MODULE 5

KEY DISTRIBUTION AND SYSTEM SECURITY

KEY MANAGEMENT

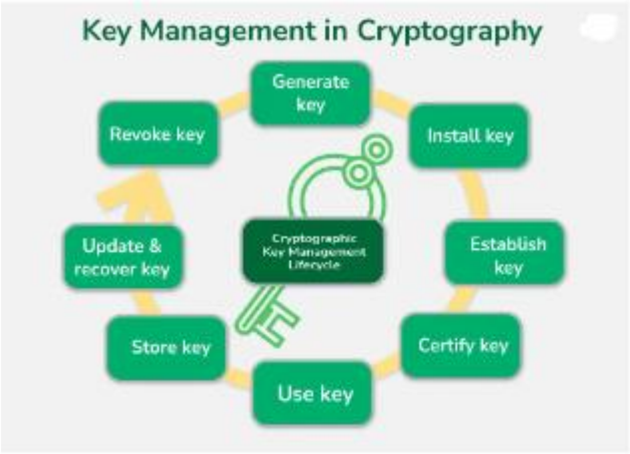
- It refers to the processes and procedures involved in generating, storing, distributing, and managing cryptographic keys used in cryptographic algorithms to protect sensitive data.
- It ensures that keys used to protect sensitive data are kept safe from unauthorized access or loss.
- Good key management helps maintain the security of encrypted information and is important for protecting digital assets from cyber threats.
- Effective key management is crucial for ensuring the confidentiality, integrity, and availability of encrypted information by securing cryptographic keys from unauthorized access, loss, or compromise.
- Cryptographic keys are special codes that protect information by locking (encrypting) and unlocking (decrypting) it.
- In **symmetric key cryptography**, a single shared key does both jobs, so the same key must be kept secret between users.
- In **asymmetric key cryptography**, there are two keys:
  - a public key that anyone can use to encrypt messages or verify signatures
  - a private key that only the owner uses to decrypt messages or create signatures.
- This makes it easier to share the public key openly while keeping the private key secret.
- These keys are crucial for secure communication, like when you visit a secure website (HTTPS), where they help encrypt your data and keep it safe from eavesdroppers and criminals.
- So, to manage these keys properly is vital to keep digital information secure and dependable.

Types of Key Management

- There are two aspects of Key Management:
  - Distribution of public keys.
  - Use of public-key encryption to distribute secrets.

Key Management Lifecycle

- The **key management lifecycle** outlines the stages through which cryptographic keys are generated, used, and eventually retired or destroyed.
- Proper management of these keys is critical to ensuring the security of cryptographic systems.



Key Management in Cryptography

1. **Key Generation**:
   - **Creation**: Keys are created using secure algorithms to ensure randomness and strength.
   - **Initialization**: Keys are initialized with specific parameters required for their intended use (e.g., length, algorithm).
2. **Key Distribution**:
   - **Sharing**: For symmetric keys, secure methods must be used to share the key between parties.
   - **Publication**: For asymmetric keys, the public key is shared openly, while the private key remains confidential.
3. **Key Storage**:
   - **Protection**: Keys must be stored securely, typically in hardware security modules (HSMs) or encrypted key stores, to prevent unauthorized access.
   - **Access Control**: Only authorized users or systems should be able to access keys.
4. **Key Usage**:
   - **Application**: Keys are used for their intended cryptographic functions, such as encrypting/decrypting data or signing/verifying messages.
   - **Monitoring**: Usage is monitored to detect any unusual or unauthorized activities.
5. **Key Rotation**:
   - **Updating**: Keys are periodically updated to reduce the risk of exposure or compromise.
   - **Re-Keying**: New keys are generated and distributed, replacing old ones while ensuring continuity of service.
6. **Key Revocation**:
   - **Invalidation**: Keys that are no longer secure or needed are invalidated.
   - **Revocation Notices**: For public keys, revocation certificates or notices are distributed to inform others that the key should no longer be trusted.
7. **Key Archival**:
   - **Storage**: Old keys are securely archived for future reference or compliance purposes.
   - **Access Restrictions**: Archived keys are kept in a secure location with restricted access.
8. **Key Destruction**:
   - **Erasure**: When keys are no longer needed, they are securely destroyed to prevent any possibility of recovery.
   - **Verification**: The destruction process is verified to ensure that no copies remain.

DISTRIBUTION OF SECRET KEYS USING SYMMETRIC & ASYMMETRIC ENCRYPTION

Distribution Of Secret Keys Using Symmetric

- In symmetric key cryptography, both parties must possess a secret key which they must exchange prior to using any encryption.

- Distribution of secret keys has been problematic until recently, because it involved face-to-face meeting, use of a trusted courier, or sending the key through an existing encryption channel.

- When two parties share the same key (i.e. symmetric key) that protect from access by others, the process between two parties that exchanges that key called as symmetric key distribution.

- If two person wants to communicates with each other via messages or exchange data without interference of other.

- Two parties/person A and B achieved the key distribution in various ways:
  1. A can select a key and physically deliver it to B.
  2. A third party can select the key and physically deliver it to A and B.
  3. If A and B have previously and recently used a key, one party can transmit the new key to the other, encrypted using the old key.
  4. If A and B each has an encrypted connection to a third-party C, C can deliver a key on the encrypted links to A and B.

- 1 and 2 calls for manual delivery of a key to the users. In manual delivery of key is difficult in a wide-area distributed system.
- 3 is possibility for either link encryption or end-to-end encryption, but if an attacker ever succeeds in gaining access to one key, then all subsequent keys will be revealed.
- For end-to-end encryption some variation on 4 has been widely adopted. In this scheme, a key distribution centre responsible for distributing keys to pairs of users (hosts, processes, applications) as needed. Each user must share a unique key with the distribution centre for purposes of key distribution.

Key distribution Scenario

- The key distribution concept can be deployed in a number of ways. A typical scenario is :

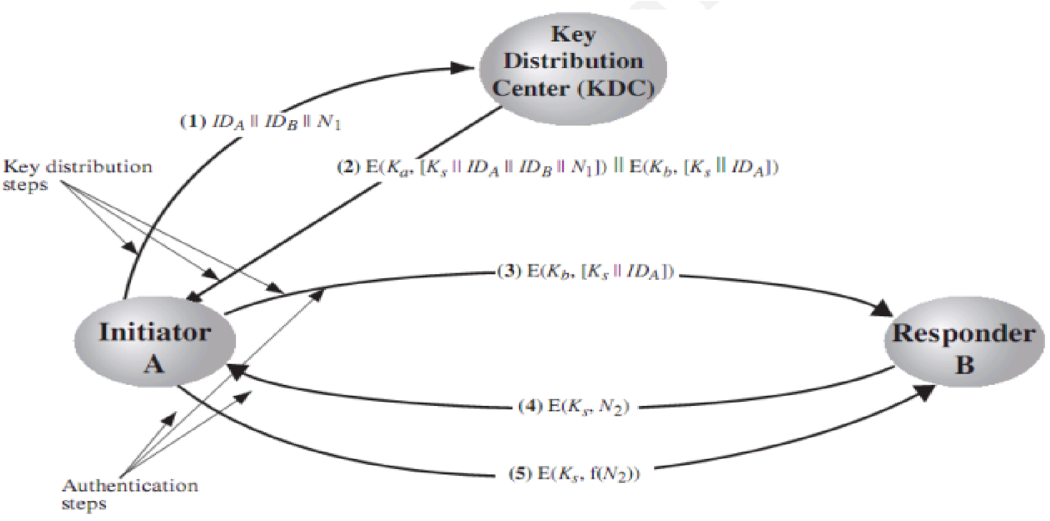
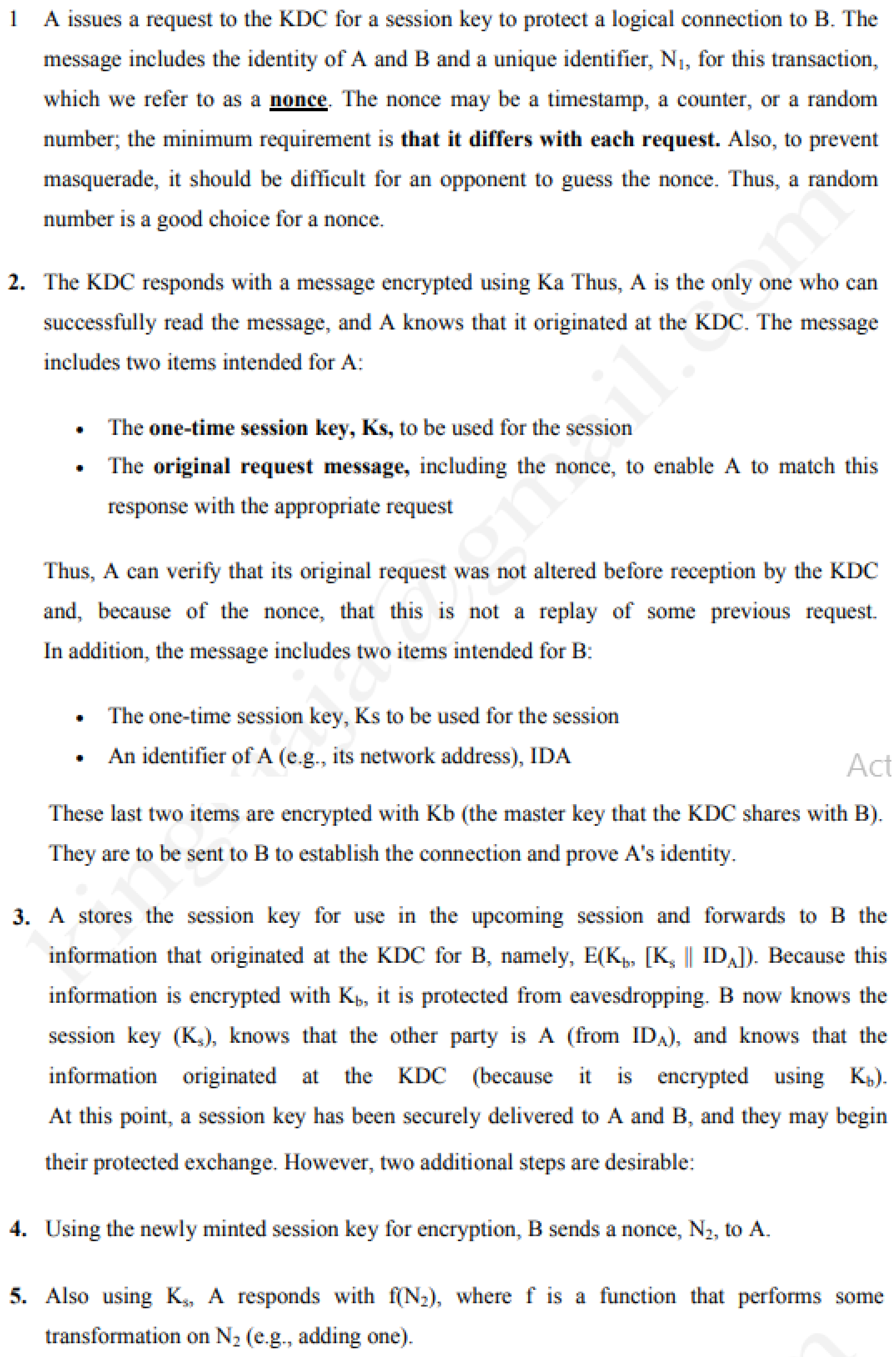

Key Distribution Center (KDC)
(1) $ID_A \| ID_B \| N_1$
Key distribution steps
(2) $E(K_a, [K_s \| ID_A \| ID_B \| N_1]) \| E(K_b, [K_s \| ID_A])$
(3) $E(K_b, [K_s \| ID_A])$
Initiator A
Responder B
(4) $E(K_s, N_2)$
(5) $E(K_s, f(N_2))$
Authentication steps

Figure 14.3 Key Distribution Scenario

- The scenario assumes that each user shares a unique master key with the key distribution centre (KDC).
- Let us assume that user A wishes to establish a logical connection with B and requires a one-time session key to protect the data transmitted over the connection. User A has a master key, Ka, known only to itself and the KDC; similarly, User B shares the master key K_b with the KDC.



1 A issues a request to the KDC for a session key to protect a logical connection to B. The message includes the identity of A and B and a unique identifier, N_1, for this transaction, which we refer to as a **nonce**. The nonce may be a timestamp, a counter, or a random number; the minimum requirement is **that it differs with each request.** Also, to prevent masquerade, it should be difficult for an opponent to guess the nonce. Thus, a random number is a good choice for a nonce.

2. The KDC responds with a message encrypted using Ka Thus, A is the only one who can successfully read the message, and A knows that it originated at the KDC. The message includes two items intended for A:

- The **one-time session key, Ks,** to be used for the session
- The **original request message,** including the nonce, to enable A to match this response with the appropriate request

Thus, A can verify that its original request was not altered before reception by the KDC and, because of the nonce, that this is not a replay of some previous request. In addition, the message includes two items intended for B:

- The one-time session key, Ks to be used for the session
- An identifier of A (e.g., its network address), IDA

Act

These last two items are encrypted with Kb (the master key that the KDC shares with B). They are to be sent to B to establish the connection and prove A's identity.

3. A stores the session key for use in the upcoming session and forwards to B the information that originated at the KDC for B, namely, $E(K_b, [K_s \| ID_A])$. Because this information is encrypted with K_b, it is protected from eavesdropping. B now knows the session key (K_s), knows that the other party is A (from ID_A), and knows that the information originated at the KDC (because it is encrypted using K_b). At this point, a session key has been securely delivered to A and B, and they may begin their protected exchange. However, two additional steps are desirable:

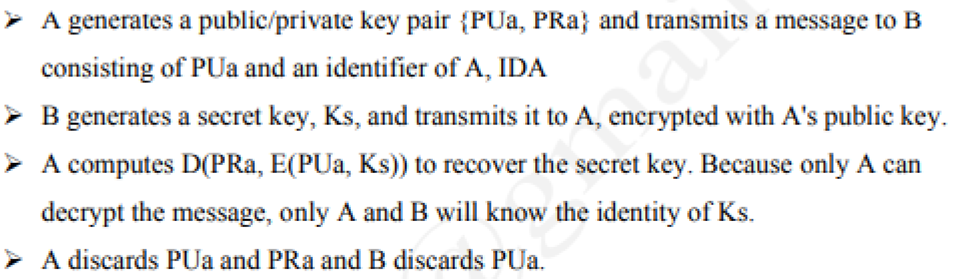
4. Using the newly minted session key for encryption, B sends a nonce, N_2, to A.

5. Also using K_s, A responds with $f(N_2)$, where f is a function that performs some transformation on N_2 (e.g., adding one).

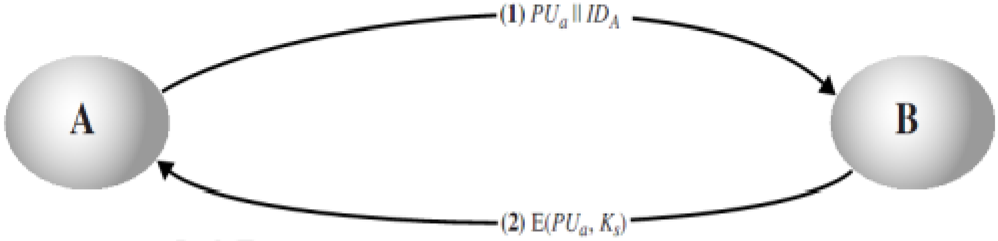
Distribution Of Secret Keys Using Symmetric

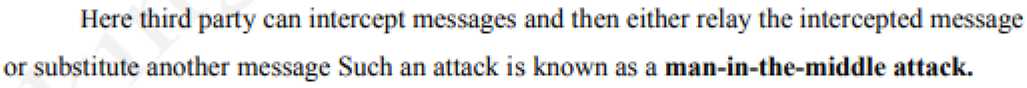
- Once public key have been distributed or have become accessible, secure communication that thwarts eavesdropping, tampering or both, is possible
- Public key encryption provides for the distribution of secret keys to be used for conventional encryption

Simple Secret Key Distribution



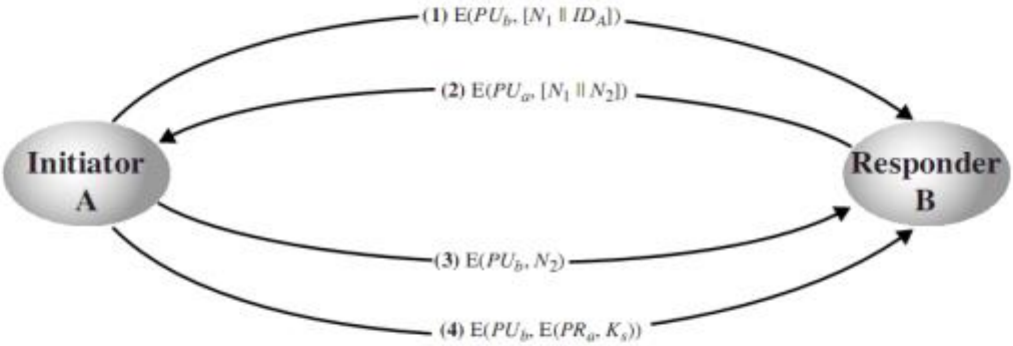
- ➢ A generates a public/private key pair {PUa, PRa} and transmits a message to B consisting of PUa and an identifier of A, IDA
- ➢ B generates a secret key, Ks, and transmits it to A, encrypted with A's public key.
- ➢ A computes D(PRa, E(PUa, Ks)) to recover the secret key. Because only A can decrypt the message, only A and B will know the identity of Ks.
- ➢ A discards PUa and PRa and B discards PUa.

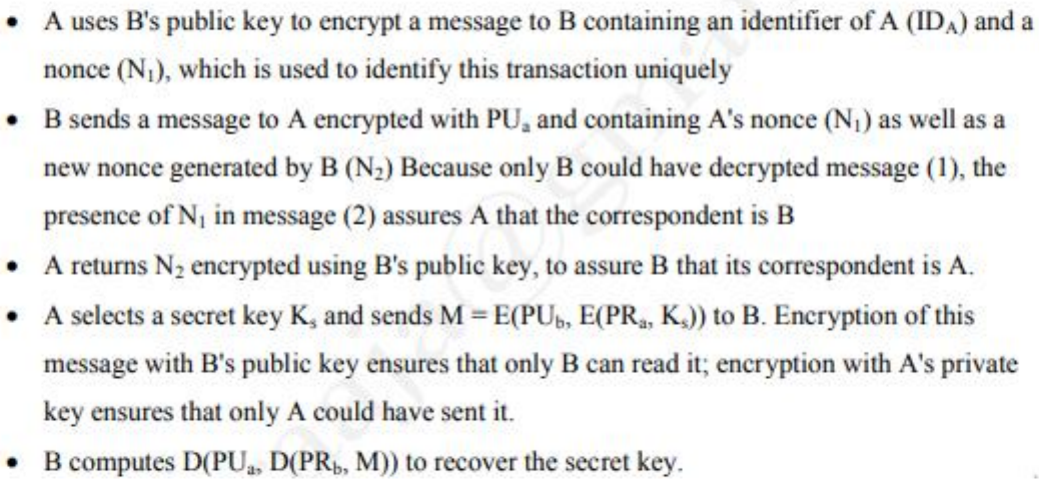




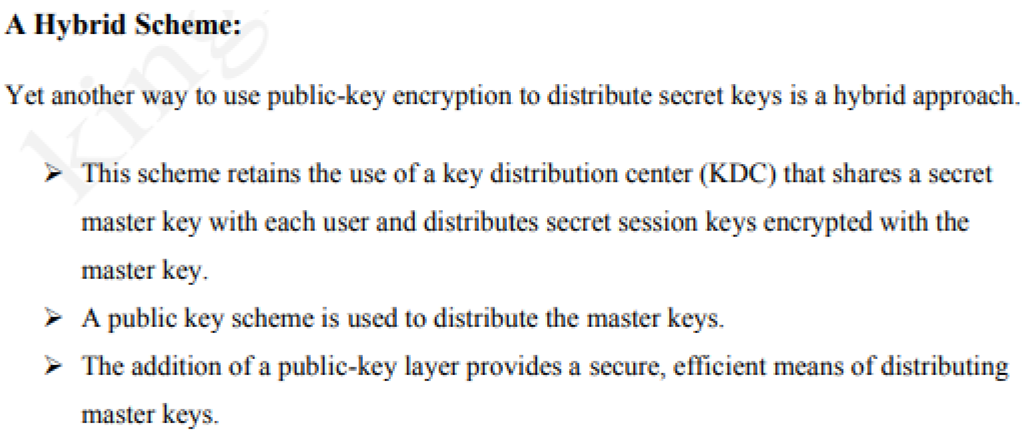
Here third party can intercept messages and then either relay the intercepted message or substitute another message Such an attack is known as a **man-in-the-middle attack.**

Secret Key Distribution with Confidentiality and Authentication:


(1) E(PU_b, [$N_1 \| ID_A$])
(2) E(PU_a, [$N_1 \| N_2$])
Initiator A
Responder B
(3) E(PU_b, N_2)
(4) E(PU_b, E(PR_a, K_s))



- A uses B's public key to encrypt a message to B containing an identifier of A (ID_A) and a nonce (N_1), which is used to identify this transaction uniquely
- B sends a message to A encrypted with PU_a and containing A's nonce (N_1) as well as a new nonce generated by B (N_2) Because only B could have decrypted message (1), the presence of N_1 in message (2) assures A that the correspondent is B
- A returns N_2 encrypted using B's public key, to assure B that its correspondent is A.
- A selects a secret key K_s and sends M = E(PU_b, E(PR_a, K_s)) to B. Encryption of this message with B's public key ensures that only B can read it; encryption with A's private key ensures that only A could have sent it.
- B computes D(PU_a, D(PR_b, M)) to recover the secret key.



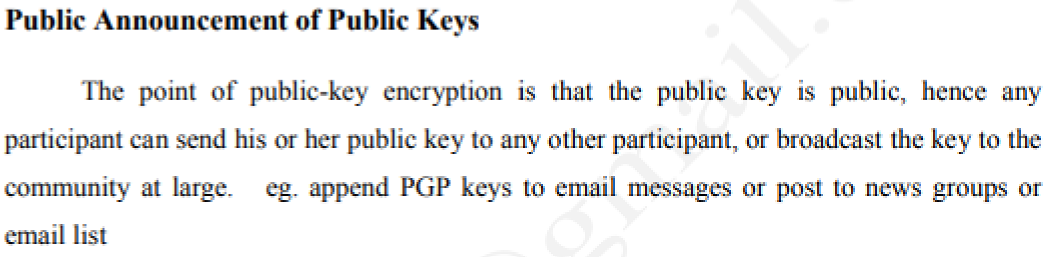
A Hybrid Scheme:

Yet another way to use public-key encryption to distribute secret keys is a hybrid approach.

- This scheme retains the use of a key distribution center (KDC) that shares a secret master key with each user and distributes secret session keys encrypted with the master key.
- A public key scheme is used to distribute the master keys.
- The addition of a public-key layer provides a secure, efficient means of distributing master keys.

DISTRIBUTION OF PUBLIC KEYS

- Several techniques have been proposed for the distribution of public keys , which can mostly be grouped into the categories:
    - Public announcement
    - Publicly available directory
    - Public key authority
    - Public key certificates

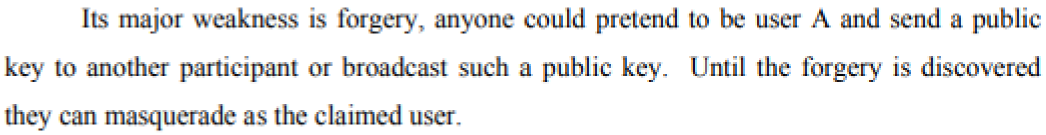


Public Announcement of Public Keys

The point of public-key encryption is that the public key is public, hence any participant can send his or her public key to any other participant, or broadcast the key to the community at large. eg. append PGP keys to email messages or post to news groups or email list

A
PU_a
PU_a
PU_a
PU_a
B
PU_b
PU_b
PU_b
PU_b

Figure 10.1 Uncontrolled Public Key Distribution



Its major weakness is forgery, anyone could pretend to be user A and send a public key to another participant or broadcast such a public key. Until the forgery is discovered they can masquerade as the claimed user.

Publicly Available Directory

- Can obtain greater security by registering keys with a public directory
- Directory must be trusted with properties
  - The authority maintains a directory with a { name, public key} entry for each participant
  - Each participant registers a public key with the directory authority
  - A participant may replace the existing key with a new one at any time because the corresponding private key has been compromised in some way
  - Participant could also access the directory electronically. For this purpose , secure, authenticated communication from the authority to the participant is mandatory

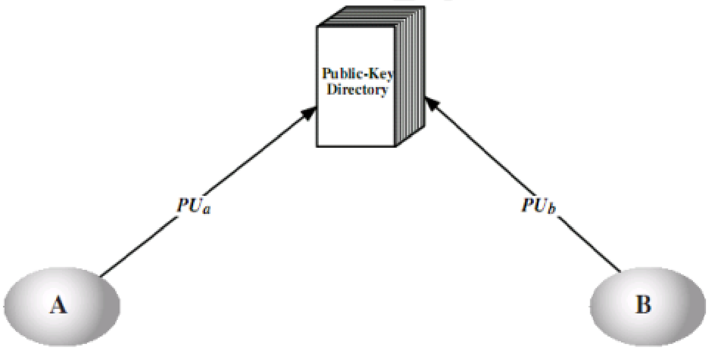
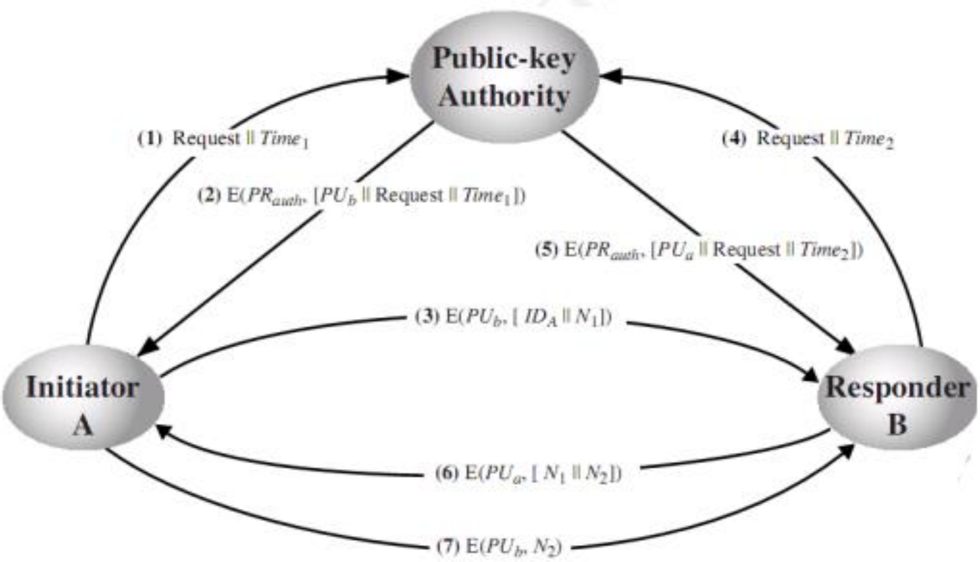


Figure 10.2 Public Key Publication

This scheme is more secure than the individual public announcements, but still has vulnerabilities

Public Key Authority

- Stronger security for public key distribution can be achieved by providing tighter control over the distribution of public keys from the directory
- It requires users to know the public key for the directory, and that they interact with directory in real time to obtain any desired public key securely


Public-key Authority
(1) Request || $Time_1$
(2) E(PR_{auth}, [PU_b || Request || $Time_1$])
(4) Request || $Time_2$
(5) E(PR_{auth}, [PU_a || Request || $Time_2$])
(3) E(PU_b, [ID_A || N_1])
Initiator A
Responder B
(6) E(PU_a, [N_1 || N_2])
(7) E(PU_b, N_2)

1. A sends a timestamped message to the public-key authority containing a request for the current public key of B.

2. The authority responds with a message that is encrypted using the authority's private key, PR_{auth} Thus, A is able to decrypt the message using the authority's public key. Therefore, A is assured that the message originated with the authority. The message includes the following:

    - B's public key, PU_b which A can use to encrypt messages destined for B

    - The original request, to enable A to match this response with the corresponding earlier request and to verify that the original request was not altered before reception by the authority.
    - The original timestamp, so A can determine that this is not an old message from the authority containing a key other than B's current public key.

3. A stores B's public key and also uses it to encrypt a message to B containing an identifier of A (ID_A) and a nonce (N_1), which is used to identify this transaction uniquely.

4. B retrieves A's public key from the authority in the same manner as A retrieved B's public key.
5. At this point, public keys have been securely delivered to A and B, and they may begin their protected exchange. However, two additional steps are desirable:

6. B sends a message to A encrypted with PU_a and containing A's nonce (N_1) as well as a new nonce generated by B (N_2) Because only B could have decrypted message (3), the presence of N_1 in message (6) assures A that the correspondent is B.

7. A returns N_2, encrypted using B's public key, to assure B that its correspondent is A.

Public-Key Certificates

- A user must appeal to the authority for a public key for every other user that it wishes to contact and it is vulnerable to tampering too.
- Public key certificates can be used to exchange keys without contacting a public-key authority.
- A certificate binds an **identity** to **public key**, with all contents **signed** by a trusted Public-Key or Certificate Authority (CA).
- This can be verified by anyone who knows the public-key authorities public-key.

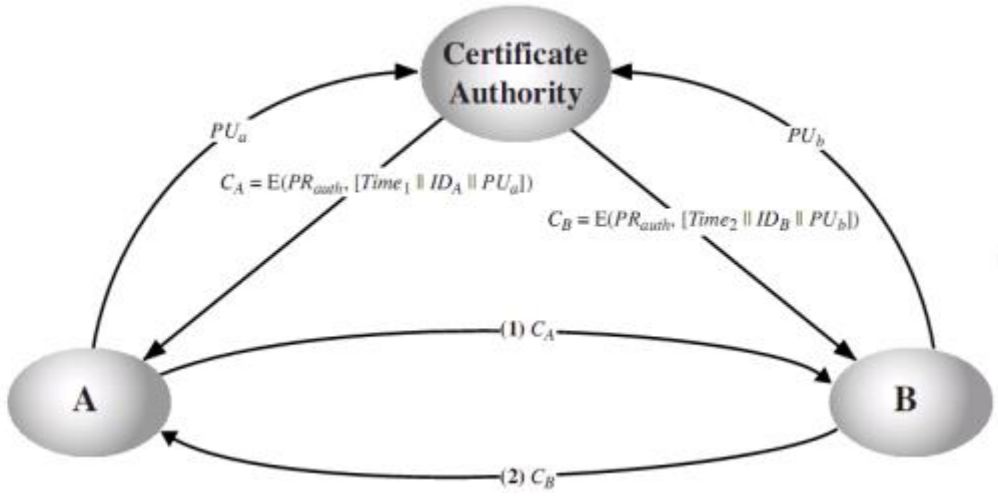
A participant can also convey its key information to another by transmitting its certificate.

Other participants can verify that the certificate was created by the authority. We can place the following requirements on this scheme:

1. Any participant can read a certificate to determine the name and public key of the certificate's owner.
2. Any participant can verify that the certificate originated from the certificate authority and is not counterfeit.
3. Only the certificate authority can create and update certificates.
4. Any participant can verify the currency of the certificate.

One scheme has become universally accepted for formatting public-key certificates: the X.509 standard.

X.509 certificates are used in most network security applications, including IP security, secure sockets layer (SSL), secure electronic transactions (SET), and S/MIME.


Certificate Authority
PU_a
PU_b
$C_A = E(PR_{auth}, [Time_1 \| ID_A \| PU_a])$
$C_B = E(PR_{auth}, [Time_2 \| ID_B \| PU_b])$
(1) C_A
(2) C_B
A
B

SYSTEM SECURITY

- The security of a computer system is a crucial task.
- It is a process of ensuring the confidentiality and integrity of the OS.
- Security is one of most important as well as the major task in order to keep all the threats or other malicious tasks or attacks or program away from the computer's software system.
- A system is said to be secure if its resources are used and accessed as intended under all the circumstances, but no system can guarantee absolute security from several of various malicious threats and unauthorized access.
- The security of a system can be threatened via two violations:
  - **Threat:** A program that has the potential to cause serious damage to the system.
  - **Attack:** An attempt to break security and make unauthorized use of an asset.
- Security violations affecting the system can be categorized as malicious and accidental threats.
- **Malicious threats**, as the name suggests are a kind of harmful computer code or web script designed to create system vulnerabilities leading to back doors and security breaches.

- Security can be compromised via any of the breaches mentioned:
    - **Breach of confidentiality:** This type of violation involves the unauthorized reading of data.
    - **Breach of integrity:** This violation involves unauthorized modification of data.
    - **Breach of availability:** It involves unauthorized destruction of data.
    - **Theft of service:** It involves the unauthorized use of resources.
    - **Denial of service:** It involves preventing legitimate use of the system. As mentioned before, such attacks can be accidental in nature.

Security System Goal:

1. **Integrity:**

    The objects in the system mustn't be accessed by any unauthorized user & any user not having sufficient rights should not be allowed to modify the important system files and resources.

2. **Secrecy:**

    The objects of the system must be accessible only to a limited number of authorized users. Not everyone should be able to view the system files.

3. **Availability:**

    All the resources of the system must be accessible to all the authorized users i.e. only one user/process should not have the right to hog all the system resources.

    If such kind of situation occurs, denial of service could happen.

    In this kind of situation, malware might hog the resources for itself & thus preventing the legitimate processes from accessing the system resources.

Threats can be classified into the following two categories:

1. **ProgramThreats:**

    - A program was written by a cracker to hijack the security or to change the behavior of a normal process.
    - In other words, if a user program is altered and further made to perform some malicious unwanted tasks, then it is known as Program Threats.

2. **SystemThreats:**

    - These threats involve the abuse of system services.
    - They strive to create a situation in which operating-system resources and user files are misused.
    - They are also used as a medium to launch program threats.

Types of Program Threats:

Virus:

- It is a self-replicating and malicious thread that attaches itself to a system file and then rapidly replicates itself, modifying and destroying essential files leading to a system breakdown.

TrojanHorse:

- A code segment that misuses its environment is called a Trojan Horse.
- They seem to be attractive and harmless cover programs but are really harmful hidden programs that can be used as the virus carrier.

TrapDoor:

- The designer of a program or system might leave a hole in the software that only he is capable of using, the Trap Door works on similar principles.
- Trap Doors are quite difficult to detect as to analyze them, one needs to go through the source code of all the components of the system.

LogicBomb:

- A program that initiates a security attack only under a specific situation.
- To be very precise, a logic bomb is actually the most malicious program which is inserted intentionally into the computer system and that is triggered or functions when specific conditions have been met for it to work.

Worm:

- A computer worm is a type of malware that replicates itself and infects other computers while remaining active on affected systems.
- A computer worm replicates itself

Types of System Threats –

Worm:

- An infection program that spreads through networks.
- Unlike a virus, they target mainly LANs.
- A computer affected by a worm attacks the target system and writes a small program “hook” on it. This hook is further used to copy the worm to the target computer. This process repeats recursively, and soon enough all the systems of the LAN are affected.
- It uses the spawn mechanism to duplicate itself.
- The worm spawns copies of itself, using up a majority of system resources and also locking out all other processes.

PortScanning:

- The cracker identifies the vulnerabilities of the system to attack.
- It is an automated process that involves creating a TCP/IP connection to a specific port

DenialofService:

- Not aimed for the purpose of collecting information or destroying system files.
- Rather, they are used for disrupting the legitimate use of a system or facility.
- These attacks are generally network-based.

INTRUDERS

- Intruders are often referred to as hackers and are the most harmful factors contributing to security vulnerability.
- They have immense knowledge and an in-depth understanding of technology and security.
- Intruders breach the privacy of users and aim to steal the confidential information of the users.
- The stolen information is then sold to third parties, aiming to misuse it for personal or professional gains.

Types of Intruders

- **Masquerader:**
  - The category of individuals that are not authorized to use the system but still exploit users' privacy and confidential information by possessing techniques that give them control over the system, such category of intruders is referred to as Masquerader.
  - Masqueraders are outsiders and hence they don't have direct access to the system, they aim to attack unethically to steal data.

- **Misfeasor:**
  - The category of individuals that are authorized to use the system, but misuse the granted access and privilege.
  - These are individuals that take undue advantage of the permissions and access given to them, such category of intruders is referred to as Misfeasor.
  - Misfeasors are insiders and they have direct access to the system, which they aim to attack unethically for stealing data/ information.

- **Clandestine User:**
  - The category of individuals who have supervision/administrative control over the system and misuse the authoritative power given to them.
  - The misconduct of power is often done by superlative authorities for financial gains, such a category of intruders is referred to as Clandestine Users.
  - A Clandestine User can be any of the two, insiders or outsiders, and accordingly, they can have direct/ indirect access to the system, which they aim to attack unethically by stealing data/ information.

How to Protect From Intruders?

- By being aware of all the security measures that help us to protect ourselves from Intruders.
- By increasing the security and strengthening the security of the system.

INTRUSION DETECTION SYSTEM (IDS)

- An Intrusion Detection System (IDS) is a security tool that monitors a computer network or systems for malicious activities or policy violations.
- It helps detect unauthorized access, potential threats, and abnormal activities by analyzing traffic and alerting administrators to take action.
- An IDS is crucial for maintaining network security and protecting sensitive data from cyber-attacks.
- An Intrusion Detection System (IDS) maintains network traffic looks for unusual activity and sends alerts when it occurs.
- The main duties of an Intrusion Detection System (IDS) are anomaly detection and reporting, however, certain Intrusion Detection Systems can take action when malicious activity or unusual traffic is discovered.
- It is software that checks a network or system for malicious activities or policy violations.
- Each illegal activity or violation is often recorded either centrally using an SIEM system or notified to an administration.
- IDS monitors a network or system for malicious activity and protects a computer network from unauthorized access from users, including perhaps insiders.

- The intrusion detector learning task is to build a predictive model (i.e. a classifier) capable of distinguishing between ‘bad connections’ (intrusion/attacks) and ‘good (normal) connections’.

Working of Intrusion Detection System(IDS)

- An IDS (Intrusion Detection System) monitors the traffic on a computer network to detect any suspicious activity.
- It analyzes the data flowing through the network to look for patterns and signs of abnormal behavior.
- The IDS compares the network activity to a set of predefined rules and patterns to identify any activity that might indicate an attack or intrusion.
- If the IDS detects something that matches one of these rules or patterns, it sends an alert to the system administrator.
- The system administrator can then investigate the alert and take action to prevent any damage or further intrusion.

Classification of Intrusion Detection System(IDS)

Intrusion Detection System are **classified into 5 types**:

- **Network Intrusion Detection System (NIDS):**
  - Network intrusion detection systems (NIDS) are set up at a planned point within the network to examine traffic from all devices on the network.
  - It performs an observation of passing traffic on the entire subnet and matches the traffic that is passed on the subnets to the collection of known attacks.
  - Once an attack is identified or abnormal behavior is observed, the alert can be sent to the administrator.
  - An example of a NIDS is installing it on the subnet where firewalls are located in order to see if someone is trying to crack the firewall.
- **Host Intrusion Detection System (HIDS):**
  - Host intrusion detection systems (HIDS) run on independent hosts or devices on the network.
  - A HIDS monitors the incoming and outgoing packets from the device only and will alert the administrator if suspicious or malicious activity is detected.
  - It takes a snapshot of existing system files and compares it with the previous snapshot.
  - If the analytical system files were edited or deleted, an alert is sent to the administrator to investigate.
  - An example of HIDS usage can be seen on mission-critical machines, which are not expected to change their layout.
- **Protocol-Based Intrusion Detection System (PIDS):**
  - Protocol-based intrusion detection system (PIDS) comprises a system or agent that would consistently reside at the front end of a server, controlling and interpreting the protocol between a user/device and the server.
  - It is trying to secure the web server by regularly monitoring the HTTPS protocol stream and accepting the related HTTP protocol.
  - As HTTPS is unencrypted and before instantly entering its web presentation layer then this system would need to reside in this interface, between to use the HTTPS.

- **Application Protocol-Based Intrusion Detection System (APIDS):**

    - An application Protocol-based Intrusion Detection System (APIDS) is a system or agent that generally resides within a group of servers.
    - It identifies the intrusions by monitoring and interpreting the communication on application-specific protocols.
    - For example, this would monitor the SQL protocol explicitly to the middleware as it transacts with the database in the web server.

- **Hybrid Intrusion Detection System:**

    - Hybrid intrusion detection system is made by the combination of two or more approaches to the intrusion detection system.
    - In the hybrid intrusion detection system, the host agent or system data is combined with network information to develop a complete view of the network system.
    - The hybrid intrusion detection system is more effective in comparison to the other intrusion detection system. Prelude is an example of Hybrid IDS.

Benefits of IDS

- **Detects Malicious Activity:** IDS can detect any suspicious activities and alert the system administrator before any significant damage is done.

- **Improves Network Performance:** IDS can identify any performance issues on the network, which can be addressed to improve network performance.

- **Compliance Requirements:** IDS can help in meeting compliance requirements by monitoring network activity and generating reports.

- **Provides Insights:** IDS generates valuable insights into network traffic, which can be used to identify any weaknesses and improve network security.

Detection Method of IDS

- **Signature-Based Method:**

    - Signature-based IDS detects the attacks on the basis of the specific patterns such as the number of bytes or a number of 1s or the number of 0s in the network traffic.
    - It also detects on the basis of the already known malicious instruction sequence that is used by the malware.
    - The detected patterns in the IDS are known as signatures. Signature-based IDS can easily detect the attacks whose pattern (signature) already exists in the system but it is quite difficult to detect new malware attacks as their pattern (signature) is not known.

- **Anomaly-Based Method:**

    - Anomaly-based IDS was introduced to detect unknown malware attacks as new malware is developed rapidly.
    - In anomaly-based IDS there is the use of machine learning to create a trustful activity model and anything coming is compared with that model and it is declared suspicious if it is not found in the model.
    - The machine learning-based method has a better-generalized property in comparison to signature-based IDS as these models can be trained according to the applications and hardware configurations.

Advantages

- **Early Threat Detection**: IDS identifies potential threats early, allowing for quicker response to prevent damage.
- **Enhanced Security**: It adds an extra layer of security, complementing other cybersecurity measures to provide comprehensive protection.
- **Network Monitoring**: Continuously monitors network traffic for unusual activities, ensuring constant vigilance.
- **Detailed Alerts**: Provides detailed alerts and logs about suspicious activities, helping IT teams investigate and respond effectively.

Disadvantages

- **False Alarms**: IDS can generate false positives, alerting on harmless activities and causing unnecessary concern.
- **Resource Intensive**: It can use a lot of system resources, potentially slowing down network performance.
- **Requires Maintenance**: Regular updates and tuning are needed to keep the IDS effective, which can be time-consuming.
- **Doesn't Prevent Attacks**: IDS detects and alerts but doesn't stop attacks, so additional measures are still needed.
- **Complex to Manage**: Setting up and managing an IDS can be complex and may require specialized knowledge.

Comparison of IDS with Firewalls

- IDS and firewall both are related to network security but an IDS differs from a firewall as a firewall looks outwardly for intrusions in order to stop them from happening.
- Firewalls restrict access between networks to prevent intrusion and if an attack is from inside the network it doesn't signal. An IDS describes a suspected intrusion once it has happened and then signals an alarm.

PASSWORD MANAGEMENT IN CYBER SECURITY

- A Password is defined as a system that facilitates an easy and secure way to store passwords and access them quickly when needed.
- Password management is an integral part of most organizations' IT infrastructure today.
- The password management solution ensures improved cybersecurity and convenience for users in homes and offices.

What is Password Management?

- Password management is a system that facilitates an easy and secure way to store passwords and quickly access them when needed.
- One solution to this modern problem is password management.
- With a password manager, users can manage all of their passwords personal and business from one central location.
- A password manager does more than just remember your passwords.
- It helps you choose strong enough passwords, ensures timely password changes, and enforces many computer security best practices.

Issues Related to Managing Passwords

- The main problem with password management is that it is not safe to use the same password for multiple sites, therefore having different passwords for different sites and on top of that remembering them is quite difficult.

- To escape from this situation people often tend to use password managers (A password manager is a computer program that allows users to store, generate, and manage their passwords for local applications and online services.).

- Password managers to a certain extent reduce the problem by having to remember only one “master password” instead of having to remember multiple passwords.

- The only problem with having a master password is that once it is out or known to an attacker, the rest of all the passwords become available.

The main issues related to managing passwords are as follows:

- Login spoofing
- Sniffing attack
- Brute force attack
- Shoulder surfing attack
- Data breach

Methods to Manage Password

- **Strong and long passwords:**

  A minimum length of 8 to 12 characters long, also it should contain at least three different character sets (e.g., uppercase characters, lowercase characters, numbers, or symbols).

- **Password Encryption:**

  Using irreversible end-to-end encryption is recommended. In this way, the password remains safe even if it ends up in the hands of cybercriminals.

- **Multi-factor Authentication (MFA):**

  Adding MFA layer as some security questions and a phone number that would be used to confirm that it is indeed you who is trying to log in will enhance the security of your password.

- **Make the password pass the test:**

  Yes, put your password through some testing tools that you might find online in order to ensure that it falls under the strong and safe password category.

- **Avoid updating passwords frequently:**

  Though it is advised or even made mandatory to update or change your password. as frequently as in 60 or 90 days.

MALICIOUS SOFTWARES

- M**alware** is malicious software and refers to any software that is designed to cause harm to computer systems, networks, or users.

- Malware can take many forms.

- Malware is software that gets into the system without user consent to steal the user’s private and confidential data, including bank details and passwords.

- They also generate annoying pop-up ads and make changes in system settings

- They get into the system through various means:
    - Along with free downloads.
    - Clicking on a suspicious link.
    - Opening emails from malicious sources.
    - Visiting malicious websites.
    - Not installing an updated version of antivirus in the system.

Types of Malware

1. Virus

- Computer virus refers to a program which damages computer systems and/or destroys or erases data files.
- A computer virus is a malicious program that self-replicates by copying itself to another program
- The purpose of creating a computer virus is to infect vulnerable systems, gain admin control and steal user sensitive data.

2. Worm

- A worm is a destructive program that fills a computer system with self-replicating information, clogging the system so that its operations are slowed down or stopped.

3. Logical Bomb

- A logical bomb is a destructive program that performs an activity when a certain action has occurred.
- These are hidden in programming code. Executes only when a specific condition is met

4. Trojan / Backdoor

- Trojan Horse is a destructive program.
- It usually pretends as computer games or application software.
- If executed, the computer system will be damaged.
- Trojan Horse usually comes with monitoring tools and key loggers.
- These are active only when specific events are alive.
- These are hidden with packers, crypters and wrappers

5. RootKits

- Rootkit is a collection of tools that allow an attacker to take control of a system.
- Can be used to hide evidence of an attacker's presence and give them backdoor access.
- Can contain log cleaners to remove traces of attacker.

How To Protect From Malware?

- Update your operating system and software.
- Never click on a popup's link.
- Don't install too many apps on your devices.
- Be cautious when using the internet.
- Do not click on unidentified links.
- Choose the websites you visit wisely.
- Emails requesting personal information should be avoided.

How To Remove Malware?

- A large number of security software programs are made to both find and stop malware as well as to eliminate it from infected systems.
- An anti-malware tool that handles malware detection and removal is Malware bytes.
- Malware can then be quarantined and removed if it is found.

Advantages of Detecting and Removing Malware

- **Improved Security:** By detecting and removing malware, individuals, and organizations can improve the security of their systems and reduce the risk of future infections.
- **Prevent Data Loss:** Malware can cause data loss, and by removing it, individuals and organizations can protect their important files and information.
- **Protect Reputation:** Malware can cause harm to a company's reputation, and by detecting and removing it, individuals and organizations can protect their image and brand.
- **Increased Productivity:** Malware can slow down systems and make them less efficient, and by removing it, individuals and organizations can increase the productivity of their systems and employees.

Disadvantages of Detecting and Removing Malware

- **Time-Consuming:** The process of detecting and removing malware can be time-consuming and require specialized tools and expertise.
- **Cost:** Antivirus software and other tools required to detect and remove malware can be expensive for individuals and organizations.
- **False Positives:** Malware detection and removal tools can sometimes result in false positives, causing unnecessary alarm and inconvenience.
- **Difficulty:** Malware is constantly evolving, and the process of detecting and removing it can be challenging and require specialized knowledge and expertise.
- **Risk of Data Loss:** Some malware removal tools can cause unintended harm, resulting in data loss or system instability.

VIRUSES & RELATED THREATS

- A virus is a fragment of code embedded in a legitimate program.
- Viruses are self-replicating and are designed to infect other programs.

What Does a Computer Virus Do?

- A virus can **harm** or **destroy data**, **slow down system resources**, and **log keystrokes**, among other things.
- A virus can have **unexpected** or **harmful outcomes** during this procedure, such as **destroying system software** by corrupting data.
- Some viruses are made to mess things up by **deleting files**, **messing up programs**, or even **wiping out your hard drive** completely.
- Even if they're not super harmful, viruses can still **slow down your computer** a lot, **using up memory and making it crash often**.

How To Prevent Your Computer From Viruses?

- Install Antivirus Software:
- Update Regularly
- Be Cautious with Emails and Downloads
- Use Strong Passwords
- Backup Your Data

How To Remove Computer Viruses?

To remove a computer infection, you can choose from two options:

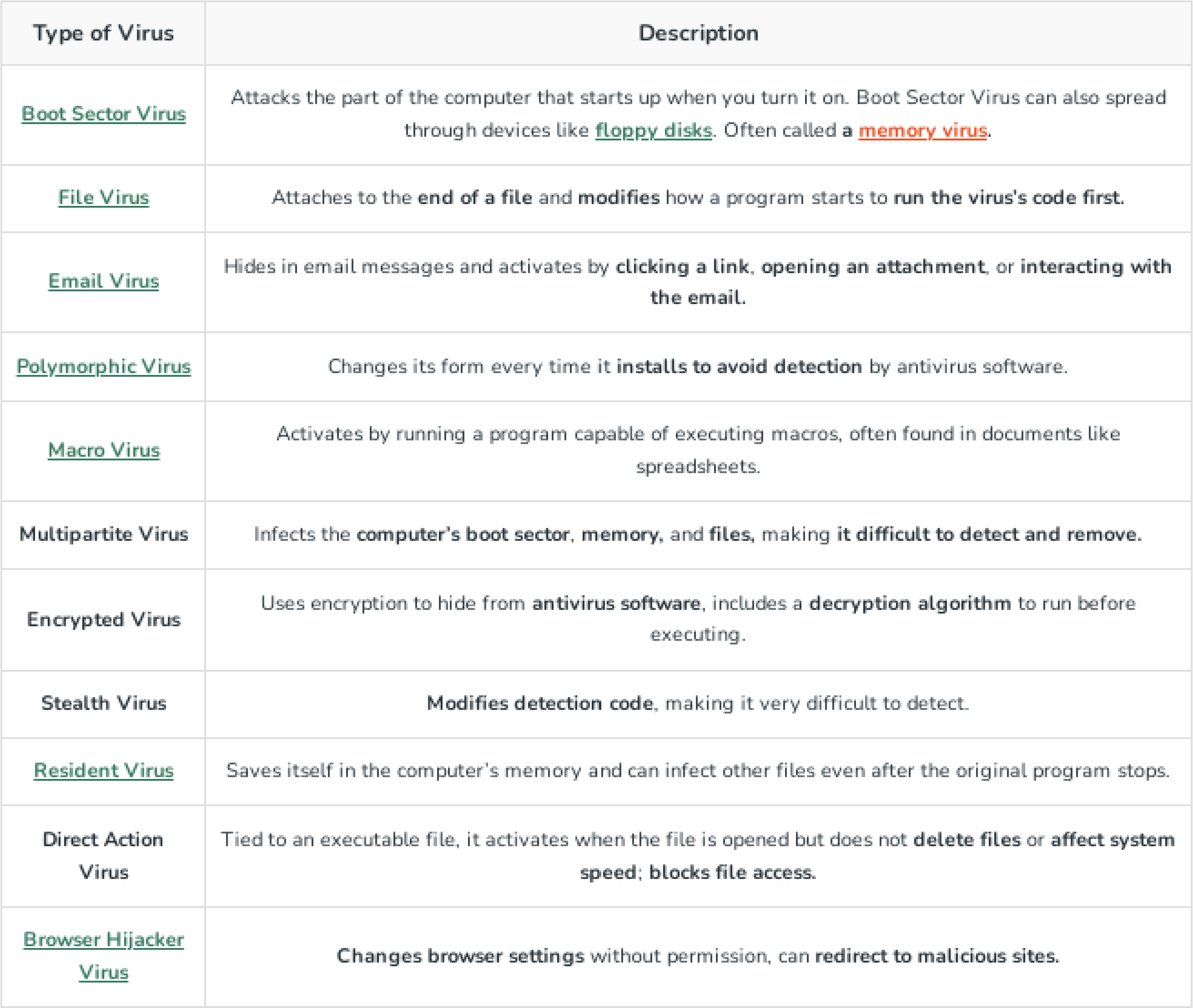
- **Do-it-yourself manual approach:**

  This means you try to fix the problem on your own. Usually, you start by **searching online** for solutions. Then, you might have to do a lot of tasks to **clean up your computer**
  It can take time and might need some experience to finish everything.

- **Get help from a reliable antivirus product**

  Another option is to use **antivirus software**.
  This software is designed to find and remove viruses from your computer.

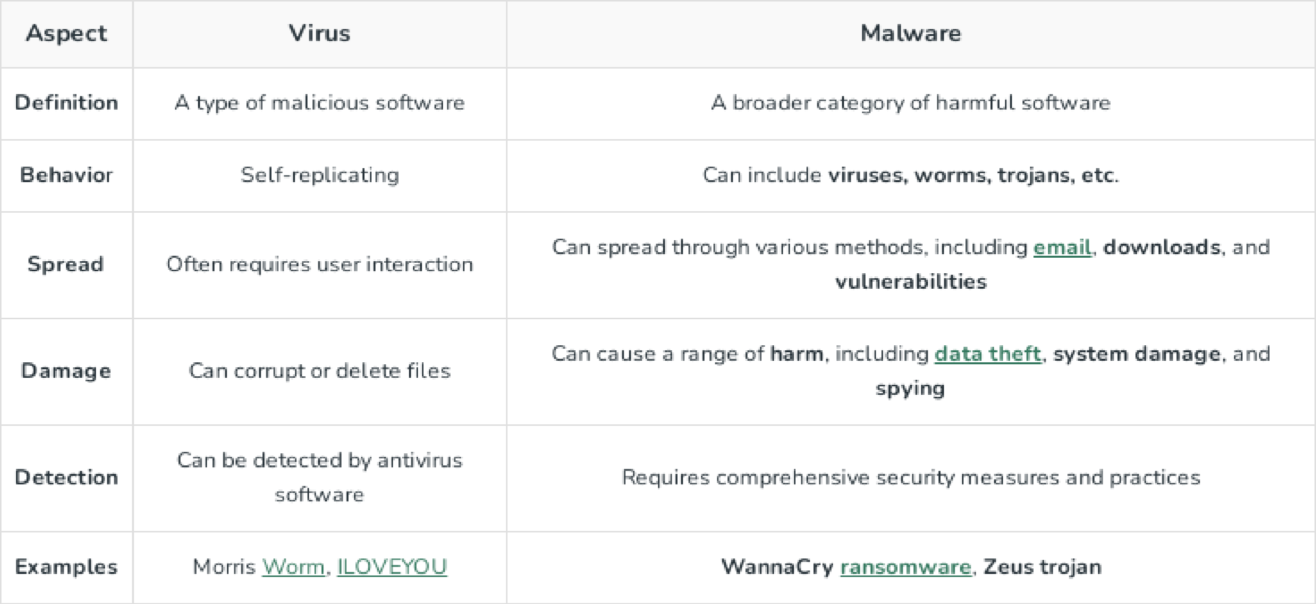
TYPES OF VIRUSES



Type of Virus	Description
Boot Sector Virus	Attacks the part of the computer that starts up when you turn it on. Boot Sector Virus can also spread through devices like floppy disks. Often called **a** memory virus.
File Virus	Attaches to the **end of a file** and **modifies** how a program starts to **run the virus's code first.**
Email Virus	Hides in email messages and activates by **clicking a link**, **opening an attachment**, or **interacting with the email.**
Polymorphic Virus	Changes its form every time it **installs to avoid detection** by antivirus software.
Macro Virus	Activates by running a program capable of executing macros, often found in documents like spreadsheets.
Multipartite Virus	Infects the **computer's boot sector**, **memory**, and **files**, making **it difficult to detect and remove.**
Encrypted Virus	Uses encryption to hide from **antivirus software**, includes a **decryption algorithm** to run before executing.
Stealth Virus	**Modifies detection code**, making it very difficult to detect.
Resident Virus	Saves itself in the computer's memory and can infect other files even after the original program stops.
Direct Action Virus	Tied to an executable file, it activates when the file is opened but does not **delete files** or **affect system speed**; **blocks file access.**
Browser Hijacker Virus	**Changes browser settings** without permission, can **redirect to malicious sites.**

.

Virus V/S Malware – What is the difference?



Aspect	Virus	Malware
Definition	A type of malicious software	A broader category of harmful software
Behavior	Self-replicating	Can include **viruses, worms, trojans, etc.**
Spread	Often requires user interaction	Can spread through various methods, including email, **downloads**, and **vulnerabilities**
Damage	Can corrupt or delete files	Can cause a range of **harm**, including data theft, **system damage**, and **spying**
Detection	Can be detected by antivirus software	Requires comprehensive security measures and practices
Examples	Morris Worm, ILOVEYOU	**WannaCry** ransomware, **Zeus trojan**

VIRUS – COUNTER MEASURES

- To protect against viruses, antivirus software should be installed.
- Those who are using antivirus software must perform scan using the latest virus-scanning engine and virus definition files.
- The control measures against viruses are:
    - Install antivirus or anti-malware software.
    - Keep your antivirus software up to date.
    - Run antivirus scans regularly.
    - Keep your operating system up to date.
    - Protect your network.
    - Think before you click.
    - Keep your personal information secure.
    - Don't use unsecured Wi-Fi.
    - Use several secured passwords

DISTRIBUTED DENIAL OF SERVICE (DDoS) ATTACKS

- Distributed Denial of Service (DDoS) is a type of DOS attack where multiple systems, which are trojan infected, target a particular system which causes a DoS attack.
- A DDoS attack uses multiple servers and Internet connections to flood the targeted resource.
- A DDoS attack is one of the most powerful weapons on the cyber platform

Types of DDoS Attacks

1. **Volumetric Attacks:**

Volumetric attacks focus on consuming the network bandwidth and saturating it by amplification or botnet to hinder its availability to the users. They are easy to generate by directing a massive amount of traffic to the target server.

Examples: NTP Amplification, DNS Amplification, UDP Flood attack, and TCP Flood attack.

2. **Protocol Attacks:**

    They are also known as state-exhaustion attacks. These attacks focus on vulnerabilities in layer 3 and layer 4 of the protocol stack. These types of attacks consume resources like servers, firewalls, and load balancers.

    Examples: SYN Flood attack and Ping of Death.

3. **Application Attacks:**

    These attacks focus on attacking layer 7 of the OSI model where the web pages are generated in response to the request initiated by the end-user. For a client, generating a request does not take any heavy load and it can easily generate multiple requests to the server. On the other hand, responding to a request takes a considerable load for the server as it has to build all the pages, compute any queries, and load the results from the database according to the request.

4. **Fragmentation Attacks:**

    The cybercriminal exploits frangibility in the datagram fragmentation process, in which IP datagrams are divided into smaller packets, transferred across a network, and then reassembled. In such attacks, fake data packets are unable to be reassembled.

How to Protect from DDoS Attacks?

1. Take quick action
2. Configure firewalls and
3. Consider artificial intelligence
4. Secure your Internet of Things devices