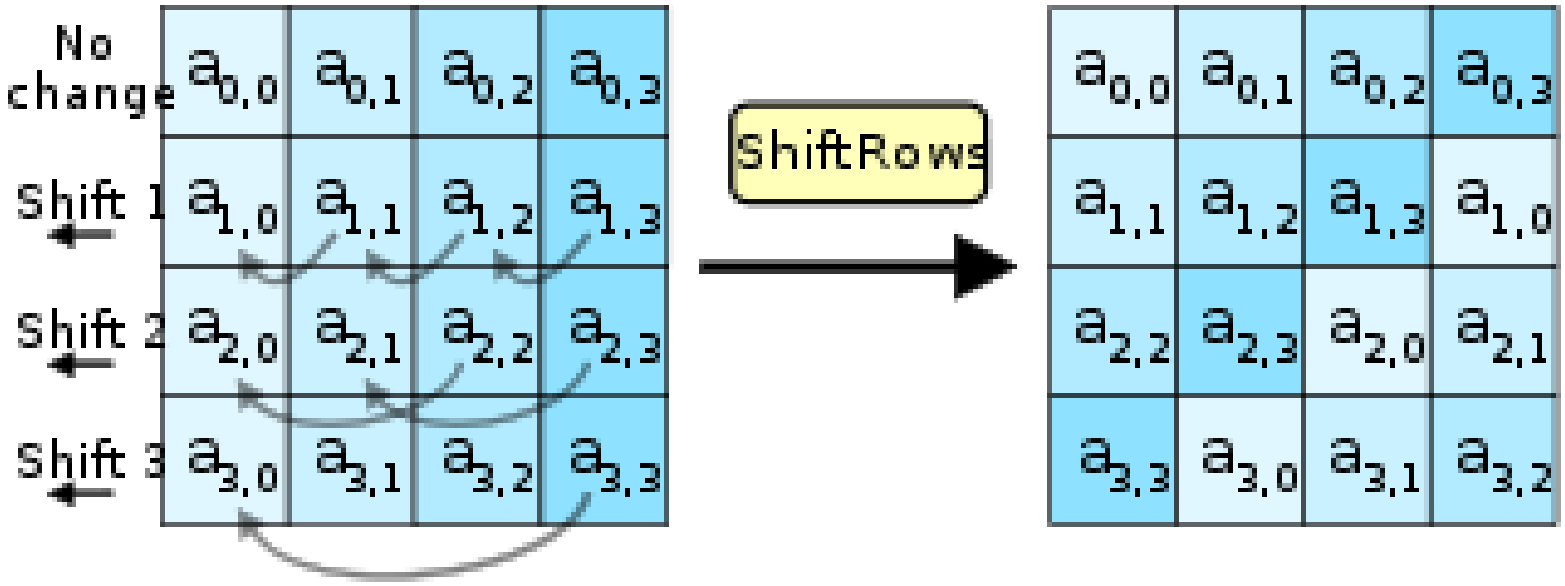
Stage 1: substitute bytes

- AES computes a matrix which maps every 8-bit value to a different 8-bit value



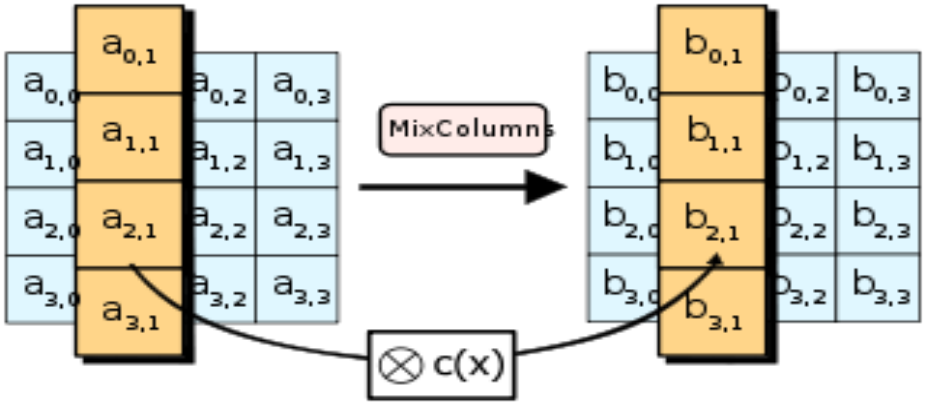
Stage 2: permute

- AES then shifts each row, where each row is shifted a different amount



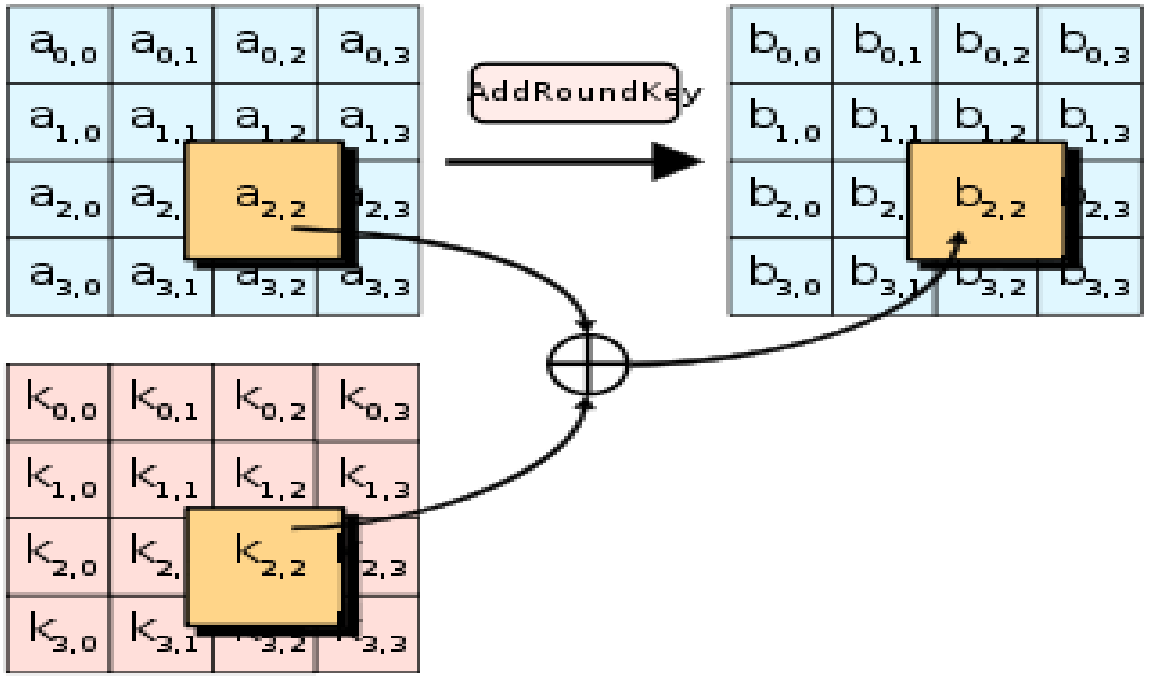
Stage 3: Mix columns

- Here, the 4 bytes in each column are combined using a linear transformation
- Essentially, the output of any byte depends on all the input bytes, so this “mixes” them together



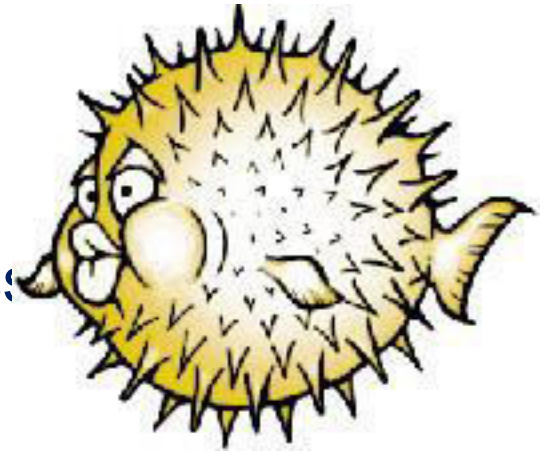
Stage 4: Add round key

- Use XOR to combine the key with the message



Blowfish

- 64-bit block cipher
- Much faster than DES
- Variable key length:
  32-448 bits
- Many attempted crytanalyses, none success
- Widely used: ssh, OpenBSD, PGPFone



Blowfish → Twofish

- Blowfish: runs encryption 521 times to produce S-boxes
  - Too slow for AES, requires too much memory for smart cards
- Twofish
  - Provides options for how many key-dependant S-boxes (tradeoff security/time-space)
  - Also: increase block size (128 required by AES), change key schedule, etc.

RC5

- Also a block cipher, invented by Ron Rivest (1994)
  - Similar in structure to Feistel
- Operations: XORs, Additions (mod bitsize), and Rotations
  - Word-oriented, Low-cycle operations – Fast in software
- Variable length blocks, keys, and number of rounds (*r*)
  - Each block is made of 2 w-bits blocks (A, B) (*w*=16,/32/64)
  - Each key is made of *b*x8 bits (0<*b*<255; can be larger than a block)
  - Round keys (S_{2i} , S_{2i+1}), each with w bits, are derived from the key
  - Encryption and decryption consist of *r* rounds
- With 16+ rounds, RC5 resists differential attack
  - 12 round RC5 shown susceptible with 2^{44} chosen plaintexts
- Data-dependent shifts is one of the innovations of RC5

Asymmetric / Public Key Encryption

- Some examples of public-key cryptosystems are
  - Diffie-Hellman (named, you guessed it, for its inventors)
  - RSA (named for its inventors, Ron Rivest, Adi Shamir, and Leonard Adleman)
  - Elgamal (named for its inventor, Taher Elgamal)
  - DSA, the Digital Signature Algorithm (invented by David Kravitz).
  - PGP (Pretty Good Privacy) is fairly popular and inexpensive

Diffie-Hellman

- Whitfield Diffie and Martin Hellman, “New Directions in Cryptography” in 1976
- Public Key Cryptography
  - Exponential key agreement
  - Allows two users to exchange a secret key
  - Requires no prior secrets
  - Real-time over an untrusted network
- For 100 people all communicating with each other, you need 4950 keys.
- Group of N people, $N(N-1)/2$ keys. Unmanageable.
- Diffie-Hellman key exchange protocol, often shortened to “DH protocol”

Diffie-Hellman

- Basic DH
  - A specific method of securely exchanging cryptographic keys over a public channel and was one of the first public-key protocols as originally conceptualized by Ralph Merkle.
  - One of the earliest practical examples of public key exchange.

Diffie-Hellman

How does it work?

- Traditionally, secure encrypted communication between two parties required that they first exchange keys by some secure physical channel, such as paper key lists transported by a trusted courier.
- The Diffie–Hellman key exchange method allows two parties that have no prior knowledge of each other to jointly establish a shared secret key over an insecure channel.
- This key can then be used to encrypt subsequent communications using a symmetric key cipher.

Asymmetric Encryption

- Two most popular algorithms are RSA & El Gamal
  - RSA
    - Developed by Ron Rivest, Adi Shamir, Len Adelman
    - Both public and private key are interchangable
    - Variable Key Size (512, 1024, or 2048 buts)
    - Most popular public key algorithm
  - El Gamal
    - Developed by Taher ElGamal
    - Variable key size (512 or 1024 bits)
    - Less common than RSA, used in protocols like PGP

What is it?

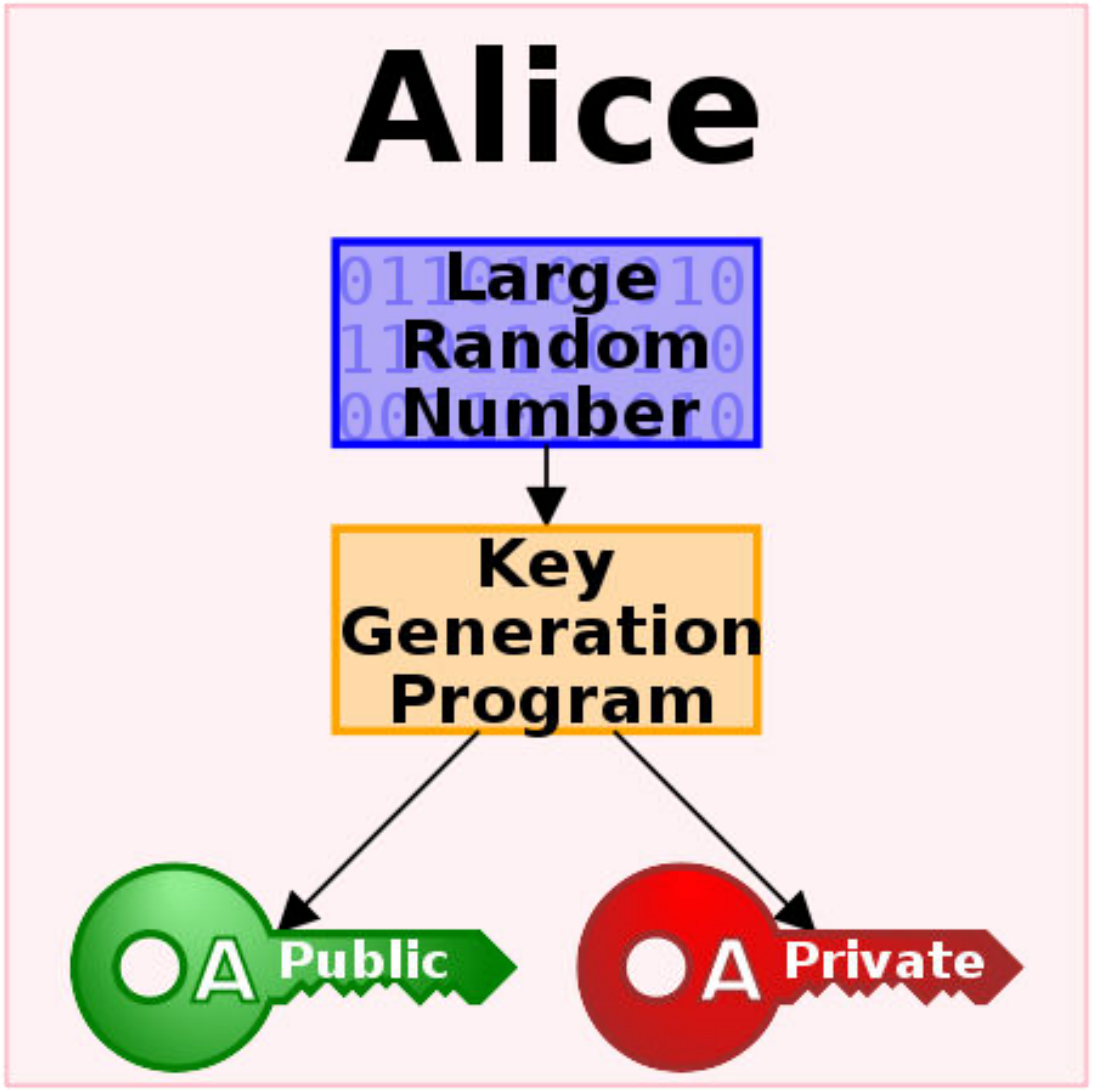
1. RSA – Rivest, Shamir, Adleman

2. Algorithm for public key cryptography. Public-key cryptography refers to a cryptographic system requiring two separate keys, one to lock or encrypt the plaintext and one to unlock or decrypt the cypher text.

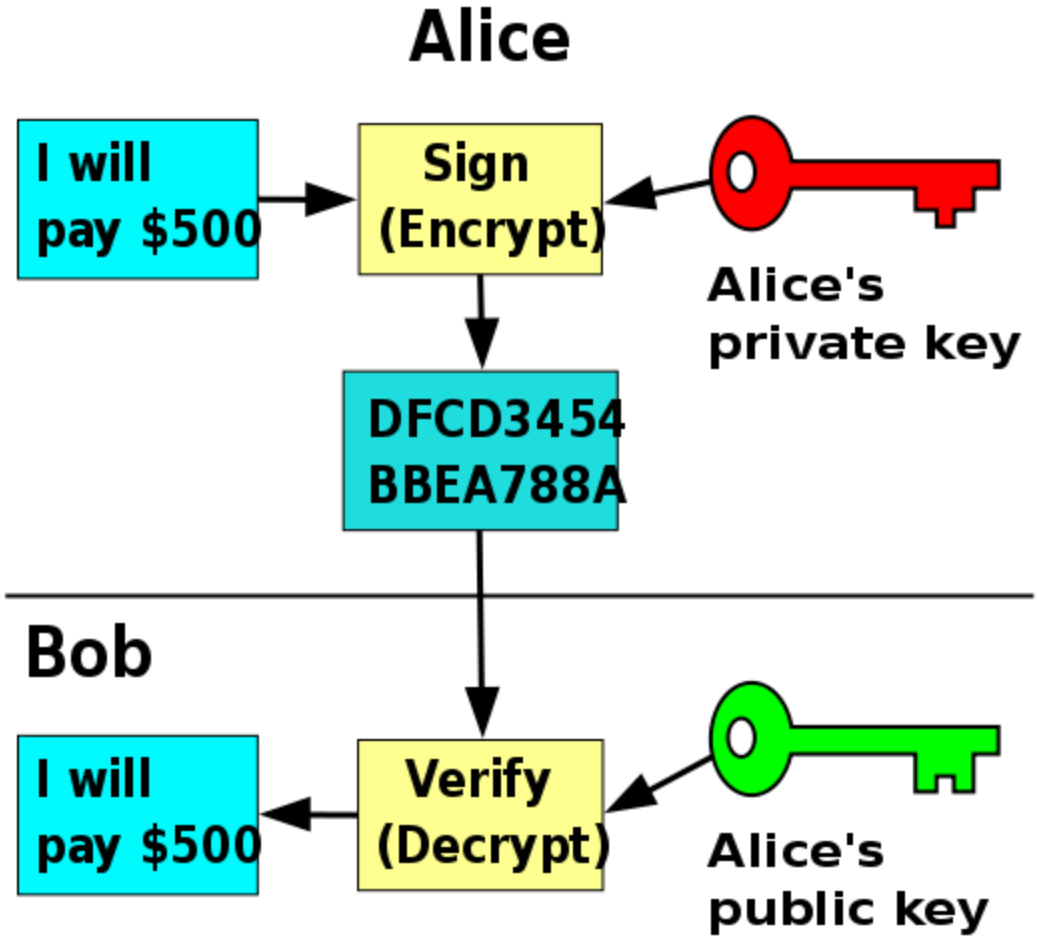
3. Wildly used in e-commerce

4. Involves 3 steps – key generation, encryption and decryption

RSA


Alice
Large Random Number
Key Generation Program
Public
Private

RSA


Alice
I will pay $500
Sign (Encrypt)
Alice's private key
DFCD3454 BBEA788A
Bob
I will pay $500
Verify (Decrypt)
Alice's public key

RSA Algorithm

- Generating the private and public key requires four steps:
- Choose two very large prime numbers, p and q
- Compute $n = p \times q$ and $z = (p - 1) \times (q - 1)$
- Choose a number d that is relatively prime to z
- Compute the number e such that $e \times d = 1 \bmod z$

RSA Encryption and Decryption

- Public key consist of pair (n, e)
- Private key consists of pair (n, d)

- Encryption of message block m:
  - $c = m^e \bmod n$
- Decryption of ciphertext c:
  - $m = c^d \bmod n$

PGP(Pretty Good Privacy)

- PGP is an e-mail security program written by Phil Zimmermann, based on the IDEA algorithm for encryption of plaintext and uses the RSA Public Key algorithm for encryption of the private key.
- PGP incorporates tools for developing a public-key trust model and public-key certificate management.
- PGP is an open-source freely available software package for e-mail security.
- It provides authentication; confidentiality; compression; e-mail compatibility; and segmentation and reassembly.

Pretty Good Privacy

- Public Key encryption used in PGP
- Generally available, and can be used for
  - encryption of messages
  - digital signatures.
- PGP combines DES and RSA
  - DES fast, but symmetric, hence key distribution problem
  - RSA slower, but no key distribution problem
  - Solution: Use RSA to encrypt and distribute key for DES encryption!!!

PGP Services

Digital signature	DSS/SHA or RSA/SHA	A hash code of a message is created using SHA-1. This message digest is encrypted using DSS or RSA with the sender's private key and included with the message.
Message encryption	CAST or IDEA or Three-key Triple DES with Diffie-Hellman or RSA	A message is encrypted using CAST-128 or IDEA or 3DES with a one-time session key generated by the sender. The session key is encrypted using Diffie-Hellman or RSA with the recipient's public key and included with the message.

PGP (Pretty Good Privacy)

- PGP Algorithms
  - Symmetric encryption:
    - DES, 3DES, AES and others.
  - Public key encryption of session keys:
    - RSA or ElGamal.
  - Hashing:
    - SHA-1, MD-5 and others.
  - Signature:
    - RSA, DSS, ECDSA and others.

PGP(Pretty Good Privacy)

- PGP use:
  - public keys for encrypting session keys / verifying signatures.
  - private keys for decrypting session keys / creating signatures.
- PGP adopts a trust model called the *web of trust.*
- No centralised authority
- Individuals sign one another's public keys, these "certificates" are stored along with keys in key rings.
- PGP computes a *trust level* for each public key in key ring.
- Users interpret trust level for themselves.

S/MIME

- Secure/Multipurpose Internet Mail Extensions
- A standard way for email encryption and signing
- IETF effort (RFCs 2632, 2633 – for version 3.0; RFCs 3850, 3851 for version 3.1; 5750, 5751 for version 3.2)
- Industry support
- Not a standalone software, a system that is to be supported by email clients
  - such as MS Outlook and Thunderbird
- S/MIME handles digital signatures
  - Also provides encryption

Quick E-mail History

- SMTP and RFC 822 (later RFC 5322)
  - SMTP is the email transfer protocol running over TCP
  - RFC 822/5322 defines the message format and headers
    - only ASCII messages (7-bit)
- MIME (Multipurpose Internet Mail Extensions)
  - content type
    - Almost any type of information can appear in an email message
  - transfer encoding
    - specifies how the message body is encoded into textual form (radix64 is common)
- S/MIME: Secure MIME
  - new content types, like signature, encrypted **data**

S/MIME

- Security enhancement to the MIME internet e-mail format
- There are five message header fields
  - MIME-Version
  - Content-Type
  - Content-Transferring Encoding
  - Content-ID
  - Content-Description

S/MIME Functionality

- S/MIME provides the following functions
  - Enveloped Data
    - Consists of encrypted content of any type of encrypted content encryption keys
  - Signed Data
    - Contains a digital signature
  - Clear-signed data
    - Encoded digital signature
  - Signed and enveloped data
    - Encrypted and Signed data

S/MIME Cryptographic Algorithms

- hash functions: switched to SHA256
- digital signatures: Mostly RSA is used
- session key encryption: Mostly RSA is used
- message encryption: Triple-DES, AES and others, but AES-128 is preferred
- Base64 (radix64) encoding is used for email compatibility (ASCII conversion)
- sender should know the capabilities of the receiving entity (public announcement or previously received messages from receiver)
  - otherwise sender takes a risk of sending unintelligible e-mail.

S/MIME – User Agent Role

- Key generation
  - Generating key with RSA
- Registration
  - Register a user's public key must be registered with a certification authority
- Certificate storage and retrieval
  - Access to a local list of certificates in order to verify incoming signatures and encrypt outgoing

S/MIME – Enhanced Security Services

- Signed receipts
  - The receiver returns a signed receipt back to the sender to verify the message arrived
- Security labels
  - Permission, priority or role of message being sent
- Secure mailing lists
  - Sending to multiple recipients at once securely by using a public key for the whole mailing list

Public-Key Digital Signature

- Same as authentication
  - The sender encrypts a message with his own private key
  - The receiver, by decrypting, verifies key possession

What is a Digital Signature ?

- A Digital Signature is the result of encrypting the Hash of the data to be exchanged.
- A Hash (or Message Digest) is the process of mathematically reducing a data stream down to a fixed length field.
- The Hash uniquely represents the original data.
- The probability of producing the same Hash with two sets of different data is <.001%.
- Signature Process is opposite to Encryption Process
  - Private Key is used to Sign (encrypt) Data
  - Public Key is used to verify (decrypt) Signature

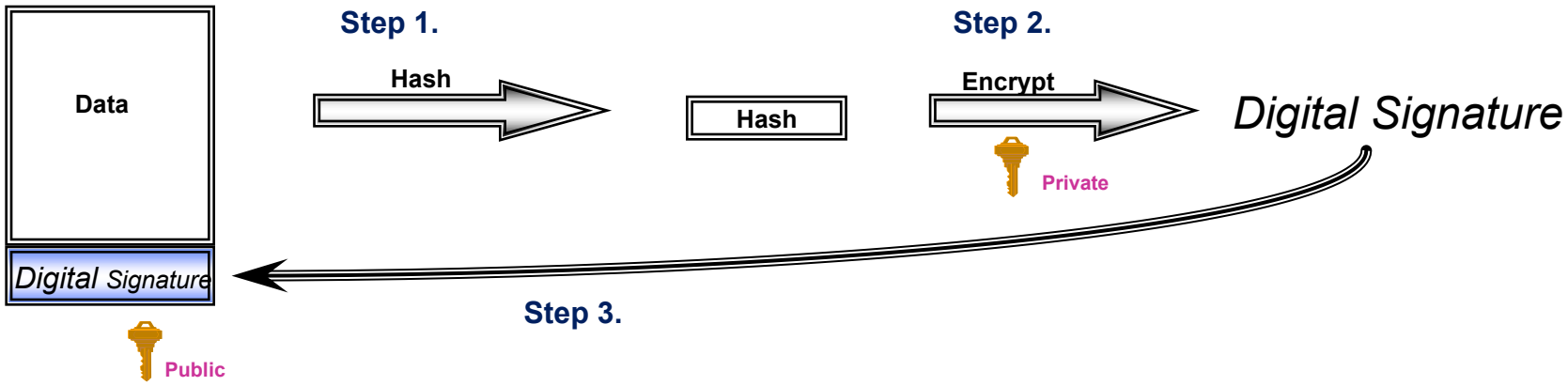
Digital Signatures

- It is possible to use the entire message, encrypted with the private key, as the digital signature
  - But, this is computationally expensive
  - And, anyone can then decrypt the original message
- Alternatively, a *digest* can be used
  - Should be short
  - Prevent decryption of the original message
  - Prevent modification of original message
  - Difficult to fake signature for
- If message authentication (integrity) is needed, we may use the hash code of the message
- If only source authentication is needed, a different message can be used (certificate)

Digital Signature Algorithm (DSA)

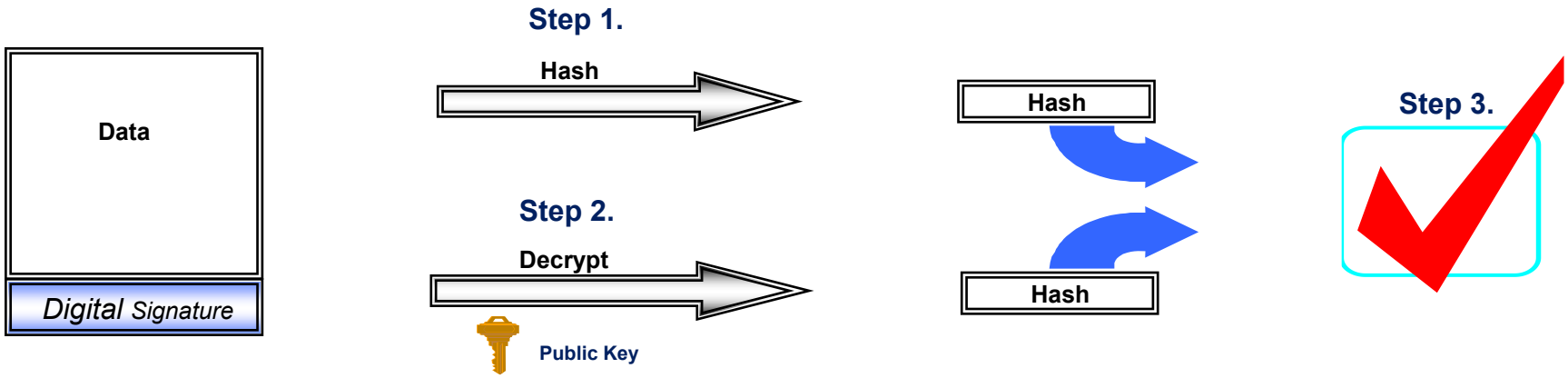
- Proposed in 1991 by NIST as a standard (DSS)
  - Based on difficulty of computing discrete logarithms (like Diffie-Hellman and El Gamal)
- Encountered resistance because RSA was already de-facto standard, and already drew significant investment
  - DSA cannot be used for encryption or key distribution
  - RSA is advantageous in most applications (exc. smart cards)
    - RSA is 10x faster in signature
    - DSA is faster in verification
  - Concerns about NSA backdoor (table can be built for some primes)
- Key size was increased from 512 to 2048 and 3072 bits
  - In DSA, the key size needs to be 4 times the security level
- DSA has an Elliptic Curve version
  - Faster to compute, and requires half the bits

Digital Signature Process



- Step 1. Hash (digest) the data using one of the supported Hashing algorithms, e.g., MD2, MD5, or SHA-1.
- Step 2. Encrypt the hashed data using the sender's private key.
- Step 3. Append the signature (and a copy of the sender's public key) to the end of the data that was signed.

Signature Verification Process



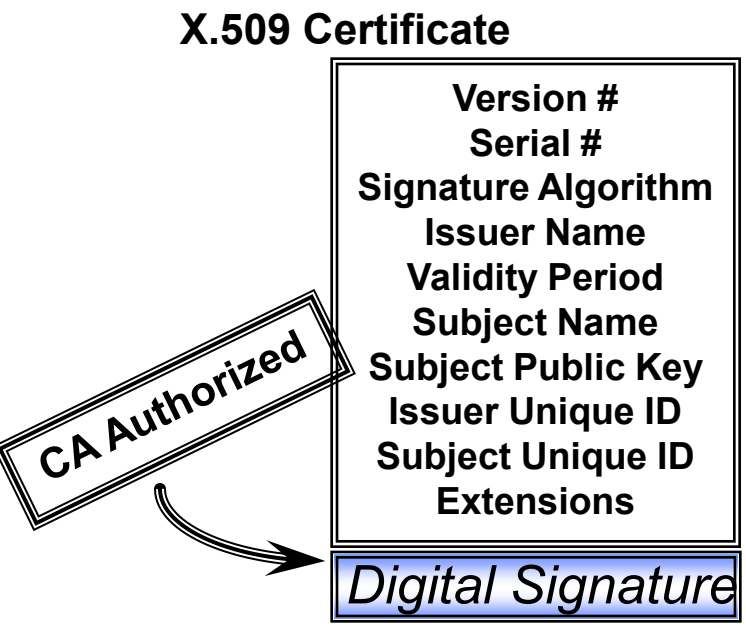
- Step 1. Hash the original data using the same hashing algorithm.
- Step 2. Decrypt the digital signature using the sender's public key. All digital signatures contain a copy of the signer's public key.
- Step 3. Compare the results of the hashing and the decryption. If the values match then the signature is verified. If the values do not match, then the data or signature was probably modified in transit.

Digital Certificates

- Before two parties exchange data using Public Key cryptography, each wants to be sure that the other party is authenticated
- Before B accepts a message with A's Digital Signature, B wants to be sure that the public key belongs to A and not to someone masquerading as A on an open network
- One way to be sure, is to use a trusted third party to authenticate that the public key belongs to A. Such a party is known as a Certification Authority (CA)
- Once A has provided proof of identity, the Certification Authority creates a message containing A's name and public key. This message is known as a Digital Certificate.

Digital Certificates

- **A Digital Certificate is simply an X.509 defined data structure with a Digital Signature. The data represents who owns the certificate, who signed the certificate, and other relevant information**

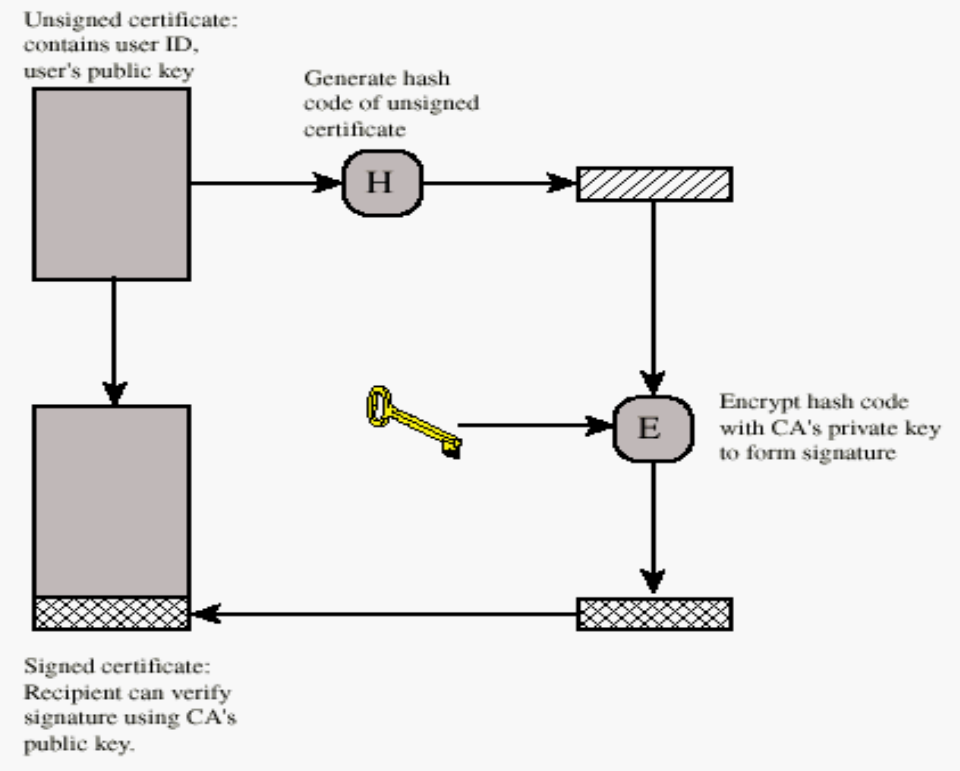


- When the signature is generated by a Certification Authority (CA), the signature can be viewed as trusted.
- Since the data is signed, it can not be altered without detection.
- Extensions can be used to tailor certificates to meet the needs of end applications.

Certificate Authority: Verifying the Public Key

- How to ensure that Charles doesn't pretend to be Bob by publishing a public-key for Bob. Then, using a Man-in-the-Middle attack, Charles can read the message and reencrypt-resend to Bob

  - Bob prepares certificate with his identifying information and his public key
  - The Certificate Authority (CA) verifies the details and sign Bob's certificate
  - Bob can publish the signed certificate



More on (Public) Key Management

- Alice may have more than one key
  - e.g., personal key and work key
- Where shall Alice store her keys?
  - Alice may not want to trust her work administrator with her personal banking key
- Distributed certification a la X.509
  - CA certifies Agents who certify organizations who certify others
- Distributed certification a la PGP
  - Alice will present her certificate with “introducers” who will vouch for her (“PKI parties”)
- Key Escrow
  - US American Escrowed Encryption Standard suggests that private keys be broken in half and kept by two Government agencies
  - Clipper – for cellular phone encryption
  - Capstone – for computer communication

Message Digest

- Message Digest or “Fingerprint”
  - → Condensed Representation
  - → Easy to generate for a given file.
- Computationally infeasible to produce two messages with same message digest
- Impossible to recreate a message given a message digest.
- Data Integrity and Comparison Checking
  - → Message Integrity Validation

Applications: One-way hash functions

- Public Key Algorithms
  - Password Logins
  - Encryption Key Management
  - Digital Signatures
- Integrity Checking
  - Virus and Malware Scanning
- Authentication
  - Secure Web Connections
    - (PGP, SSL, SSH, S/MIME)

Variants

- MD4 and MD5 by Ron Rivest (1990,1994)
- SHA-0, SHA-1 by NSA (1993, 1995)
- RIPEMD-160 (1996)
- SHA-2 (2002 – 224, 256, 385, 512)
- Whirlpool
- Tiger
- GOST-3411
- SHA-3

MD5

- MD5 represents the fifth iteration designed by Ronald Rivest (RSA)
- Others from other authors include Whirlpool and SHA
- MD5 is open-source and released under the GPL
- MD5 is optimized for use on 64-bit computers
- MD5 uses a short 128-bit hash
- MD5 has become a popular hashing tool through PHP
- PASSWORD HASHING
- Long messages that need an integrity check before encryption

History of MD5 Collisions

- 2004: first collision attack
  - The only difference between colliding messages is 128 random-looking bytes
- 2007: chosen-prefix collisions
  - For any prefix, can find colliding messages that have this prefix and differ up to 716 random-looking bytes
- 2008: rogue SSL certificates
  - Talk about this in more detail when discussing PKI
- 2012: MD5 collisions used in cyberwarfare
  - Flame malware uses an MD5 prefix collision to fake a Microsoft digital code signature

What is SHA

- SHA stands for Secure Hash Algorithm
- It is based off the Merkle-Dangard hash function
- There are 3 versions of it
  - SHA-0
  - SHA-1
  - SHA-2
  - SHA-3
- SHA-1 and SHA-2 were designed by the National Security Agency (NSA)

SHA-0

- SHA-0 is a 160-bit hash function
- It was briefly introduced in 1993 and quickly was revoked
- It was revoked due to flaws in the system
- Runs in 80 rounds

SHA-1

- SHA-1 is a modification of SHA-0 to correct those flaws that it produced
- Produces a 160-bit message digest (MD) that is on principle based on MD4.
- The MD is then inputted into a Digital Signature Algorithm (DSA)
- Which generates/verifies the signature for the message.
- Runs in 80 rounds.

SHA-1 (continued)

- Signing the MD makes the message more efficient since the MD is a smaller size then the message.
- The same hash function is needed to verify the message.
- SHA-1 was considered secure because it was computationally infeasible to find a message that corresponds to a given MD.
- It was also hard to find two different messages that produce the same MD.
- Any changes made to the messages will result in a different MD

Attacks on SHA-1

- Since SHA-1 is 160-bits
  - Birthday attack can find a collision in 2^{80} trials
  - February 15, 2005 Wang, Yin, and Yu have found collisions in 2^{69} trials
- Is SHA-1 broken?
  - Not quite – The resources needed to break it is enormous
    - Considered the same amount of time to factor 760-bit RSA modulus
  - Only well-funded intelligence agencies can do it within a reasonable amount of time

SHA-2

- SHA-2 consists of a set of 4 hash functions with the digests of (224, 256, 384, and 512-bits)
- In 2005 security flaws were identified – math weakness may exist
  - SHA-1 attacks have not worked on SHA-2
- SHA-224 and SHA-224 produces 32-bit words
- SHA-384 and SHA-512 produces 64-bit words
- They all use different shift amounts, constants, and number of rounds

SHA-2 (continued)

- SHA-224 and SHA-384 are truncated versions of 256 and 512-bit.
  - But with different initial values
- SHA-224 and SHA-256 runs in 64 rounds and SHA-384 and SHA-512 runs in 80 rounds
- Although SHA-2 has better security it is not as widely used as SHA-1

Attacks on SHA-2

- There are two meet-in-the-middle preimage attacks
  - The first one attacks SHA-256 41/64 rounds in $2^{253.5}$ time with a space complexity of 2^{16} - SHA-512 46/80 rounds in $2^{511.5}$ time and space complexity of 2^{3}
  - The second one attacks SHA-256 42/64 in $2^{251.7}$ time and 2^{12} space complexity – SHA-512 42/80 in 2^{502} time and space 2^{22}

Comparison of the SHA functions

Version	Output (bits)	Input (bits)	Block Size (bits)	Max Size/ Message	Word size	Rounds	Collisions ?
SHA-0	160	160	512	$2^{64}-1$	32	80	Yes
SHA-1	160	160	512	$2^{64}-1$	32	80	Yes (2^{51})
SHA-256/224	256/224	256	512	$2^{64}-1$	32	64	None
SHA-512/384	512/384	512	1024	$2^{128}-1$	64	80	None

What are SSL and TLS?

- SSL – Secure Socket Layer
- TLS – Transport Layer Security
- both provide a secure transport connection between applications (e.g., a web server and a browser)
- SSL was developed by Netscape
- SSL version 3.0 has been implemented in many web browsers (e.g., Netscape Navigator and MS Internet Explorer) and web servers and widely used on the Internet
- SSL v3.0 was specified in an Internet Draft (1996)
- it evolved into TLS specified in RFC 2246
- TLS can be viewed as SSL v3.1
- Encrypt the web traffic between two sites, so no one can listen in and get credit card numbers

SSL components

- SSL Handshake Protocol
  - negotiation of security algorithms and parameters
  - key exchange
  - server authentication and optionally client authentication
- SSL Record Protocol
  - fragmentation
  - compression
  - message authentication and integrity protection
  - encryption
- SSL Alert Protocol
  - error messages (fatal alerts and warnings)
- SSL Change Cipher Spec Protocol
  - a single message that indicates the end of the SSL handshake

The Implementation

- The secure web site includes a digital certificate signed by some certificate authority. The certificate includes the server name, its public key, IP number, and an expiration date. It is typically signed with a 1024 bit key by the CA

How It Works

- The browser reads the site certificate; if it is signed by one of the trusted certificate authorities, browser accepts the certificate as valid
- If the certificate is signed by some unknown certificate authority, Netscape will ask you if you want to trust the guy who signed it
- The browser negotiates a secure session using something like the following protocol:

1: A->B: hello
2: B->A: Hi, I'm Bob, bobs-certificate
3: A->B: prove it
4: B->A: Alice, This Is bob
{ digest[Alice, This Is Bob] } bobs-private-key
5: A->B: ok bob, here is a secret {secret} bobs-public-key
6: B->A: {some message}secret-key

How It Works

- Step 1: your browser introduces itself to the secure server
- Step 2: the server responds by sending back a message with the certificate included
- Step 3: Your browser tells the secure site to prove its identity, that it really is who it says it is.
- Step 4: The secure server proves who it is by creating a message for the browser, generating a “fingerprint” of that message, and encrypting the “fingerprint” with the private key that is matched to the public key in the certificate. The browser generates the “fingerprint” for the message itself, then decrypts the “fingerprint” generated by the server using the public key provided in the certificate.

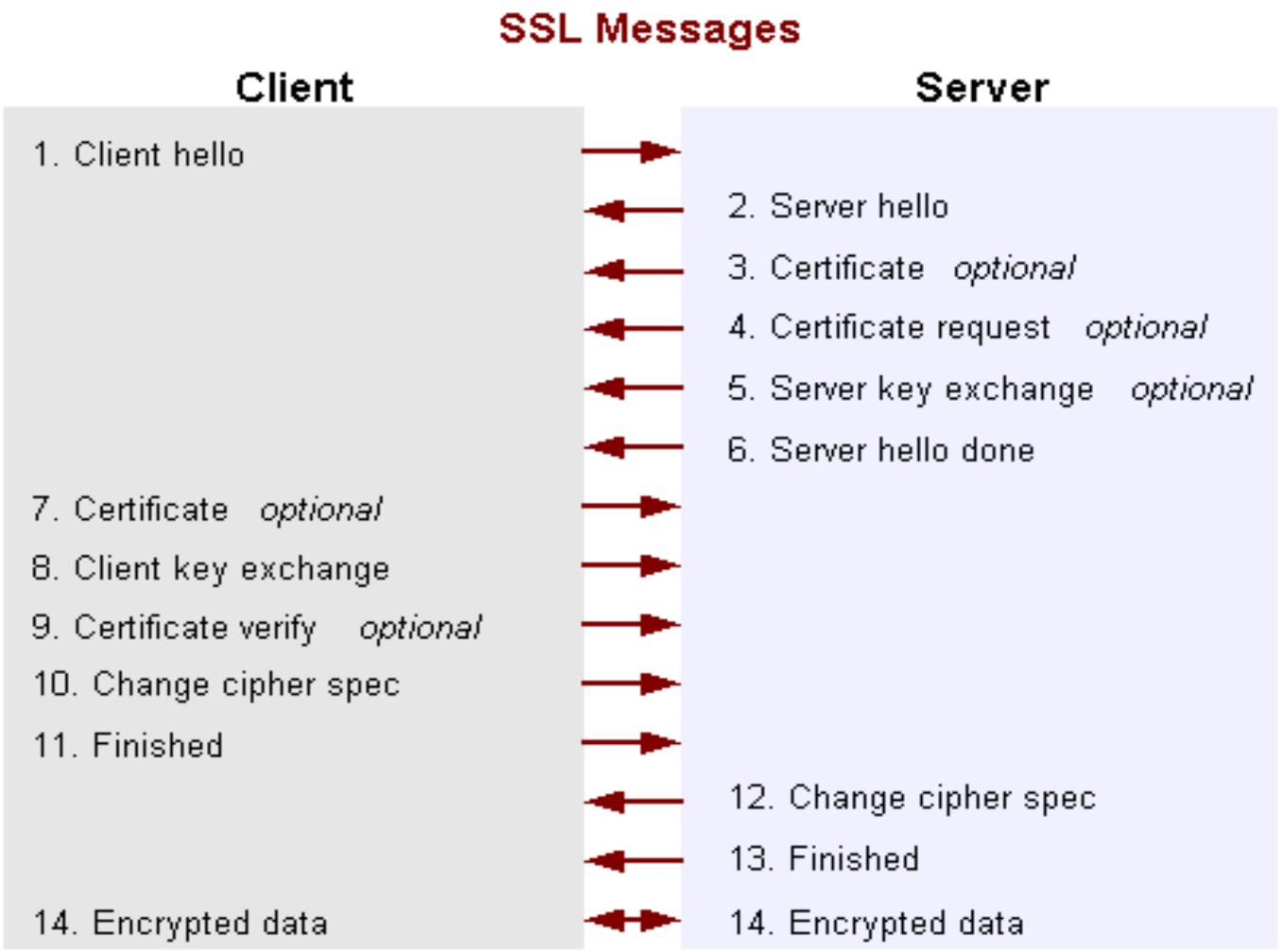
How It Works

- At this point the browser is sure that the server is how it says it is. It can send it secret messages encrypted with the public key provided in the certificate. The server (and only the server) can decrypt these messages, because only it has the private key.
- At this point what typically happens is that the browser generates a *session key* using a completely different encryption algorithm. A new session key is generated for every connection; this does not have to be a public key algorithm. You can use any encryption algorithm you like; usually a faster conventional, non-PK algorithm is used. This is usually 40 or 128 bits long in Netscape.

How It Works

- You'll use a completely different key for encrypting traffic to the web site every time you connect. This makes cracking communication more difficult; you need to discover the keys for every session rather than just one key.

How SSL Works: the *Handshake* in Detail



Supported key exchange methods

- RSA based (SSL_RSA_with...)
  - the secret key (pre-master secret) is encrypted with the server's public RSA key
  - the server's public key is made available to the client during the exchange
- fixed Diffie-Hellman (SSL_DH_RSA_with… or SSL_DH_DSS_with…)
  - the server has fix DH parameters contained in a certificate signed by a CA
  - the client may have fix DH parameters certified by a CA or it may send an unauthenticated one-time DH public value in the client_key_exchange message
- ephemeral Diffie-Hellman (SSL_DHE_RSA_with… or SSL_DHE_DSS_with…)
  - both the server and the client generate one-time DH parameters
  - the server signs its DH parameters with its private RSA or DSS key
  - the client may authenticate itself (if requested by the server) by signing the hash of the handshake messages with its private RSA or DSS key
- anonymous Diffie-Hellman
  - both the server and the client generate one-time DH parameters
  - they send their parameters to the peer without authentication
- Fortezza
  - Fortezza proprietary key exchange scheme

Server certificate and key exchange messages

- certificate
  - required for every key exchange method except for anonymous DH
  - contains one or a chain of X.509 certificates (up to a known root CA)
  - may contain
    - public RSA key suitable for encryption, or
    - public RSA or DSS key suitable for signing only, or
    - fix DH parameters
- server_key_exchange
  - sent only if the certificate does not contain enough information to complete the key exchange (e.g., the certificate contains an RSA signing key only)
  - may contain
    - public RSA key (exponent and modulus), or
    - DH parameters (p, g, public DH value), or
    - Fortezza parameters
  - digitally signed
    - if DSS: SHA-1 hash of (client_random | server_random | server_params) is signed
    - if RSA: MD5 hash and SHA-1 hash of (client_random | server_random | server_params) are concatenated and encrypted with the private RSA key

Reference

- Wikipedia http://en.wikipedia.org/wiki/Wiki
- News
  - http://www.zone-h.org/
  - http://www.zdnet.com.tw/news/software/0,2000085678,20143834,00.htm
  - http://udn.com/NEWS/WORLD/WOR3/6025617.shtml
  - http://tw.nextmedia.com/rnews/article/SecID/109/art_id/42470/IssueID/20110623
  - http://lieberman.senate.gov/index.cfm/news-events/news/2010/12/amazon-severs-ties-with-wikileaks
- All resources of the materials and pictures were partially retrieved from the Internet