FIT5037 REVISION NOTES

Mid-term quiz

Rebecca

Contents

Week1

知识点：Computer Network and Communication, Security Models, Security Goals, Adversary capabilities, Cryptographic foundations

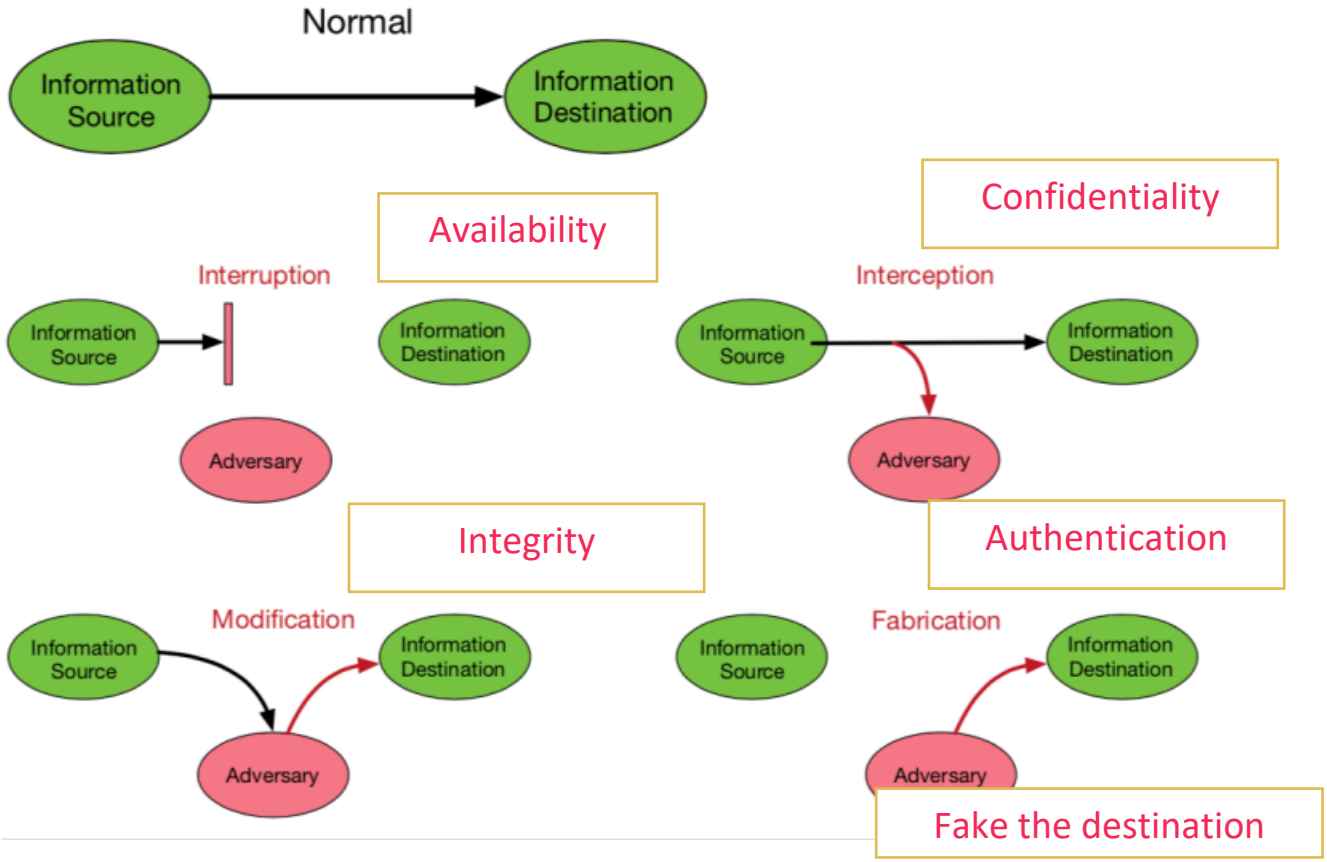
- Communication 要素

  computer networking: top down approach:

  mobile network -> golobal ISP -> home &regional ISP -> institutional network

- Hosts
  - massive connected devices (network edge)
- Communication links
  - fiber, copper, radio, satellite (physical media)
- Packet switches
  - router (network core), switch
- Protocols
  - control sending, receiving of msgs, e.g., TCP, IP, HTTP…
  - standard: RFC, IETF

- Types of Network Attacks



攻击 3 层：1） Network attack surface (e.g. network protocol vulnerabilities, disruption of communications links, intruder attacks etc.)

2）Software (e.g. Interfaces, SQL, Web forms)

3）Human (e.g. social engineering)

攻击分 2 种：1）Passive: eavesdropping, traffic analysis

2）Active: Dos, injection

防御 mechanism： preventaion(cryptography), detection(IDS), Recovery(forensics)

- Security Models: GOL-CAP
  - Confidentiality
  - Integrity

- Authentication
- Non-repudiation
- Availability

- Cryptography overview

	Symmetric	Asymmetric
Confidentiality	One-time Pad cipher, Stream ciphers, Block Ciphers	Encryption with Public Key
Integrity	Message Authentication Code(e.g. HMAC)	Digital Signature
Authentication	MAC+Nonce	Digital Signature + Nonce

Encryption -> goal: CONF

- Symmetric encryption
  - fast
  - has a key distribution problem
  - Block cipher: ARS
  - Stream cipher modes: RC4
- Public-key encryption
  - slow
  - has no key distribution problem
  - RSA, ElGamal

Stream cipher

- an encryption algorithm that uses a symmetric key to encrypt and decrypt a given amount of data.

One-time pad cipher:

- a randomly generated private key is used only once to encrypt a message and decrypted by the recipient using a matching one-time pad and key.

Problem with symmetric key ciphers:

Q: How 2 hosts can share secret key?

A: It they are far apart & channel insecure

Q: Why they have to share secret key?

A: Encrypt with secret key, decrypt need the same secret key

Q: How many keys need to be generated for n hosts if each of them communicates with the rest of n-1 hosts?

A: $\frac{n(n-1)}{2}$

Removing the Key Distribution Problem:

Q: Can remove the need to share the secret key?

A: then no key distribution problem

Q: Enc with one key, dec use a different key, is that possible?

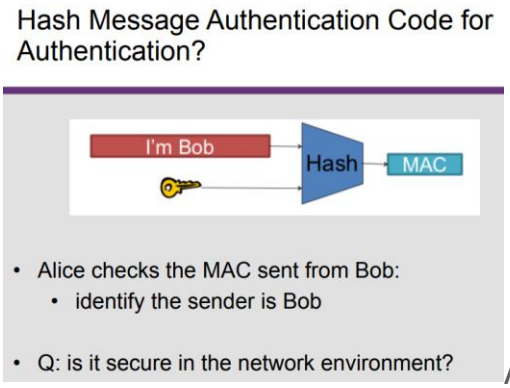
A: Yes, asymmetric cryptography

Symmetric Key: Hash message authentication code for integrity

- Cryptographic hash function:
    a. One way (once hashed, cannot be unhashed to retrieve the original data
    b. Collision resistance

Q: difference with encryption?

A: Hash functions always produce a fixed-size output irrespective of input size. While encryption needs keys for operation, standard hash functions don't, except for certain types like HMACs which use a secret key for authentication.



A: MAC Replay attack

Message Authentication Code(MAC)

- The sender forwards the message along with the MAC which uses a secret key to compress the message.
- The message is sent in clear with MAC.
- The secret key is only known to the sender and the intended recipient.
- cannot achieve Non-repudiation -> we cannot identify who sent this MAC - sender or recipient?

Ciphertext-only adversary

- the attacker knows ciphertext by eavesdropping

Known-plaintext adversary

- the attacker knows at least one sample of both plaintext and ciphertext.
- e.g. XOR cipher can be compromised to use plaintext XOR ciphertext to get the key

Chosen-plaintext attack

- the attacker can choose what plaintext will be encrypted. (Attacker can run the encryption functions for selected inputs)

Chosen-ciphertext Adversary

- the attacker can choose ciphertext to be decrypted.

RSA vs IND-CPA

Q: Can decryption be non-deterministic?

A: Yes, when considering probabilistic encryption schemes

Week2

知识点：Man-in-the-middle attacks on Pk encryption, PK infrastructure, Attacks on PKI

Man-in-the-middle attacks

Q: Can public-key encryption address a more powerful adversary who can intercept the communication?

A: True

Scenario:

- Mallory intercepts pk_alice and forwards pk_mallory to Bob
- Bob uses pk_mallory to encrypt messages as he cannot tell the difference
- Mallory intercepts Bob's encrypted messages and decrypts them.

Solution:

- Alice get a certificate from a trusted third party (certificate authority)
- 3rd party verify Alice's
- 3rd party issues a certificate with Alice's name & Alice's public key
- Alice sends the certificate to Bob
- Bob verifies the certificate using Alice's public key - make sure it's from Alice, not others.
- The certificate cannot be forged or tampered

Digital Signature

- Use the private key to encrypt the message -> generate the digital signature
- Use the public key to decrypt the digital signature -> verify the digital signature -> check if M == M'
- Can be used to defeat MITM attacks

Digital Certificate

X.509 Certificate Example

The CA's identity (Symantec)

The owner of the certificate (paypal)

```
Certificate:
Data:
  Serial Number:
          2c:d1:95:10:54:37:d0:de:4a:39:20:05:6a:f6:c2:7f
  Signature Algorithm: sha256WithRSAEncryption
  Issuer: C=US, O=Symantec Corporation, OU=Symantec Trust Network,
          CN=Symantec Class 3 EV SSL CA - G3
  Validity
     Not Before: Feb  2 00:00:00 2016 GMT
     Not After : Oct 30 23:59:59 2017 GMT
  Subject: 1.3.6.1.4.1.311.60.2.1.3=US/
           1.3.6.1.4.1.311.60.2.1.2=Delaware/
           businessCategory=Private Organization/
           serialNumber=3014267, C=US/
           postalCode=95131-2021, ST=California,
           L=San Jose/street=2211 N 1st St,
           O=PayPal, Inc., OU=CDN Support, CN=www.paypal.com
```

Public Key

CA's signature

```
  Subject Public Key Info:
      Public Key Algorithm: rsaEncryption
         Public-Key: (2048 bit)
         Modulus:
           00:da:43:c8:b3:a6:33:5d:83:c0:63:14:47:fd:6b:22:bd:
           bf:4e:a7:43:11:55:eb:20:8b:e4:61:13:ee:de:fe:c6:e2:
           ... (omitted) ...
           7a:15:00:c5:01:69:b5:10:16:a5:85:f8:fd:07:84:9a:c9:
         Exponent: 65537 (0x10001)
Signature Algorithm: sha256WithRSAEncryption
4b:a9:64:20:cc:77:0b:30:ab:69:50:d3:7f:de:dc:7c:e2:fb:93:84:fd:
78:a7:06:e8:14:03:99:c0:e4:4a:ef:c3:5d:15:2a:81:a1:b9:ff:dc:3a:
... (omitted) ...
fb:00:3e:7d:6a:de:cb:9f:ff:ef:8c:65:35:e4:22:b5:88:b2:48:32:1e:
```

Other fields:

- Usage period
- Serial number

Types of Digital Certificate (requires different levels of information -> vary the security level):

- Domain Validated Certificates (DV)
  - Most popular (e.g. personal website)
  - The CA verifies the domain records to check if the domain belongs to the applicant(before issuing the certificate).

        - Domain Control Validation is performed on the domain name in the certificate request (uses information in the WHOIS database, usually register with email).
    - Organizational Validated Certificated (OV)
        - CAs verify the following before issuing OV certificates:
            - Domain control validation.
            - Applicant's identity and address.
            - Applicant's link to the organisation.
            - Organisation's address.
            - Organisation's WHOIS record.
            - Callback on the organisation's verified telephone number
        - E.g. ABN
    - Extended Validated Certificates (EV)
        - CAs issuing EV certificates require documents that are legally signed from registration authorities- needs to verify the organization. (will verify the legal and proper standings of the organisation)
        - More infor required -> attacker need more effort

Digital signature in Digital Certificate

- CA generates a digital signature in the certificate using its private key. Anyone can verify the digital signature to see if the certificate has been modified using CA's public key.

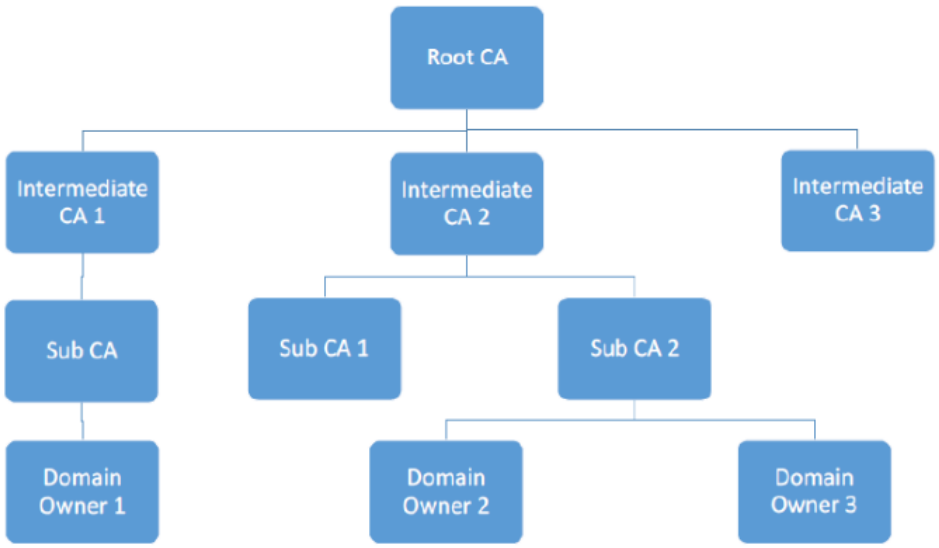
Certificate authorities

- Core functions:
  - Verify the subject
  - Signing digital certificates
    - CA generates a digital signature for the certificate using its private key
    - CA's private key cannot be compromised
    - Once the signature is applied, the certificate cannot be modified
    - Signatures can be verified by anyone with the CA's public key

  Q: If the ModelCA's certificate is self-signed, how do we verify it?

  A: There is no way to verify it

- Hierarchical structure:



- Root CA's certificates are self-signed.

  Q: How Root CA can be trusted?

  A: Public keys of CAs will be preinstalled in the OS, browser and other software.

        - Using Root CA's public key to verify Intermediate CA's certificate, Using Intermediate CA's public key to verify Sub CA's certificate.
    - Attack Scenario: Authentic Certificate
        - The attacker forwards the authentic certificate to Alice.
        - Alice finds the certificate is authentic so she uses the certificate's public key to encrypt the secret and send it to the "server".
        - The attacker will intercept the request but he cannot decrypt the secret because he doesn't know the private key
    - Attack Scenario: Fake Certificate
        - The attacker creates a fake certificate for the domain example.com with his own public key.
        - CA will not sign the certificate as the attacker is not the owner of example.com.
        - The attacker tries to self-sign the certificate and sends it to Alice.
        - Alice's browser will give the warning as it cannot find any trusted certificate to verify the received certificate.
    - Attack Scenario: Attacker's Certificate
        - The attacker has his own valid certificate.
        - The attacker sends his certificate to Alice.
        - Alice's browser checks if the certificate's subject field matches Alice's intent.

Attacks on PKI

- The Man-in-the-Middle proxy
    - The proxy creates a self-signed CA certificate installed on the user's browser.
    - The proxy will intercept the communication.
- Attacks on CA's verification process
- Attacks on CA's signing process: the private key is compromised

Q: How to protect the private key?

A: Use Hardware Security Model

- Attacks on Algorithms: Digital certificate depends on one-way hash and digital signature.
    - Use stronger algorithms
- Attacks on User confirmation: Some software does not compare these two pieces of information(the common name field inside the certificate and information provided or approved by user): security flaw

Week3

知识点：Internet Email Architecture and Protocols (SMTP, MIME), Email security, privacy-preserving email services with provider-supplied functions, email phishing attacks

Tutorial questions:

1. What are the principal services provided by PGP?

(a) Authentication (digital signature),

(b) confidentiality (message encryption),

(c) compression (ZIP),

(d) e-mail compatibility (Radix-64 conversion)

2. What is the utility of a detached signature?

A detached signature is useful in several contexts:

- A user may wish to maintain a separate signature log of all messages sent or received.
- A detached signature of an executable program can detect subsequent virus infection.
- Finally, detached signatures can be used when more than one party must sign a document, such as a legal contract. Each person's signature is independent and therefore is applied only to the document. Otherwise, signatures would have to be nested, with the second signer signing both the document and the first signature, and so on.

3. Why does PGP generate a signature before applying compression?

(a) It is preferable to sign an uncompressed message so that one can store only the uncompressed message together with the signature for future

verification. If one signed a compressed document, then it would be necessary either to store a compressed version of the message for later verification or to recompress the message when verification is required.

(b) Even if one were willing to generate dynamically a recompressed message for verification, PGP's compression algorithm presents a difficulty. The algorithm is not deterministic; various implementations of the algorithm achieve different tradeoffs in running speed versus compression ratio and, as a result, produce different compressed forms. However, these different compression algorithms are interoperable because any version of the algorithm can correctly decompress the output of any other version. Applying the hash function and signature after compression would constrain all PGP implementations to the same version of the compression algorithm.

4. What is R64 conversion?

- R64 converts a raw 8-bit binary stream to a stream of printable ASCII characters. Each group of three octets of binary data is mapped into four ASCII characters. 5. Why is R64 conversion useful for an e-mail application?

    (a) When cryptographic services are used, at least part of the block to be transmitted is comprised of raw binary data.

    (b) If only the signature service is used, then the message digest is encrypted (with the sender's private key).

    (c) If the confidentiality service is used, the message plus signature (if present) are encrypted (with a one-time symmetric key).

    Thus, part or all of the resulting block consists of a stream of arbitrary 8-bit octets. However, many electronic mail systems only permit the use of blocks consisting of ASCII text.

6. How does PGP use the concept of trust?

- PGP includes a facility for assigning a level of trust to individual signers and to keys.

- PGP computes a “key legitimacy field" for each public key certificate in the key ring.
- The higher the trust level, the higher the confidence the certificate is authentic.
- PGP uses two trust fields to maintain key legitimacy:
  - Signature trust field: indicates the degree to which the PGP user trusts the signer to certify public keys
  - Owner trust field: indicates the degree to which this public key is trusted to sign other public key certificates

7. What is MIME?

- MIME is an extension to the RFC 822 framework (latest RFC5322) that is intended to address some of the problems and limitations of the use of SMTP (Simple Mail Transfer Protocol) or some other mail transfer protocol for electronic mail.

8. (a) What is S/MIME?

- S/MIME (Secure/Multipurpose Internet Mail Extension) is a security enhancement to the MIME Internet e-mail format standard.

(b) What are the cryptographic functions used in S/MIME?

- S/MIME incorporates many algorithms for the various cryptographic functions it provides:
  - The DSA and RSA for digital signature.
  - RSA encryption or Diffie-Hellman key exchange for session key management.
  - MD5, SHA-1 and SHA2 for creating hash of the message for the Digital signature.
  - For message encryption it supports AES, triple DES. 40-bit RC2 is listed in original document however is not considered secure.
  - It creates a message authentication code using HMAC with SHA-1. Authenticated encryption modes of operation AES-GCM and AES-CCM are added as extension in RFC6033.

9. (a) What is DKIM?

- DomainKeys Identified Mail (DKIM) is a specification for cryptographically signing e-mail messages, permitting a signing domain to claim responsibility for a message in the mail stream.

(b) How is the DKIM e-mail authentication service different when compared to S/MIME or PGP?

- DKIM e-mail authentication service is different when compared to S/MIME or PGP as indicated below:
  - S/MIME needs both sender and receiver to employ S/MIME. Most of the S/MIME mail users, bulk of the incoming mail does not use S/MIME.
  - S/MIME (and PGP) signs only the message contents. Header information maybe compromised.
  - DKIM is not implemented in client programs (MUAs) and is therefore transparent to the user; the user need not take any action.
  - DKIM applies to all mail from cooperating domains.
  - DKIM allows good senders to prove that they did send a particular message and prevent forgers from masquerading as good senders.

Simple Mail Transfer Protocol (SMTP)

- text only
- limited to ASCII
- Size limit
- command only
- issues
  - no security - clear/plain messages

Email Threats

- Authenticity
  - unauthorised access
- Integrity

        - unauthorised modification of emails
    - Confidentiality
        - unauthorised disclosure of sensitive information
    - Availability
        - prevent users from sending/receiving emails

Email security

Q: Why not sufficient to rely on transport layer security (TLS) to secure emails?

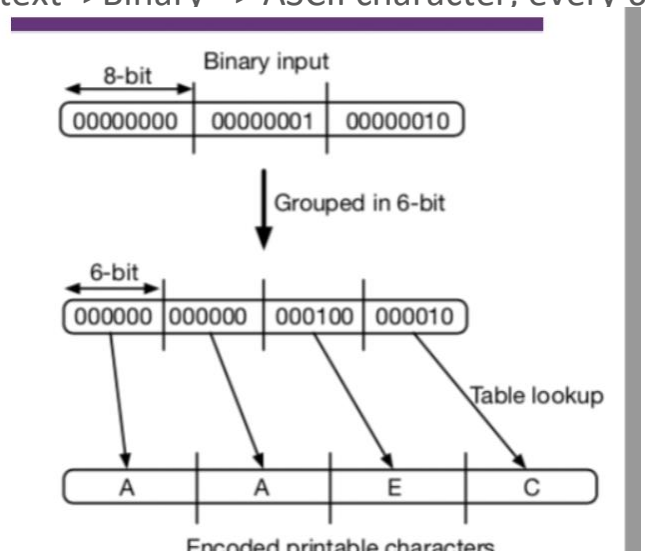
A: Only secures communication between two hops; all intermediate hops see plaintext (end to end encryption)

Pretty Good Privacy (PGP)

- key management
- AUTH + INT
    - Digital signature(Non-repudiation)
- CONF
    - Encryption
- Encryption steps
    - HASH(M) => digest
    - RSA(digest,secretKey) => digital signature
    - RSA is a public-key encryption especially used for signing.
    - Digital signature || M => certificate
    - Compress(certificate) => compressed certificate
    - AES(sessionKey, compressed certificate) => encrypted compressed certificate(using symmetric encryption)
    - RSA(sessionKey, ReceiverPublicKey) => encrypted sessionKey, used to transmit session key, another user can use his private key to get the session key.
    - Send to the recipient: encrypted compressed certificate || encrypted sessionKey
- Decryption steps

        - Get decrypted session key using the private key => sessionKey
        - Get decrypted compressed certificate using the seesionKey => compressed cetrificate
        - Get M||digital signature by decompressing the data
          => M||digital signature = DeCompress(compressed certificate)
        - Verify hash(M) and hash(M')
        - Hash(M') = decrypt the digital signature using the sender's public key.
    - Compression
        - PGP uses a ZIP compression algorithm, after applying the signature and before the encryption
        - Why does this order matter?
            - It is preferable to sign an uncompressed message so that the signature does not depend on the compression algorithm. If you do the compression first, the signatures will be different when different compression algorithms are applied.
            - This has the benefit of saving space both for e-mail transmission and for file storage.
            - Encryption after compression strengthens the encryption, since compression reduces redundancy in the message.
    - Radix-64 conversion
        - used for achieving compatibility with email protocols

- text=>Binary => ASCII character, every 6bits = 1 ASCII character



- Key management
- send key identifier (KeyID) instead of full pk for bandwidth efficiency
- users can have multiple key pairs
- two types of key rings to maintain
  - her own public/private key pairs

Timestamp	Key ID*	Public Key	Encrypted Private Key	User ID*
• • •	• • •	• • •	• • •	• • •
T_i	$PU_i \bmod 2^{64}$	PU_i	$E(H(P_i), PR_i)$	User i
• • •	• • •	• • •	• • •	• • •

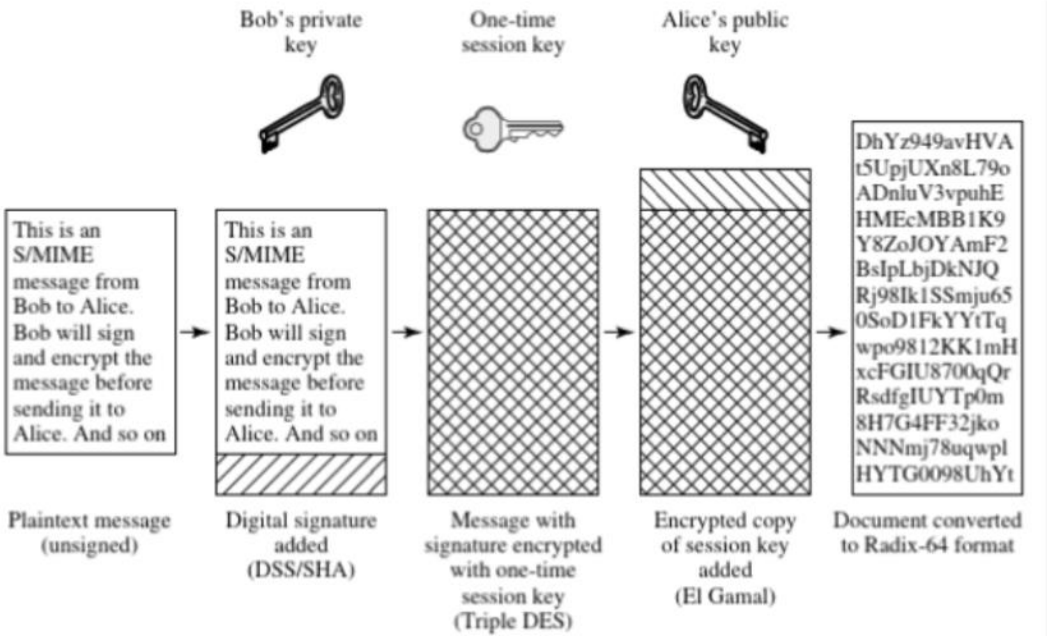
  - Store encrypted private keys instead of clear private keys
- public keys of other correspondents
  - all pk of other users known to the user, indexed by their key Ids.

Timestamp	Key ID*	Public Key	Owner Trust	User ID*
• • •	• • •	• • •	• • •	• • •
T_i	$PU_i \bmod 2^{64}$	PU_i	$trust\_flag_i$	User i
• • •	• • •	• • •	• • •	• • •

- Trust Model
  - not rely on certificate authorities (CAs)
  - every user is its own CA
    - Sign keys for users they know
  - forms web of trust
    - trusted keys signed
    - trusted keys other have signed if there is a chain of signatures to them
- Limitations
  - Must exchange public key
  - Target attacks against PGP keyIDs
  - prevent useful functionality like search, spam filtering, topic extraction..

MIME

- MIME(Secure/MultipurposeInternet Mail Extensions) supports different types of content.



- Each client has a list of trusted CA's certs and their own pairs & certs signed by trusted CA's

Domain Keys Identified Mail(DKIM)

- Tx(user)'s email must be signed by a secret key of the admin domain of Tx before leaving the domain.
- Transparent to user.
- Rx can verify the signature using the domain's public key.
- the public key will be stored in the DNS server, the signature will be attached to the header of the email.

Week4

知识点： Network Layer, Ip, Tor, IPSec architecture, protocols, applications, Encapsulating security payload (ESP) transport and tunnel modes of operations in IPSec

IPSec

- Network layer
- IPSec secures IP datagrams at the Internet layer according to the security policy of a communicating IP node, before forwarding them to the network interface layer.
- The intended receiving IP node verifies the datagrams according to the established security paramaeters and rejects any that have not been protected in accordance with the policy defined for such traffic.

Tutorial questions:

1. Give examples of applications of IPSec.
   Secure branch office connectivity over the Internet: A company can build a secure virtual private network over the Internet or over a public WAN. This enables a business to rely heavily on the Internet and reduce its need for private networks, saving costs and network management overhead .
   (a) Secure remote access over the Internet: An end user whose system is equipped with IP security protocols can make a local call to an Internet service provider (ISP) and gain secure access to a company network. This reduces the cost of toll charges for traveling employees and telecommuters.
   (b) Establishing extranet and intranet connectivity with partners: IPSec can be used to secure communication with other organizations, ensuring authentication and condentiality and providing a key exchange mechanism. Enhancing electronic commerce security: Even though some Web and electronic commerce applications have built-in security protocols, the use of IPSec enhances that security.

   IPSec can assure that:
   (a) A router or neighbour advertisement comes from an authorized router
   (b) A redirect message comes from the router to which the initial packet was sent
   (c) A routing update is not forged

2. What is the difference between transport mode and tunnel mode?
   Transport mode provides protection primarily for upper-layer protocols. That is, transport mode protection extends to the payload of an IP packet. Transport mode is meant to be used between two fixed hosts, or to put it another way, when the VPN endpoints are the final destinations of the traffic in the VPN. In particular, transport mode cannot be used to connect two networks or a network and a host.
   Tunnel mode provides protection to the entire IP packet. The typical use of tunnel mode is to connect either two networks or a host and a network: for example, a remote office network to a home office network. It is more flexible than transport mode, but this flexibility comes at the expense of increased bandwidth requirements.
3. What is a replay attack and how can IPSec prevent it?
   A replay attack is one in which an attacker obtains a copy of an authenticated packet and later transmits it to the intended destination. The receipt of duplicate, authenticated IP packets may disrupt service in some way or may have some other undesired consequence. The sequence number field in AH or ESP header associated with a particular SA is not duplicated. When a packet with duplicated sequence number with same SA is received, it is discarded.
4. Why does ESP include a padding field?
   It is included for the following reasons:
   (a) If an encryption algorithm requires the plaintext to be a multiple of some number of bytes (e.g., the multiple of a single block for a block cipher), the Padding field is used to expand the plaintext (consisting of the Payload Data, Padding, Pad Length, and Next Header fields) to the required length.
   (b) The ESP format requires that the Pad Length and Next Header fields be right aligned within a 32-bit word. Equivalently, the ciphertext must be an integer multiple of 32 bits. The Padding field is used to assure this alignment.
   (c) Additional padding may be added to provide partial traffic flow confidentiality by concealing the actual length of the payload
5. What are the distinctions between a Phase 1 and a Phase 2 Security Association?

What the Security Associations protect:
(a) Phase 1 Security Associations are used to protect IKE messages that are exchanged between two IKE peers, or security endpoints.
(b) Phase 2 Security Associations are used to protect IP traffic, as specified by the security policy for a specific type of traffic, between two data endpoints.

The attributes of the Security Associations:
(a) The phase 1 Security Association can specify only a single IP address for the security endpoints, while the phase 2 Security Association can specify a contiguous range or subnet as the data endpoint.
(b) The phase 1 Security Association must specify an encryption method, while encryption is optional for the phase 2 Security Association. An authentication method must be specified for both the phase 1 and phase 2 Security Association.

IPSec Services

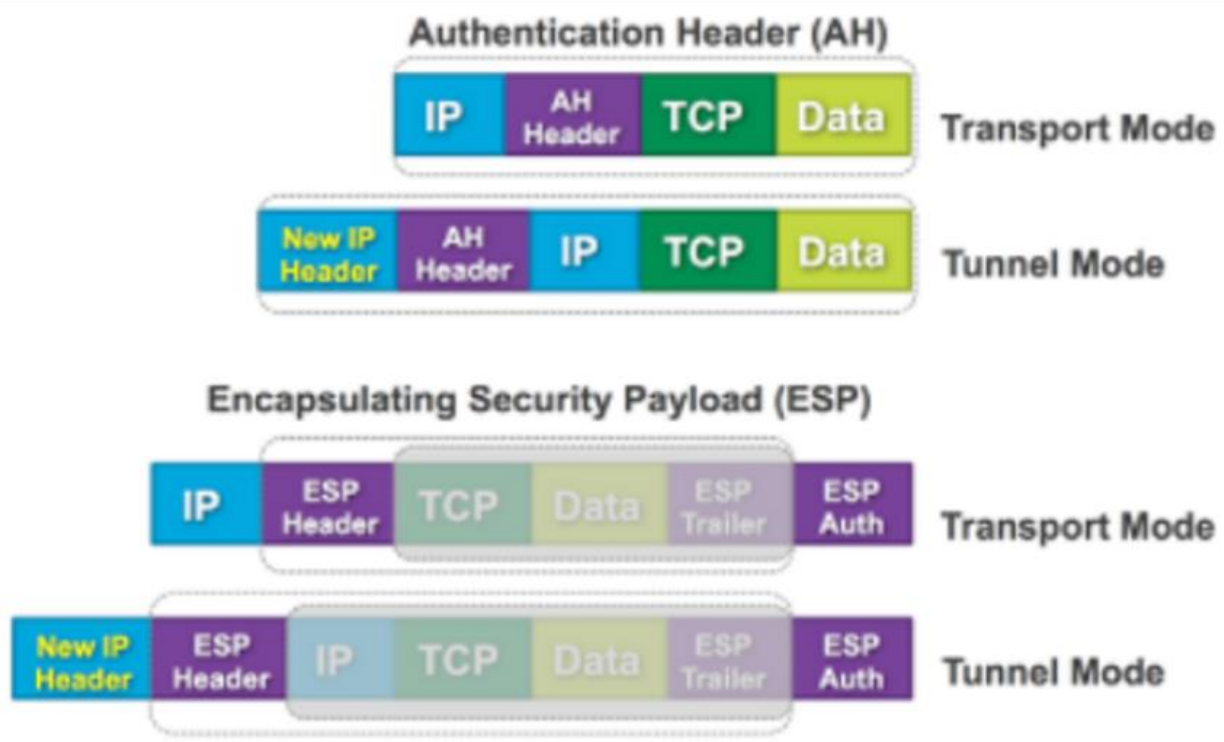
- Authentication
  - data origin authentication - verifies the claimed identify of the source data
- Integrity
  - connectionless integrity
    - detects tampering of individual ip datagrams
  - Anti-replay integrity
    - detects arrival of duplicate IP datagrams
- Confidentiality:
  - Protects data from unauthroised disclosure and provides a limited form of traffic-flow confidentiality.
    - conceals(hides) the source IP address, the destination IP address, the size of an IP datagram and frequency of communication
- Authentication Header(AH)

        - INT, data-origin AUTH, anti-replay(Sequence number), access control, no CONF
        - can only authenticate data(IP payload and selected portion of IP header), and cannot encrypt data.
    - Encapsulating Security Payload(ESP)
        - anti-replay, CONF
        - encrypts and authenticates IP payload but not IP header

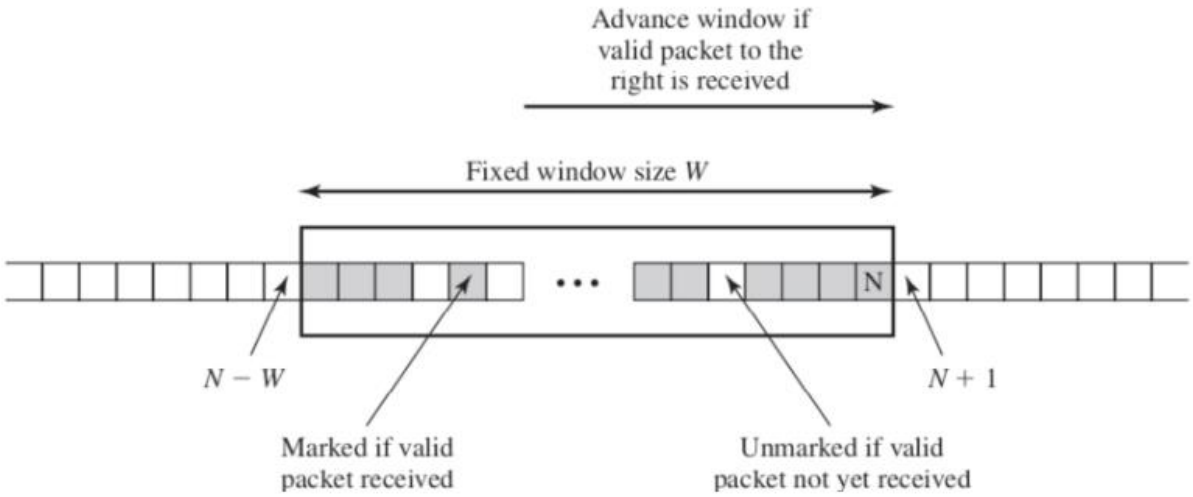
IPSec Modes

- Transport Mode
    - IP packet inserted with IPsec header
- Tunnel Mode
    - original packet preserved, new header added/prepended.
    - ESP/AH Header is immediately after the New IP Header, original IP packet preserved

	Transport Mode	Tunnel Mode
AH	Authenticates IP payload and selected portions of IP header	Authenticates entire inner IP packet (inner header plus IP payload) plus a selected portion of the outer IP header.
ESP	Encrypts IP payload. Allows traffic analysis	Encrypts entire inner IP packet. No routers on the way can examine the inner IP header.
ESP+AUTH	Encrypts IP payload. Authenticates IP payload but not IP header	Encrypts entire inner IP packet. Authenticates inner IP packet.



Anti-replay Services



Cases

- If sequence number received now is smaller than the most left size of the current window, drop the packet.
- If sequence number received now is within the current window range, check if packet has already been recived

- If sequence number received now is greater than the max size of the current window, make the sequence number received now the max edge, and calculate the min edge through N-W+1
- If the recipient sets N=60 and the window size W=64, range should be 0-60

Security Policy Database (SPD)

Protocol	Local IP	Port	Remote IP	Port	Action	Comment
UDP	1.2.3.101	500	*	500	BYPASS	IKE
ICMP	1.2.3.101	*	*	*	BYPASS	Error messages
*	1.2.3.101	*	1.2.3.0/24	*	PROTECT: ESP intransport-mode	Encrypt intranet traffic
TCP	1.2.3.101	*	1.2.4.10	80	PROTECT: ESP intransport-mode	Encrypt to server
TCP	1.2.3.101	*	1.2.4.10	443	BYPASS	TLS: avoid double encryption
*	1.2.3.101	*	1.2.4.0/24	*	DISCARD	Others in DMZ
*	1.2.3.101	*	*	*	BYPASS	Internet

3 IPSec policies:

- DISCARD
  - discard the packet
- PROTECT
  - protect the packet with AH and the ESP security protocols
- BYPASS
  - bypass the IPSec processing

Security Association Database(SAD)

- AH info
- ESP info
- Lifetime of this SA
- IPsec Protocol Mode
  - Transparent/tunnel

IPSec Architecture

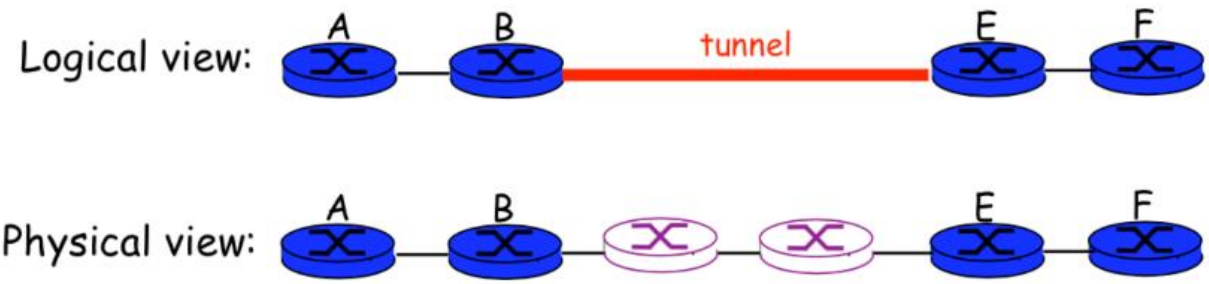
- Authentication Header(AH)

    - An extension header for message authentication
  - Encapsulating Security Payload(ESP)
    - provides encryption for combined encryption/msg INT
  - Internet Key Exchange(IKE)
    - the key management schemes for use with IPSec

Virtual Private Networks(VPN)

Isolation

Tunneling



TOR

- Components for Tor
  - Client
  - Server
  - Tor(onion) router: the special proxy relays the application adata
  - Directory server: servs holding Tor router information
- Process
  - The last router will see the clear data but not be knowing where the message from.
  - Every router only knows the predecessor and successor.
- Send Message from client:
  1. Client obtains a list of Tor nodes from a directory server
  2. Client picks a random path to destination server.
  3. Client negotiates an AES key with each router(every router has its own encryption key).
  4. Client encrypts message
     - C3 = Encrypt(K3, data||IP_Server)

- C2 = Encrypt(K2, C3 || IP_OR3)
- C1 = Encrypt(K1, C2 || IP_OR2)

5. Client sends an IP packet as: IP_Client || IP_OR1 || C1
   - send IP packet that is consist of current router/machine's IP, next destination's IP and encryped message

- Packet arrives at OR1 and OR1 performs:
  - C2 || IP_OR2 = DEC(K1, C1)
  - caches IP_Client, IP_OR2
  - sents an IP packet as: IP_OR1 || IP_OR2 || C2
- Packet arrives at OR2 and OR2 performs:
  - C3 || IP_OR3 = DEC(K2, C2)
  - caches IP_OR1 || IP_OR3
  - sents an IP packet as IP_OR2 || IP_OR3 || C3
- Packet arrvies at OR3 and OR3 performs:
  - IP_Server || data = DEC(K3,C3)
  - Caches IP_OR2, IP_Server
  - sents an IP packet as IP_OR3 || IP_SERVER || data