

Topic 0

CIA

- 1. Confidentiality
 - Prevention of unauthorized disclosure of information
- 2. Integrity
 - Prevention of unauthorized modification of information or processes
 - Non-repudiation
 - Authentication
- 3. Availability
 - Prevention of unauthorized withholding of information or resources

Threat model

- The attacker's goals
- The attacker's capabilities

Trade-off in security

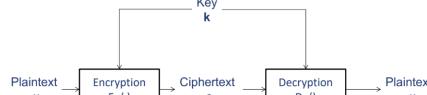
- Ease-of-use
- Performance
- Cost

Threat-Vulnerability-Control

- Threat: A set of circumstances that has the potential to cause harm (e.g. an attacker with control of the workstation in the LT could maliciously gather sensitive info like passwords)
- Vulnerability: A weakness in the system (e.g. anyone can reboot the workstation from USB or Disk to gain control)
- Control: A control, countermeasure, security mechanism is a mean to counter threats (e.g. restrict physical access to the workstation, disable USB booting)
- A threat is blocked by control of a vulnerability

Topic 1: Encryption

1.1 Definition: Encryption/decryption/keys



- A symmetric-key encryption scheme consists of encryption and decryption
- A cipher must be correct and secure
 - **Correctness:** For any plaintext x and key k , $D_k(E_k(x)) = x$
 - **Security:** Definition depends on the threat models. Informally, from the ciphertext, the eavesdropper is unable to derive useful information of the key k or the plaintext x , even if the eavesdropper can probe the system.
 - Probabilistic encryption: for the same x , there could be different c 's. But they all can be decrypted to the same x .

1.2 Security Model and Requirement

Threat model

- Attacker's goal
 - Total break (most difficult goal)
 - * Attacker wants to find the key
 - Partial break
 - * Attacker may want to decrypt a ciphertext but not interested in knowing the key
 - * Attacker may simply want to extract some info abt the plaintext (e.g. if it is a jpg or excel file)
 - Distinguishability (weakest goal)
 - * With some non-negligible probability of $> 1/2$, the attacker can correctly distinguish the ciphertexts of a given plaintext from the ciphertext of another given plaintext
- Attacker's capability
 - Ciphertext only attack
 - * Attacker is given a collection of ciphertext c . The attacker may know some properties of the plaintext (e.g. the plaintext is an English sentence)
 - Known plaintext attack
 - * The attacker is given a collection of plain-text m and their corresponding ciphertext c

- * Attacker might get this as they know the header or part of the plaintext
- Chosen plaintext attack (CPA)
 - * The attacker has access to an oracle. The attacker can choose and feed any plaintext m to the oracle and obtain the corresponding ciphertext c (all encrypted with the same key). The attacker can access the oracle many times, as long as within the attacker's compute power. He can see the ciphertext and then choose the next input. This black-box is an **encryption oracle**.
 - * e.g. attacker has access to a smartcard
 - * e.g. attacker can eavesdrop

- Chosen ciphertext attack (CCA2)
 - * Same as CPA but the attacker chooses the ciphertext and the black-box outputs the plaintext. The black-box is a **decryption oracle**.
 - * Padding oracle is a weaker form of a decryption oracle.

From defender's POV, want a cipher that can protect against the attacker with the highest capability. Cipher is secure against CCA2 (decryption oracle) \Rightarrow secure against CPA (encryption oracle)

1.3 Classical ciphers + illustration of attacks

1.3.1 Substitution cipher

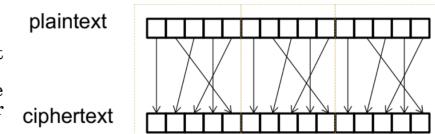
- Plaintext and ciphertext are both strings over a set of symbols U
- The key is a 1-1 onto func from U to U
- Key space: set of all possible keys
- Key space size: total number of possible keys
- Key size/length: number of bits required to represent a key
- Attacks

1. Exhaustive search (examine all possible keys 1 by 1)
 - Running time depends on size of key space
 - If the table size is 27, the key can be represented by a sequence of 27 symbols. The size of key space is $27!$. Exhaustive search needs to carry out 27! loops, which is infeasible using current compute power.

2. Known plaintext attack
 - Given sufficiently long ciphertext, the full table can be found
 - Substitution cipher is not secure under known plaintext attack.

3. Ciphertext only attack
 - Given that the attacker knows that the plaintext is an English sentence, he can do frequency analysis attack. The frequency of letters used in English is not uniform. Given a sufficiently long ciphertext, attacker may correctly guess the plaintext by mapping frequent characters in the ciphertext to the frequent character in English.

1.3.2 Permutation cipher



- AKA transposition cipher
- First group the plaintext into blocks of t characters, then apply a secret permutation to each block
- The key is the secret permutation, which is a 1-1 onto func e from $\{1, 2, \dots, t\}$ to $\{1, 2, \dots, t\}$. t can also be part of the key.
- Attack
 - Fails under known-plaintext attack
 - Easily broken under ciphertext only attack if the plaintext is English text

1.3.3 One Time Pad

Properties of xor:

- Commutative: $A \oplus B = B \oplus A$
- Associative: $A \oplus (B \oplus C) = (A \oplus B) \oplus C$
- Identity element: $A \oplus 0 = A$
- Self-inverse: $A \oplus A = 0$

One Time Pad

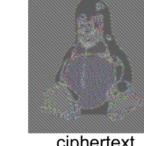
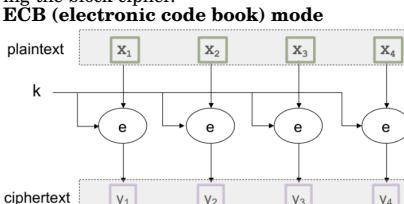
- Encryption: plaintext xor key bit by bit
- Decryption: ciphertext xor key bit by bit
- Key is only used once, so 1GB of plaintext would need a 1GB key to encrypt
- Security
 - From a pair of ciphertext and plaintext, attacker can derive the key but useless bc key won't be used anymore

1.4 Modern ciphers + recommended key length

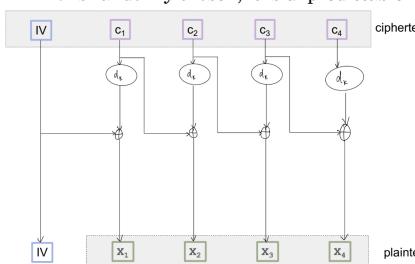
1.4.2 Block cipher & mode of operations

DES/AES are known as block ciphers. Block ciphers have a fixed size of input/output. AES: 128 bits (16 bytes). Large plaintext is divided into blocks before applying the block cipher.

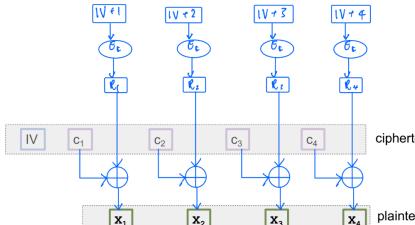
ECB (electronic code book) mode



- CBC (cipher block chaining) mode on AES**
- Initialization vector (IV) is an arbitrary value chosen during encryption, must be different in different encryptions.
 - In CBC mode, IV must be unpredictable, else it is susceptible to BEAST attack.
 - If IV is randomly chosen, it is unpredictable



CTR (counter) mode

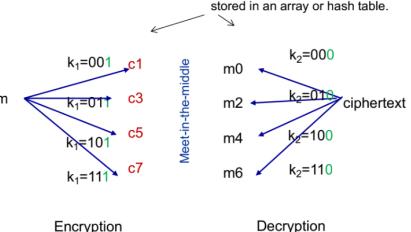
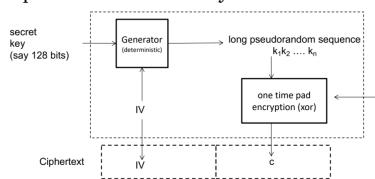


GCM mode (Galois/counter)

Authenticated encryption, ciphertext consists of extra tag for authentication. Secure in the presence of decryption oracle.

1.4.3 Stream cipher and IVs

Stream cipher is bit by bit. CTR mode is a "stream cipher" but it is not bit by bit.



- Last bit of k_1 fixed to 1, last bit of k_2 fixed to 0
- Perform meet-in-the-middle on the first 2 bits of k_1 and k_2

1.5.2 Padding Oracle

- Plaintext needs to be padded to split into blocks
- PKCS#7 is a padding standard

01
02 02
03 03 03
04 04 04 04
...
08 08 08 08 08 08 08

Padding oracle attack

Padding oracle attack on AES CBC mode

Attacker knows:

iv = $v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8$ Attacker knows the IV
c = $c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8$ Attacker knows the C
plaintext x = $x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8$ Attacker knows that the first 7 bytes must be 0. Attacker doesn't know x_1, x_2, x_3, x_4

$iv' = \begin{matrix} 0 & 0 & 0 & 0 & t & 07 & 07 & 07 \\ \oplus & & & & & & & \\ v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & v_7 & v_8 \end{matrix}$

$c = c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8$

$x = x_1, x_2, x_3, x_4, x_5, 03, 03, 03$

$x' = x_1, x_2, x_3, x_4, 04, 04, 04, t$

$$iv \oplus d(c) = 03$$

$$iv' \oplus d(c) = 04$$

xor the 2 tgt to get $iv' = 07 \oplus iv$

$iv' = iv \oplus 00 00 \dots t 07 07 07$

$d(C_5) \oplus t \oplus V_5 = 04$

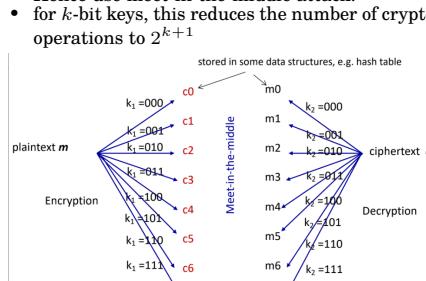
$d(C_5) \oplus V_5 \oplus t = 04$

$d(C_5) \oplus V_5 = x_5$

$x_5 \oplus t = 04$

$x_5 = 04 \oplus t$

Keep guessing t until padding oracle outputs YES, then we know x_5
To get next byte:



Tradeoff with time and space

- Certificate: A piece of document that binds a "name" to a "public key" & certified by a CA
- Certificate contains:
 - Name, public key, expiry date
 - Meta info: usages, type of crypto, name of CA, etc
 - CA's signature

4.2 PKI

4.2.1 Certificate & CA

- Certificates are used to distribute public keys. A CA issues certificates.
- CA: trusted authority that manages a directory of public keys
- CA has its own public-private key, some CA's public keys have been distributed securely through other means.
- OSes and browsers have pre-loaded CA's public keys, these CAs are known as root CAs.
- Other CA's public keys added through chain-of-trust
- A **certificate** is a digital document containing at least the following:
 - Name (e.g. alice@yahoo.com / bbc.com / *.bbc.com)
 - Public key of the owner
 - Time window that this cert is valid
 - Signature of CA

4.2.2 CA's chain-of-trust

Responsibility of CA

- Issue certificate
- Verify that information is correct (by checking that the applicant indeed owns the domain name)

Certificate chain-of-trust

- If Alice's cert is issued by CA#1, but Bob doesn't have the public key of CA#1, then Alice can send her email, her cert, and CA#1's cert (issued by root CA) to Bob
- Bob can now:
 - Verify CA#1's certificate using root CA's public key
 - Verify Alice's certificate using CA#1's public key
 - Verify Alice's email using Alice's public key
- Bob can also obtain CA#1's certificate from other sources

4.2.3 Revocation

- Reasons for revoking non-expired certs:
- Private key compromised (breaches, insider attack, vulnerability in choosing private keys)
 - System admin left an organization
 - Business entity closed
 - Issuing CA was compromised
- Verifier needs to check whether certificate is still valid, even if it has not expired. 2 approaches:
- Certificate Revocation List (CRL)
 - CA periodically signs and publishes a revocation list
 - Online Certificate Status Protocol (OCSP)
 - OCSP Responder validates a cert in question
 - Need online OCSP responder

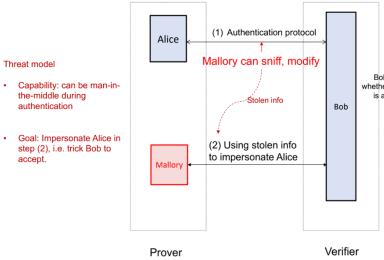
Recommendation: User periodically updates its local cache of revocation list, user does not need to check online to verify a cert

4.3 Limitations / attacks on PKI

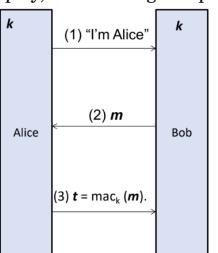
- Implementation bugs
 - Some browsers ignore substrings in the "name" field after the null characters when displaying it in the address bar but include them when verifying the cert.
 - Name in cert: www.comp.nus.edu.sg\0.hacker.com
 - Displayed in browser as: www.comp.nus.edu.sg
- Users think they are connected to www.comp.nus.edu.sg but are actually connected to www.comp.nus.edu.sg\0.hacker.com
- Abuse by CA
 - Rogue CA can forge any certificate
- Social engineering
 - Typoquatting

- An attacker registers for a domain name that looks similar to another website
- (b) Sub-domain
 - Attacker is the rightful owner of 134566.com
 - Attacker creates a sub domain luminus.nus.edu.sg.134566.com
 - Attacker can get a valid certificate

4.4 Protocol 1: Authentication

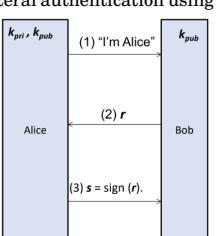


To prevent replay, use challenge-response



- m is picked at random
- k is shared secret between Alice and Bob
- Protocol only authenticates Alice. Unilateral authentication.

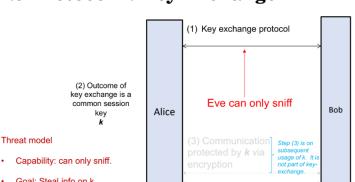
Can do unilateral authentication using PKC as well



- Alice may send Bob certificate if required (if Bob doesn't have Alice's public key)

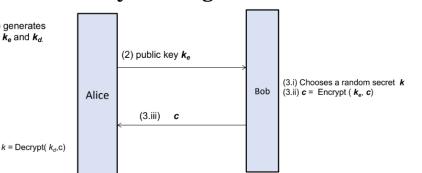
If only authentication is done, Mallory can intercept and take over the channel after Bob is convinced that he is communicating with Alice. Use authenticated key-exchange to get a new shared secret k known as session key. Then all subsequent communication will be encrypted + mac using k

4.5 Protocol 2: Key Exchange

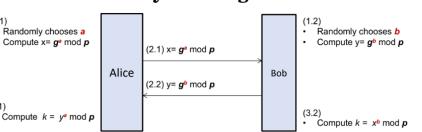


It is about confidentiality. Authenticity not considered.

PKC-based key exchange



Diffie-Hellman key exchange



- Alice and Bob have agreed on g and p . They are not secret and known to the public.
- Security relies on the CDH (computational diffie-hellman) assumption: given $g, p, x = g^a \% p$, it is computationally infeasible to find $k = g^{ab} \% p$

$$g = 2, \quad p = 23$$

$$(1.1) \text{ randomly chooses } a = 15$$

$$(2.1) x = g^a \text{ mod } p = 16$$

$$(2.2) y = g^b \text{ mod } p = 3$$

$$(3.1) \text{ Compute } k = y^a \text{ mod } p = 3^{15} \text{ mod } 23 = 12$$

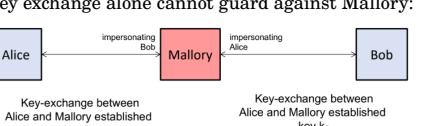
$$(3.2) \text{ Compute } k = x^b \text{ mod } p = 16^8 \text{ mod } 23 = 12$$

From 16, and 3, it is very difficult to get 12. (here, we are referring to very large, say 2000-bit integer)

Diffie-hellman meets forward secrecy requirement but PKC based method doesn't. TLS 1.3 mandates forward secrecy

4.6 Protocol 3: Authenticated Key Exchange

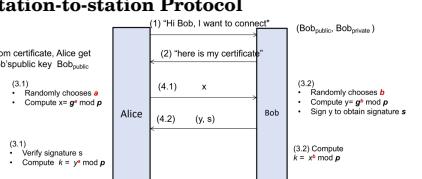
Key exchange alone cannot guard against Mallory:



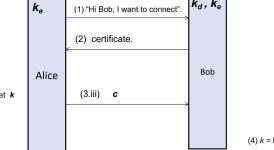
Need authenticated key-exchange, which can be easily obtained from existing key exchange.

- Diffie Hellman based:
 - Sign all communication using private key. Known as station-to-station protocol.
- PKC based:
 - Omit step (1) (generating public/private keys) and use Alice's existing public/private keys. Since only Alice has her private key, the entity that can correctly decrypt it must be Alice

Station-to-station Protocol



PKC-based Authenticated Key-exchange (AKA RSA-based)



Mutual Authenticated Key exchange

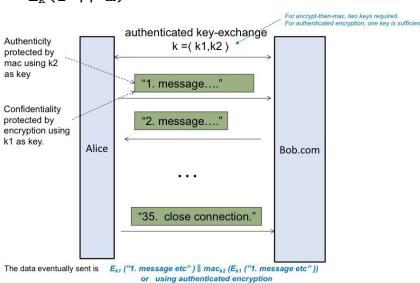
- Before:
 - Alice has a pair of public, private keys ($A_{public}, A_{private}$)
 - Bob has a pair of public, private keys ($B_{public}, B_{private}$)
 - Alice knows Bob's public key and vice versa. The two sets of keys are known as the long-term key or master key
- Carry out authenticated key exchange protocol (e.g. STS). If an entity is not authentic, the other will halt.
- After:
 - Both A and B obtain shared key k , known as session key
 - Security requirement:
 - (Authenticity) Alice is assured that she is communicating with an entity who knows $B_{private}$
 - (Authenticity) Bob is assured that he is communicating with an entity who knows $A_{private}$
 - (Confidentiality) Attacker is unable to get the session key

4.7 Securing Communication Channel

TLS

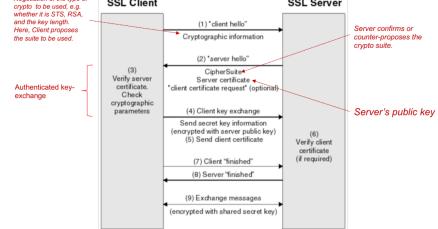
Alice wants to visit bob.com.

- Bob sends his certificate to Alice.
- Alice and bob.com carry out unilateral authenticated key exchange protocol with Bob's public/private key. After the protocol, both Bob and Alice obtain k , which could come in the form of $k = (k_1, k_2)$ where k_1 is the secret key of the MAC, and k_2 is the secret key of the symmetric-key encryption, or a single key k when authenticated encryption (e.g. GCM) is in use. These keys are called the session keys. By property of the protocol, Alice is convinced that only Bob and herself know the session key.
- Subsequent interactions between Alice and bob.com will be protected by the session keys and a sequence number. Suppose m_1, m_2, m_3 are the sequence of message exchanged, the actual data to be sent for m_i is $E_{k1}(i \parallel m) \parallel mac_{k2}(E_{k1}(i \parallel m))$ (still in use but not recommended)
- For GCM mode or other authenticated encryptions, the message to be sent is simply $E_k(i \parallel m)$



- SSL and TLS are protocols that secure communication using cryptographic means
- SSL is the predecessor of TLS
- HTTPS is built on top of TLS

TLS handshake



4.8 Forward Secrecy

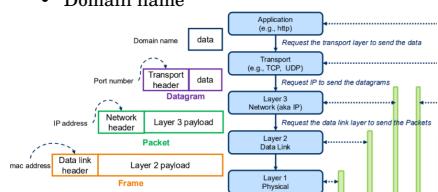
- If Eve cannot recover plaintext even though she knows the master key, then the authenticated key exchange achieves forward secrecy
- PKC-based authenticated key exchange does not achieve forward secrecy
 - Station-to-station key exchange achieves forward secrecy
 - If attacker can solve CDH, then forward secrecy of station-to-station is compromised

5. Network Security

Computer network: a collection of interconnected devices that can communicate with one another

Network layers

- Physical (wifi, ethernet)
- Data link
 - MAC address
- Network (IP)
 - IP address
- Transport (TCP, UDP)
 - Port number
- Application
 - Domain name



Transport + IP layer

- Often treated as one single layer
- Address of a communicating entity is an ip addr and a Port
- Each node in the network has 65535 ports
- Communication channel between 2 nodes is established by connecting 2 ports
- Src ip, src port to dest ip, dest port

5.3 Network attacker

Unless otherwise stated, MITM can sniff, spoof, modify, drop the header and payload

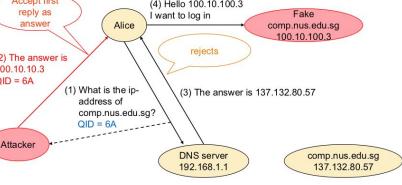
MITM in layer x means MITM along the layer x virtual connection (MITM can see and modify data unit of that layer)

- MITM in the physical layer
 - Tap into the internet cable, sniff the wireless communication in the cafe
- MITM in the link layer
 - Malicious coffee owner who offers the wifi
- MITM in the IP / Transport layer
 - ISP (e.g. Singtel)
- MITM in the application
 - Malware in the browser

5.4 DNS attacks

- ##### DNS query and answer
- Client sends a query to DNS server using UDP
 - DNS sends the answer back using UDP
 - The query contains a 16-bit number known as QID (query ID)
 - The response from the server must also contain a QID
 - If the QID in the query, the client rejects the answer
- ##### DNS spoofing

- Alice is using free cafe wifi and wants to visit and log in to comp.nus.edu.sg
- Consider an attacker who is also in the cafe. Since the wifi is not protected, the attacker can
 - Sniff data from the communication channel
 - Inject spoofed data into the communication channel
- The attacker cannot remove/modify data sent by Alice
- Attacker owns a webserver which is a spoofed NUS website



Cannot verify if response is coming from correct source or has not been modified - only matches QID
If cached into local DNS server (should have expiry), Alice will go to the fake website all the time
Denial of Service on DNS

- A DNS server can be a “single-point-of-failure” of the network
- DoS attacks can be launched on a web service instead of directly attacking the web server
- When DNS server is down, the web service is no longer reachable
- Countermeasures: redundant servers, rate limiting, etc.

5.5 Poisoning ARP table

Address resolution protocol (ARP)

- Resolution of IP address to mac address
- Data link layer
- When a device knows the IP address of the next hop router on the same network but needs the corresponding MAC address
- ARP resolves the router's IP address or its MAC address to allow packet forwarding at the data link layer

Switch

- Directs data packets between devices or nodes on the same local network using MAC addresses
- The switch keeps a table that associates the port to the mac addresses
- Switch does not understand and does not store IP addresses.

Address resolution Protocol (ARP) Table

- An ARP table is a database maintained by each device or nodes on a subnet
 - Stores mappings between IP addresses and MAC addresses

Eg, if N2 wants to send to IP addr 10.0.1.4,

- N2 looks up ARP table to find MAC addr
- N2 sends frame to switch, specifying destination MAC addr
- Switch looks up table to find port
- Switch redirects frame to correct port

If ARP table does not have info of a particular IP addr, N2 will broadcast an ARP request packet to all devices on the local network “Who has IP address X.X.X.X? Tell me your MAC addr”
The device with the requested IP address receives the ARP request and replies with an ARP reply packet. He sends the IP address and MAC address over. Then the requesting node will update its ARP table with the new IP-to-MAC address mapping. Any node can also broadcast the info or to a specific node even if there isn't a request

Attack: N1 wants to be MITM

- N1 informs N2 that N3's MAC address is N1's
- N1 informs N3 that N2's MAC address is N1's
- ARP tables of N2 and N3 are now poisoned
- All the frames will be sent to N1, and N1 can relay the frames, or modify the frames before relaying
- If N1 does not forward, then it is DoS.

DoS Attacks

- Attempt to disrupt the normal functioning of a targeted server, service or network ⇒ affect availability
- Many successful DoS attacks simply flood the victims with overwhelming requests/data
- DoS carried out by a large number of attackers is called DDoS (distributed denial of service)

Reflection Attack

- A type of DoS in which the attackers send requests to intermediate nodes, which in turn send overwhelming traffic to the victim.
- Attacker spoofs the victim's IP address as the source
- Intermediate nodes then send responses back to the spoofed IP (the victim)
- Indirect ⇒ more difficult to trace
- Preventive measures:
 - Most routers are configured to not broadcast by default
 - Configure firewalls to block incoming ICMP echo requests packets directed at broadcast addresses

Amplification Attack

- Variation of reflection attacks where the intermediate nodes response is significantly larger than the attacker's request
- A single request could trigger multiple responses from the intermediate nodes

5.7 Securing different layers

SSL / TLS

- Between layers 4 and 5
- On top of transport layer
- When an application wants to send data to the other end point, it first passes the data and the address to SSL / TLS
- SSL / TLS then protects the data using encryption and mac

IPSec

- Layer 3
- Mechanism that protects the IP layer and secures all IP traffic between endpoints
- Securing network connections between host-to-host, network-to-network (gateways), or network-to-host (gateway and host)

WPA2 (wifi protected access 2)

- Layers 1 and 2
- Protect data transmitted over wifi networks
- Offers protection in layer 2 and layer 2
- Data travelling between a wireless device and the access point is confidential and protected from eavesdropping
- Not all information in layer 2 are protected

VPN (virtual private network)

- Tunnel at layer 3
- Enable remote user to securely connect to private network
- VPN client and VPN server establish a connection, called a tunnel. After authentication and verification, establish session keys
- When Alice communicates with Bob, VPN client encrypts the entire payload and adds a new IP header
- From Bob's POV, Alice is communicating with the virtual node with IP address in NUS and does not know Alice's actual IP address

Which layer to protect

- By protecting lowest layer, can protect info in all layer but not feasible.
- Intermediate node need to access some higher layer info and sit in higher layer. Hence malicious intermediate node could be a MITM in higher layer.
- TLS / SSL + WPA2
- IPSeS is expensive to implement and difficult to deploy

5.6 Firewalls and IDS

In a computer network in an organization, some nodes contain more sensitive information than others. Some nodes are more “secure” (certain nodes operate in a more hostile environment). We need

to divide the network into segments and deny unnecessary access.

- Principle of least privilege: control access to the network
- Compartmentalization: keep things separated to limit the impact of any single failure or attack

Tools to control access to the network: firewall , IDS (intrusion detection system)

Firewall: Gatekeeper

prevents unauthorized access

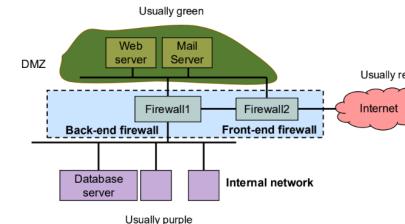
IDS: Watcher

that raises alerts by monitoring and analysing

Firewall and DMZ

- A firewall controls what traffic is allowed to enter the network (ingress filtering) or leave the network (egress filtering)
 - Firewall are devices or programs that control the flow of network traffic between networks or hosts that employ differing security postures.
- DMZ: Demilitarized zone**
 - A sub-network that exposes the organization's external service to the (untrusted) internet.
 - Separates an internal local area network (LAN) from untrusted external networks, usually the internet.
 - Adds an extra layer of security to an organization's internal network.

2-firewall setting



If the web / mail server in DMZ is compromised, the attacker still has to bypass the back-end firewall to reach sensitive data.

One firewall restricts external access, the other restricts access to the internal network

Packet filtering / screening

- Firewall's controls are achieved by “packets filtering”
- Packet filtering inspects every packet, typically only on the TCP / IP packet's header information (network & transport layer).
- If the payload is inspected, we call it deep packet inspection (DPI).
- Actions: Allow to pass, drop, reject, log, notify admin, modify

Types of Firewall

- Packet filters
 - Inspect only header (mainly IP packet's header)
 - Use: blocking traffic from certain IP or port
- Stateful inspection
 - Keep a state on previously received packets
 - E.g. counting number of connection a particular IP address has made in the past one hour
 - Use: blocking abnormal connection pattern or unauthorized session attempts
- Proxy
 - Act as intermediaries that fully receive inspect, and forward (possibly modify) packets between client and server
 - Use: block certain URLs or scan for malware in HTTP traffic

Intrusion Detection System (IDS)

- An IDS is a security tool or software that monitors computer systems and networks for signs of malicious activity, policy violations, or security breaches.

- An IDS system consists of a set of “sensors” that gather data such as logs, network packets, etc. Sensors can be deployed on hosts, or network router.
- Data are analyzed for intrusion either in real-time or after collection to detect suspicious patterns, attacks, or abnormal behaviour.

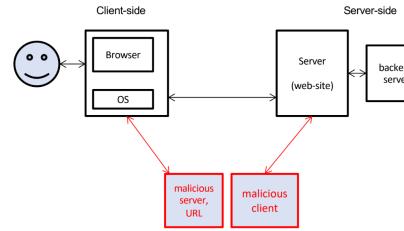
Three types of IDS

- Attack Signature Detection**
 - Looks for specific, well-defined patterns or “signatures” of known attacks in the data collected by sensors (e.g. using certain port number / source ip address)
- Abnormality Detection**
 - IDS attempts to detect abnormal patterns that deviate from the established “normal” behaviour of the network (e.g. sudden surge of packets with certain port number)
- Behavior-based IDS**
 - Anomaly detection that focuses on human behaviour (e.g. system keeps the profile of each user and detects any user who deviates from the profile)

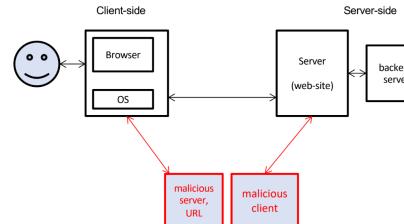
6. Web security

6.1 Background

Threat model 1: attackers as another end system



Threat model 2: attackers as MITM



- Attacker is MITM in the IP or lower layer
- Can gain MITM in a few ways
 - Cafe owner providing free wifi
 - DNS spoofing attack or ARP attack
 - Owner of the VPN server, last hop in TOR network

6.3 Misleading user on domain name

- Hostname could contain character that resembles “.”

- Address bar spoofing
 - Address bar is the only indicator of which url the page is rendering
 - If the address bar can be “modified” by the webpage, an attacker could trick the user to visit a malicious url X, while making the user falsely believe that the url is Y.
 - Poorly-designed browser can allow attacker to achieve that (e.g. Last time malicious page could overlay spoofed address bar on top of the actual bar. Now cannot anymore)

6.4 Cookie and same-origin policy

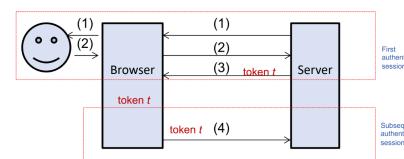
- The script in a web page A (identified by URL) can access cookies stored by another web page B (identified by URL) only if both A and B have the same origin.
Origin is defined as the combination of:

- Protocol

- Hostname

- Port number

Application of cookie: token-based authentication.



- (1) Authentication challenge (e.g. asking for password).

- (2) Authentication Response that involves the user.

- (3) Server sends a token t. Browser keeps the token

- (4) Browser presents the token t. Server verifies the token.

Choice of token If token t is a random number, server needs a db to store all issued tokens. Instead, can use

- (secure) MAC. Use random value or some meaningful information like expiry date, concat w mac computed using the server secret key
- Relyes on security of mac
- (insecure) Some meaningful info concat w a sequence number that can be predicted
- Relyes on obscurity. If attacker knows how it is generated, he can forge it

6.5 XSS (cross site scripting) attacks

The one where u put the <script></script>
In some websites, if the browser sends a text containing a substring s, then replying html sent by server will contain s. Can add a script in s.

Attack

- Attacker tricks a user to click on a url, which contains the target website, and a malicious script s.
- The request is sent to the server.
- The server constructs a response html. The response contains the script s.
- Browser renders the html page and runs s.
- Malicious script could deface the original webpage or steal cookie.

The attack exploits the client's trust of the server

Types of XSS

- Reflection (non-persistent)
- Stored XSS (persistent). Script s is stored in the targeted website (e.g. in forum page)

Defense

Input validation carried out by server

XSRF (cross site request forgery / sea surf)

Reverse of XSS. Exploits server's trust of client.

Example:

- Suppose client A is already authenticated by a targeted website www.bank.com and the site keeps a cookie as “token”.
- Attacker B tricks A to click on a url of S. The url maliciously requests for a service, e.g. transferring \$1000 to Bob's account
- Since cookie sent to S, A is already authenticated and transaction will be carried out.

Defense

Include authentication information in request as param in url

6.7 Other attacks

Web tracking, drive-by-download, pixel stealing, clickjacking, user interface redress attack, CAPTCHA, click fraud

Common simple implementation mistakes

Client side authentication / filtering, security credential embedded in public web pages, sever's secrets stored in cookies, configuration errors, URL as secrets (e.g. in password reset link or zoom link)