BCSE353E
Information Security Analysis and Audit

By,
Dr. Kovendan AKP,
Senior Assistant Professor,
Department of Database Systems,
School of Computer Science and Engineering,
Vellore Institute of Technology, Vellore.

Email: kovendan.akp@vit.ac.in   Mobile: +91-9677190102   Cabin: PRP 208-C

Guidelines to be followed

1. Be on time for class.
2. Be attentive and clarify your doubts immediately.
3. Don’t indulge in other actives when the class is in progress.
4. Complete the Assignments/Quiz within the deadline provided.
5. Maintain a dedicated notebook in class.

Course Objectives

1. To introduce system security related incidents and insight on potential defenses, counter measures against common threat/vulnerabilities.
2. To provide the knowledge of installation, configuration and troubleshooting of information security devices.
3. To make students familiarize on the tools and common processes in information security audits and analysis of compromised systems.

Course Outcomes

After successfully completing the course the student should be able to

CO1: Contribute to managing information security

CO2: Co-ordinate responses to information security incidents

CO3: Contribute to information security audits

CO4: Support teams to prepare for and undergo information security audits

CO5: Maintain a healthy, safe and secure working environment

CO6: Provide data/information in standard formats

CO7: Develop knowledge, skills and competence in information security

Modules

Module 1: Information Security Fundamentals

Module 2: System Security

Module 3: Information Security Management

Module 4: Incident Management

Module 5: Incident Response

Module 6: Conducting Security Audits

Module 7: Information Security Audit Preparation

Module 8: Self and Work Management

Text Books

1. **William Stallings, Lawrie Brown, Computer Security: Principles and Practice, 3rd edition, 2014.**
2. Nina Godbole, Information Systems Security: Security Management, Metrics, Frameworks and Best Practices, Wiley, 2017
3. Nina Godbole, Sunit Belapure, Cyber Security- Understanding cyber-crimes, computer forensics and legal perspectives, Wiley Publications, 2016
4. Andrew Vladimirov Michajlowski, Konstantin, Andrew A. Vladimirov, Konstantin V. Gavrilenko, Assessing Information Security: Strategies, Tactics, Logic and Framework, IT Governance Ltd, O’Reilly, 2010

Internal Assessment Pattern

Internal Component	Total Mark	Mark Consolidation	Mode of Conduct
Digital Assignment	10	10	VTOP
Quiz 1	20	10	Google Forms
Quiz 2	20	10	Google Forms
CAT 1	50	15	Offline Mode by COE
CAT 2	50	15	Offline Mode by COE
	Total Internal Marks	60	

WhatsApp Group

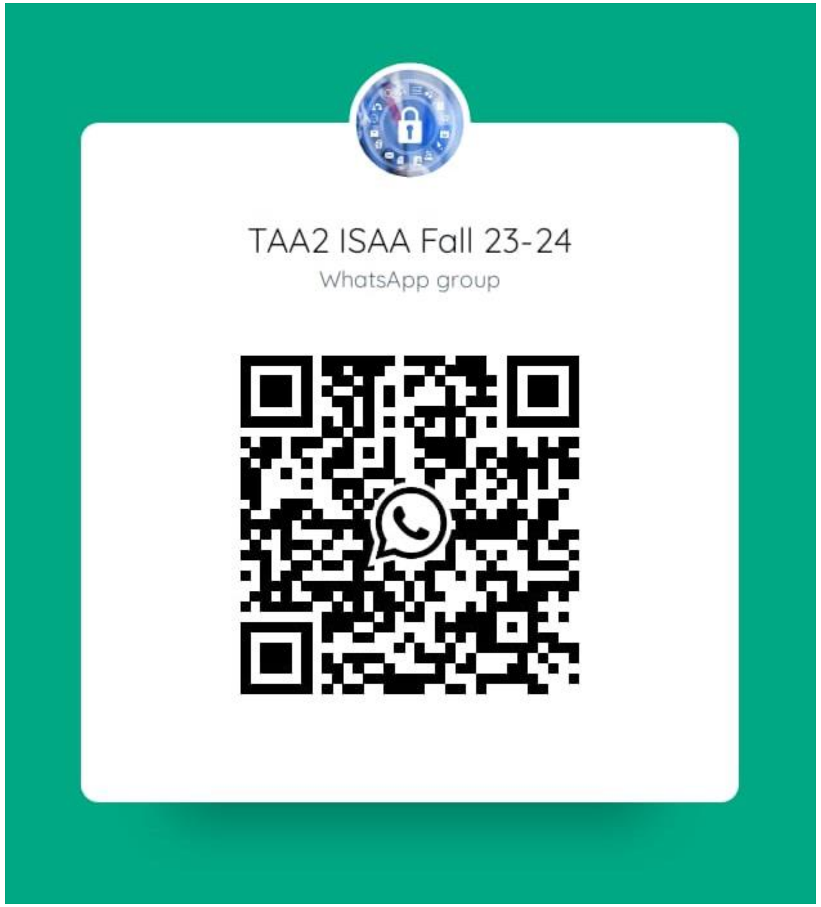
- TAA1 – Slot
- Join the Whatsapp group using the QR Code/link
- https://chat.whatsapp.com/FV2wUhNEfVxDkYXvdBuTNo

TAA1 - ISAA

WhatsApp group

WhatsApp Group

- TAA2 – Slot
- Join the Whatsapp group using the QR Code/Link
- https://chat.whatsapp.com/Dd8stpbWJdVBGcud6rV2NJ



Module 1

Definitions & challenges of security, Attacks & services, Access control structures, Firewalls.

Computer Security

- **National Institute of Standards and Technology**
- NIST Definition of Computer Security
  - The **protection** afforded to an **automated information system** in order to attain the **applicable objectives of preserving the integrity, availability, and confidentiality** of information system resources (includes hardware, software, firmware, information/data, and telecommunications).
- Three Key Objectives
  - **Confidentiality**
  - **Integrity**
  - **Availability**

CIA Triad

Confidentiality

- Preserving **authorized restrictions on information access and disclosure**, including means for protecting personal privacy and proprietary information.
- **A loss of confidentiality is the unauthorized disclosure of information.**
- Two Concepts
    - **Data Confidentiality**
        - Assures that **private or confidential information is not made available** or disclosed to **unauthorized individuals**.
    - **Privacy**
        - Assures that **individuals control** or influence what **information related to them may be collected and stored and by whom and to whom that information may be disclosed**.

Integrity

- **Guarding against improper information modification or destruction**, including ensuring information **nonrepudiation and authenticity**.
  - Non-repudiation is the assurance that someone cannot deny the validity of something.
    - Eg: Signing a document
  - Authenticity is the quality of being true.
- **A loss of integrity is the unauthorized modification or destruction of information.**
- Two Concepts
  - **Data Integrity**
    - Assures that **information and programs are changed only in a specified and authorized manner**.
  - **System Integrity**
    - Assures that a **system performs its intended function in an unimpaired manner**, free from deliberate or inadvertent unauthorized manipulation of the system.

Availability

- Assures that **systems work promptly and service is not denied to authorized users**.
- Ensuring **timely and reliable access to and use of information**.
- **A loss of availability is the disruption of access to or use of information or an information system.**

Additional Objectives of Security

- **Authenticity**
  - The property of being **genuine and being able to be verified and trusted**;
  - **Confidence in the validity of a transmission, a message, or message originator**.
  - According to FIPS 199 includes authenticity under integrity. [Federal Information Processing Standards]
- **Accountability**
  - The security goal that generates the requirement for **actions of an entity to be traced uniquely to that entity**.
  - This supports nonrepudiation, deterrence, fault isolation, intrusion detection and prevention, and after-action recovery and legal action.

Examples for CIA Triad

Levels of impact

Low	Medium	High
The loss could be expected to have a limited adverse effect.	The loss could be expected to have a serious adverse effect.	The loss could be expected to have a severe or catastrophic adverse effect.
Cause a degradation in mission capability.	Cause a significant degradation in mission capability	Cause a severe degradation in or loss of mission capability
Result in minor damage to organizational assets	Result in significant damage to organizational assets	Result in major damage to organizational assets
Result in minor financial loss	Result in significant financial loss	Result in major financial loss
Result in minor harm to individuals.	Result in significant harm to individuals that does not involve loss of life or serious, life-threatening injuries.	Result in severe or catastrophic harm to individuals involving loss of life or serious life-threatening injuries.

Examples for Confidentiality

- **Student Grade Information**
  - Only Students, Parents and staff must access the information.
  - High Confidentiality is required.
- **Student Enrolment**
  - Results in less damage if disclosed.
  - Moderate Confidentiality
- **List of Faculty, List of departments in Web Site**
  - Low Confidentiality
  - This information has to be known by a vast range of people to increase the credits of the university.

Examples for Integrity

- **Patient Allergy information**
  - If a non authorized person changes this information it results in a huge damage to the individual and the hospital.
  - High need of data Integrity
- **Web site that offers a forum for discussion**
  - A hacker can falsify the data put up as a part of the forum
  - When the site is only for entertainment purpose then it only brings in a small loss of data and time.
  - Moderate need of data Integrity
- **Anonymous online poll**
  - Just used as statistics
  - Inaccuracy and unscientific nature of these polls are well known
  - Low need of data Integrity

Examples of Availability

- **Authentication services for critical systems, applications, and devices**
  - Interruption of service results in the inability for customers to access computing resources and staff to access the resources they need to perform critical tasks.
  - Leads to a large financial loss.
  - High level of Availability is required.
- **Public Website for a University**
  - Not a critical part of universities information system.
  - But unavailability causes embarrassment
  - Moderate level of Availability
- **Online telephone directory lookup**
  - There are other ways to access the information.
  - Low level of availability.

Challenges in Computer Security

- **Mechanisms used to meet the CIA requirements can be quite complex**, and understanding them may involve rather subtle reasoning.
- In **developing a particular security mechanism or algorithm**, one must always consider potential attacks on those security features which is complex.
- It is **not obvious from the statement of a particular requirement that such elaborate measures are needed**.
  - It is only when the various aspects of the threat are considered that elaborate security mechanisms make sense.
- Deciding **where to use the security mechanisms is complex**. (Both physical placement and logical sense)
- **Security mechanisms typically involve more than a particular algorithm or protocol.**
  - Participants might be involved with some secret information.
  - So transmission and holding this secret information is questionable.
- It is the **duty of the designer to identify and eliminate all the weakness in the system** which is again complex.
- Security **requires regular, even constant, monitoring**, and this is difficult in today's short-term, overloaded environment.
- **Security is not made as the integral part of the design which** makes it even more challenging.
- Many users and even **security administrators view strong security as an impediment** to efficient and user-friendly operation of an information system.

Computer Security Terminology

Terminology	Description
Adversary	An entity that attacks, or is a threat to, a system.
Attack	An intelligent act that is a deliberate attempt to evade security services and violate the security policy of a system.
Countermeasure	An action, device, procedure, or technique that reduces a threat, a vulnerability, or an attack by eliminating or preventing it, by minimizing the harm it can cause, or by discovering and reporting it so that corrective action can be taken.
Risk	An expectation of loss expressed as the probability that a particular threat will exploit a particular vulnerability with a particular harmful result.
Security Policy	A set of rules and practices that specify or regulate how a system or organization provides security services to protect sensitive and critical system resources.
System Resource (Asset)	Data contained in an information system; or a service provided by a system; or a system capability, such as processing power or communication bandwidth; or an item of system equipment (i.e., a system component— hardware, firmware, software, or documentation); or a facility that houses system operations and equipment.
Threat	A threat is a possible danger that might exploit a vulnerability.
Vulnerability	A flaw or weakness in a system's design, implementation, or operation and management that could be exploited to violate the system's security policy.

Difference between Risk, Threat, Vulnerability

Vulnerability	Threat	Risk
The **weakness** in hardware, software, or designs	Take advantage of vulnerabilities in the system and have the **potential to steal and damage data**.	The **potential for loss or destruction of data** is caused by cyber threats.
Can be controlled.	Generally, can't be controlled.	Can be controlled.
Vulnerability management is a process of identifying the problems, then categorizing them, prioritizing them, and resolving the vulnerabilities in that order.	Can be blocked by managing the vulnerabilities.	Reducing data transfers, downloading files from reliable sources, updating the software regularly, hiring a professional cybersecurity team to monitor data, developing an incident management plan, etc. help to lower down the possibility of cyber risks.
Penetration testing and Vulnerability Scanners	Anti-virus software and threat detection logs.	Identifying mysterious emails, suspicious pop-ups, observing unusual password activities, a slower than normal network

A Model for Computer Security

- **Assets**
  - **Hardware**
    - Including computer systems and other data processing, data storage, and data communications devices.
  - **Software**
    - System Software + Application Software
  - **Data**
    - Including files and databases, as well as security-related data, such as password files.
  - **Communication Facilities and Networks**
    - Local and wide area network communication links, bridges, routers, and so on.

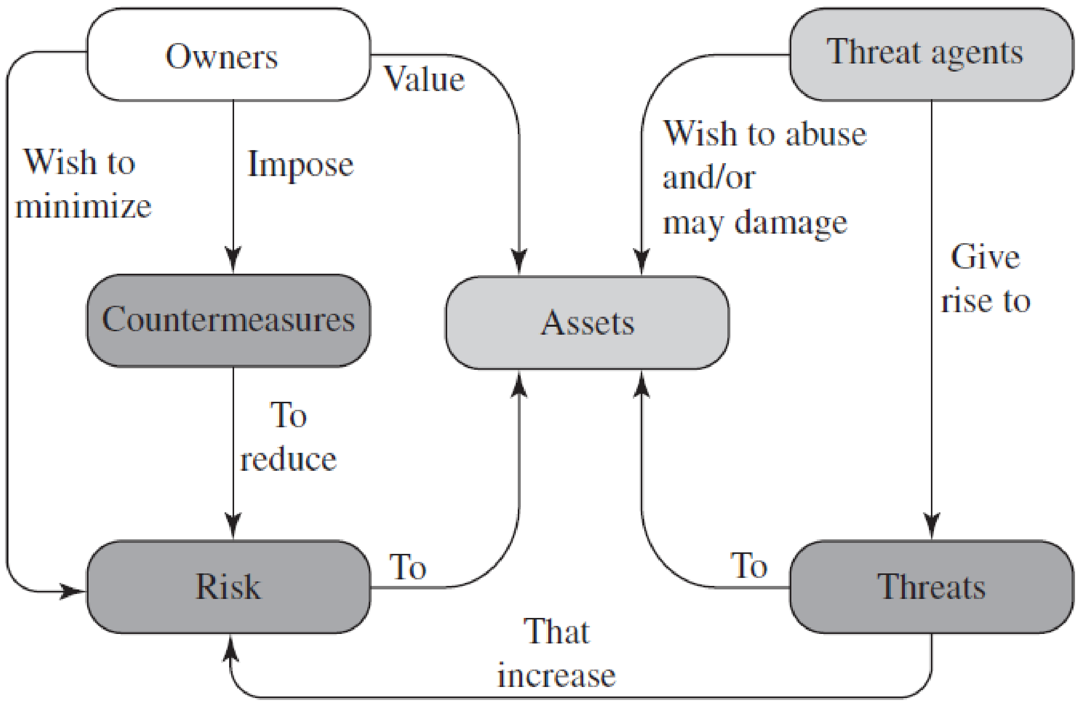


Figure 1.1 **Security Concepts and Relationships**

A Model for Computer Security

- General categories of Vulnerabilities
  - **Corrupted**
    - It does the wrong thing or gives wrong answers.
    - Eg: **Stored data values may differ** from what they should be because they have been improperly modified.
  - **Leaky**
    - Eg: **Someone who should not have access** to some or all of the information available through the network **obtains such access**.
  - **Unavailable**
    - Eg: Using the **system or network becomes impossible or impractical**.

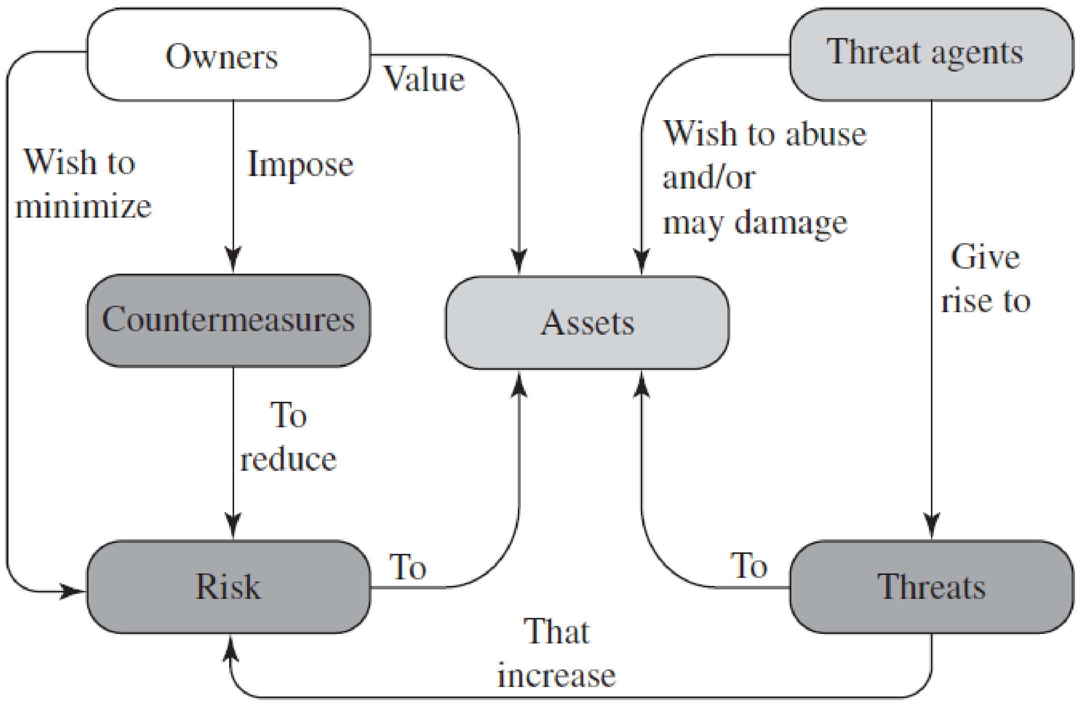


Figure 1.1 Security Concepts and Relationships

A Model for Computer Security

- Corresponding to the various types of vulnerabilities to a system resource are **threats that are capable of exploiting those vulnerabilities**.
- A threat represents a **potential security harm** to an asset.
- General caused by a Threat Agent.

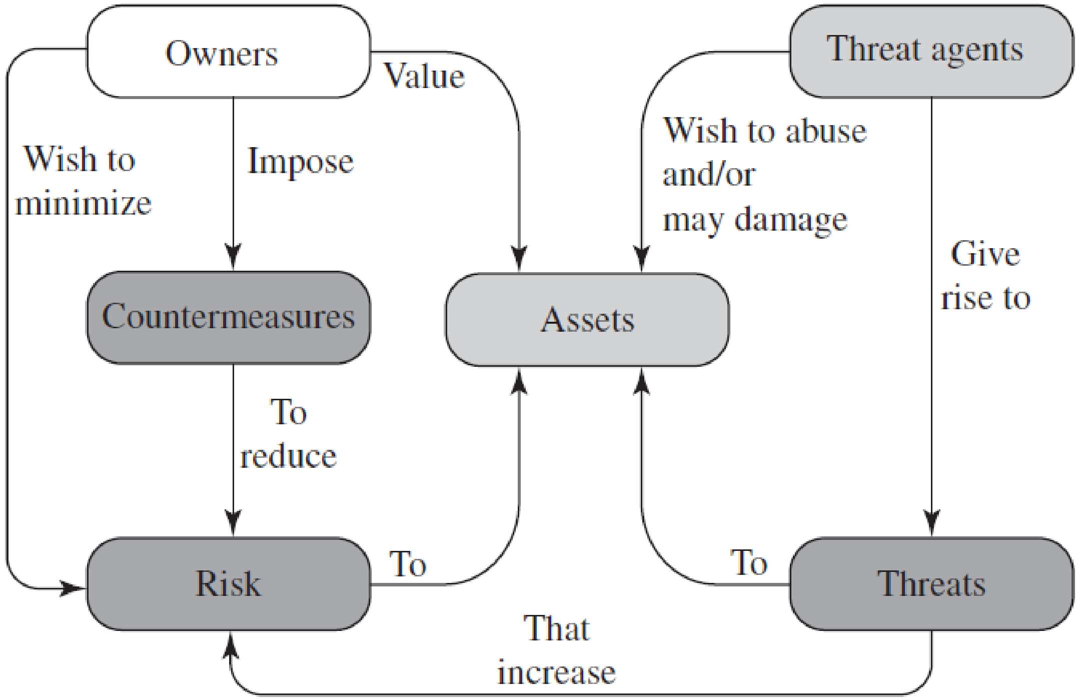


Figure 1.1 **Security Concepts and Relationships**

A Model for Computer Security

- An **attack is a threat that is carried out**.
- If successful, leads to an undesirable violation of security, or threat consequence.
- Two types of attacks
  - **Active attack**
    - An attempt to alter system resources or affect their operation.
  - **Passive attack**
    - An attempt to learn or make use of information from the system that does not affect system resources.
- Origin of the attack
  - **Inside attack**
    - The insider is authorized to access system resource but uses them in a way not approved by those who granted the authorization.
  - **Outside attack**
    - Done by an unauthorized or illegitimate user of the system

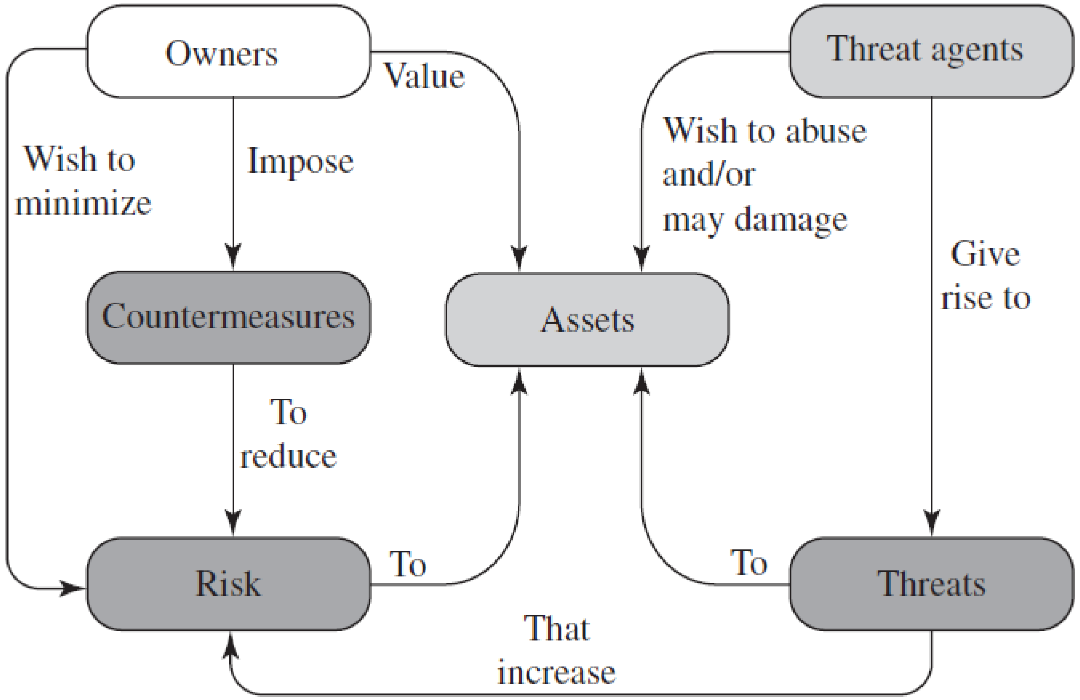


Figure 1.1 Security Concepts and Relationships

A Model for Computer Security

- Countermeasure is any **means taken to deal with a security attack**.
- A countermeasure can be devised to **prevent a particular type of attack** from succeeding.
- When prevention is not possible, or fails in some instance, the goal is to **detect the attack and then recover from the effects** of the attack.
- A countermeasure may itself introduce new vulnerabilities.
- In any case, residual vulnerabilities may remain after the imposition of countermeasures.

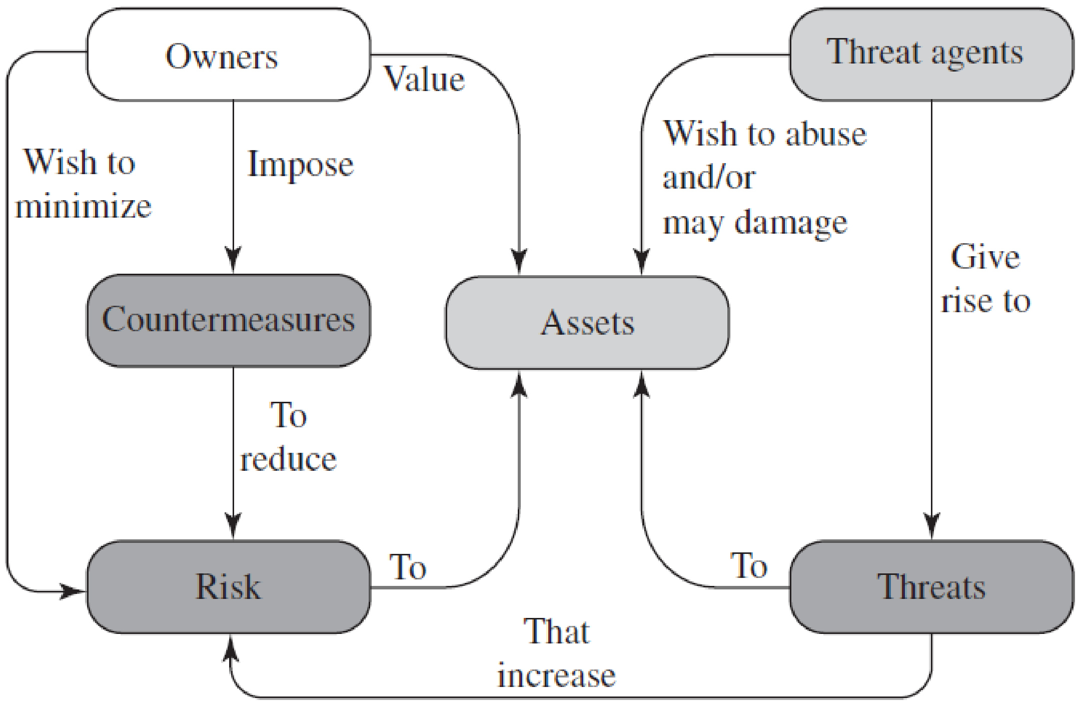


Figure 1.1 Security Concepts and Relationships

OSI Security Architecture

- ITU-T3 Recommendation X.800, Security Architecture for OSI, defines such a systematic approach.
- Computer and communications vendors have developed security features for their products and services that relate to this structured definition of services and mechanisms.
- The OSI security architecture focuses on security attacks, mechanisms, and services.
    - **Security attack**
        - Any **action that compromises the security of information** owned by an organization.
    - **Security mechanism**
        - **A process (or a device incorporating such a process) that is designed to detect, prevent, or recover from a security attack.**
    - **Security service**
        - A processing or communication service that **enhances the security of the data processing systems** and the information transfers of an organization.
        - The services are intended to counter security attacks, and they make use of one or more security mechanisms to provide the service.

Threats and Attacks

Threat Consequences	Attack
Unauthorized disclosure - threat to confidentiality	**1. Exposure** - **Sensitive data are directly released to an unauthorized** entity by an authorized person.
	2. Interception - An **unauthorized entity directly accesses sensitive data** traveling between authorized sources and destinations. - Using LAN
	3. Inference - Adversary is able to **gain information from observing the pattern of traffic on a network**. - Gets useful information from the analysis.
	4. Intrusion - An unauthorized entity gains access to sensitive data by **circumventing a system's security protections.**

Threats and Attacks

Threat Consequences	Attack
Deception • A circumstance or event that may result in an **authorized entity receiving false data and believing it to be true.** • Threat to Integrity	**1.Masquerade** • An **unauthorized entity gains access** to a system or performs a **malicious act by posing as an authorized entity.** • Eg: Unauthorized user has **learned another user's logon ID and password**
	2. Falsification • This refers to the **altering or replacing of valid data** or the **introduction of false data** into a file or database. • For example, **a student may alter his or her grades** on a school database.
	3. Repudiation • A user either **denies sending data or a user denies receiving** or possessing the data.

Threats and Attacks

Threat Consequences	Attack
Disruption • A circumstance or **event that interrupts or prevents the correct operation** of system services and functions. • Threat to availability or system integrity	**1.Incapacitation** • **Prevents or interrupts system operation** by **disabling a system component.** • Attack on system availability. • **Physical destruction of or damage** to system hardware • **Trojan horses, viruses, or worms**
	2.Corruption • Undesirably **alters system operation** by **adversely modifying system functions or data**. • Attack on system integrity
	3. Obstruction • A threat action that **interrupts delivery of system services by hindering system operation**. • Eg: By disabling communication links or altering communication control information.

Threats and Attacks

Threat Consequences	**Attack**
Usurpation - **Unauthorized entity controls the system functionality** - Threat to system integrity	**1.Misappropriation** - This can include **theft of service**. - The **malicious software makes unauthorized use of processor and operating system resources**.
	2. Misuse - Misuse can occur by means of either **malicious logic or a hacker that has gained unauthorized access** to a system. - In either case, **security functions can be disabled** or thwarted.

Threats and Assets – Hardware

- Major threat is **threat to availability**
  - Accidental and deliberate damage to equipment
  - Theft
- Theft of CD-ROMs, DVDs, Pendrives and other storage devices can lead to **loss of confidentiality**.
- **Physical and administrative security measures are needed to deal with these threats.**

Threats and Assets - Software

- System Software + Application Software
- A key threat to software is an attack on **availability**.
  - Application software, is often **easy to delete**.
  - Software can also be **altered or damaged to render it useless**.
  - Careful software configuration management, which includes making **backups of the most recent version of software, can maintain high availability.**
- **Threat to integrity**
  - **Software modification** that results in a program that still functions but that **behaves differently than before.**
  - Computer viruses and related attacks fall into this category
- **Protection against software piracy**

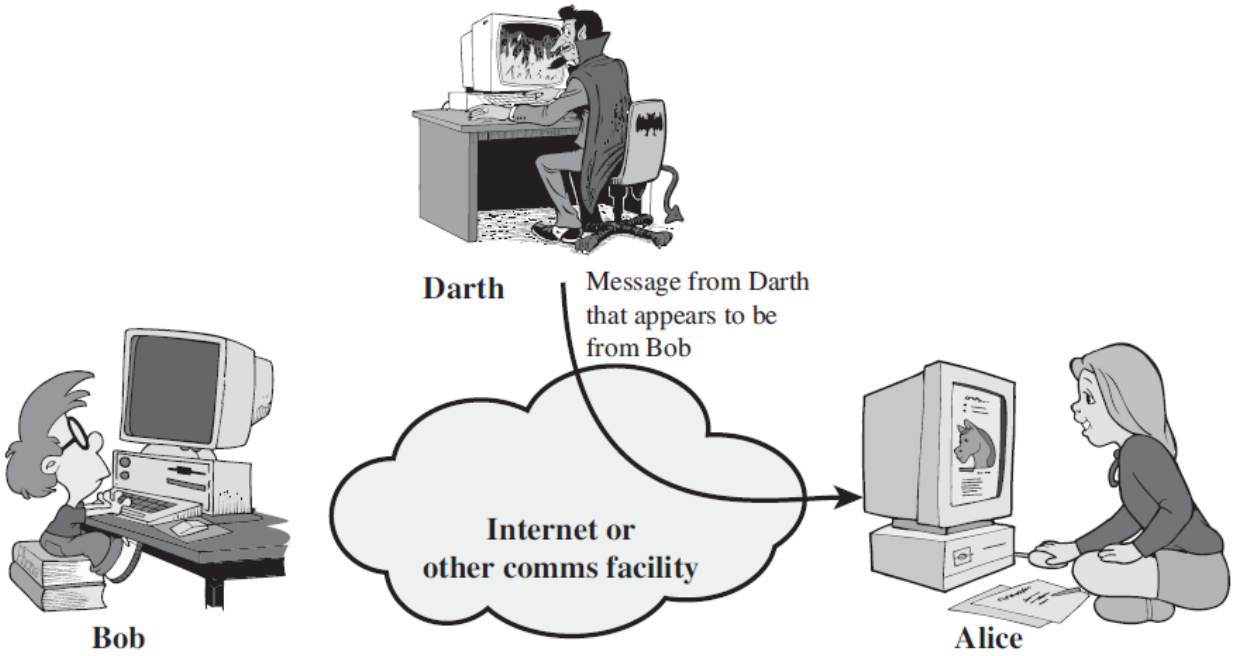
Threats and Assets - Data

- **Availability**
  - **Destruction of data files**, which can occur either accidentally or maliciously.
- **Confidentiality**
  - **Unauthorized reading** of data files or databases.
    - **Summary or aggregate information**
      - **Leakage in personal information**
- **Integrity**
  - Modifications to data files can have **consequences ranging from minor to disastrous**.

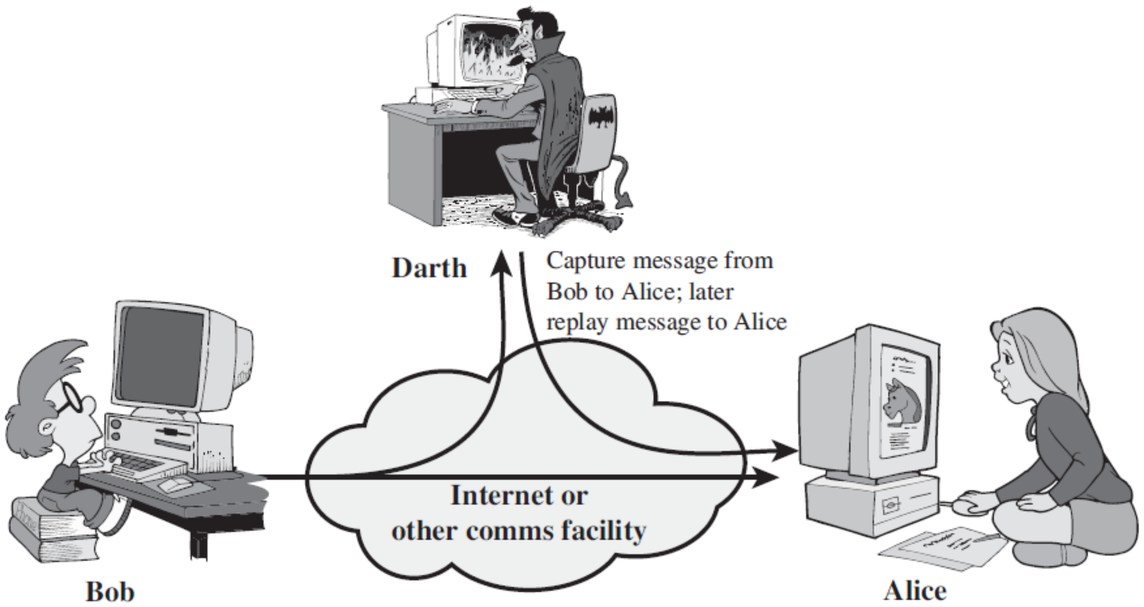
Threats and Assets - Communication Lines and Networks

- **Active Attack**
  - Modification in the Data Stream
  - Types
    - **Replay**
      - Passive capture of a data unit and its subsequent retransmission to produce an unauthorized effect.
    - **Masquerade**
      - One entity pretends to be a different entity (Impersonation)
    - **Modification of messages**
      - Some portion of a legitimate message is altered.
    - **Denial of service**
      - Inhibits the normal use or management of communication facilities.
      - Disabling the network or by overloading it with messages so as to degrade performance.
- **Passive Attack**
  - Eavesdropping on, or monitoring of, transmissions.
  - Goal: Obtain the information that is being transmitted
  - Types:
    - **Release of message contents**
    - **Traffic analysis**
      - Gathering the data from the packet
  - Very difficult to detect the attack.

Active Attacks - Illustration

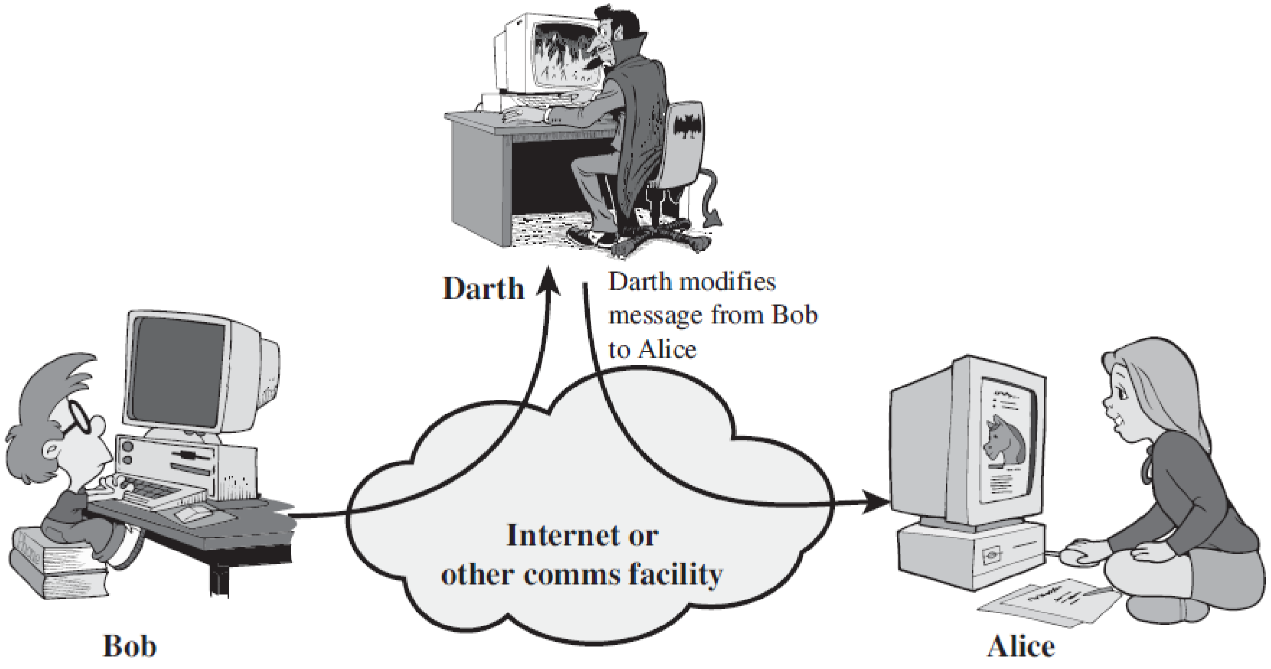

Darth
Message from Darth that appears to be from Bob
Internet or other comms facility
Bob
Alice

(a) Masquerade


Darth
Capture message from Bob to Alice; later replay message to Alice
Internet or other comms facility
Bob
Alice

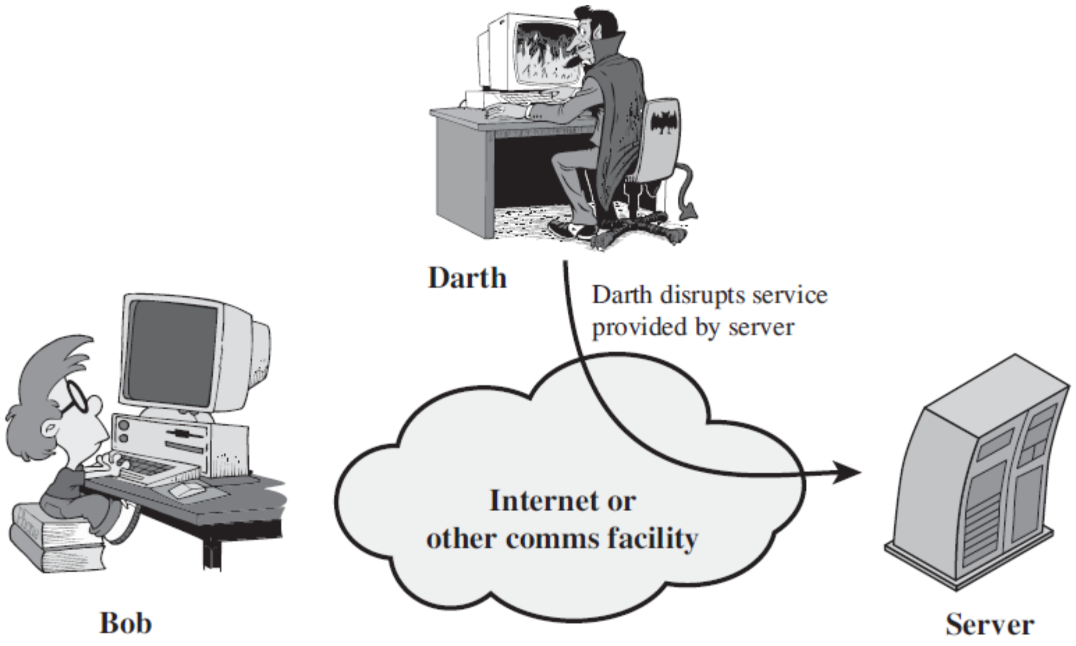
(b) Replay

Active Attacks - Illustration



Darth modifies message from Bob to Alice

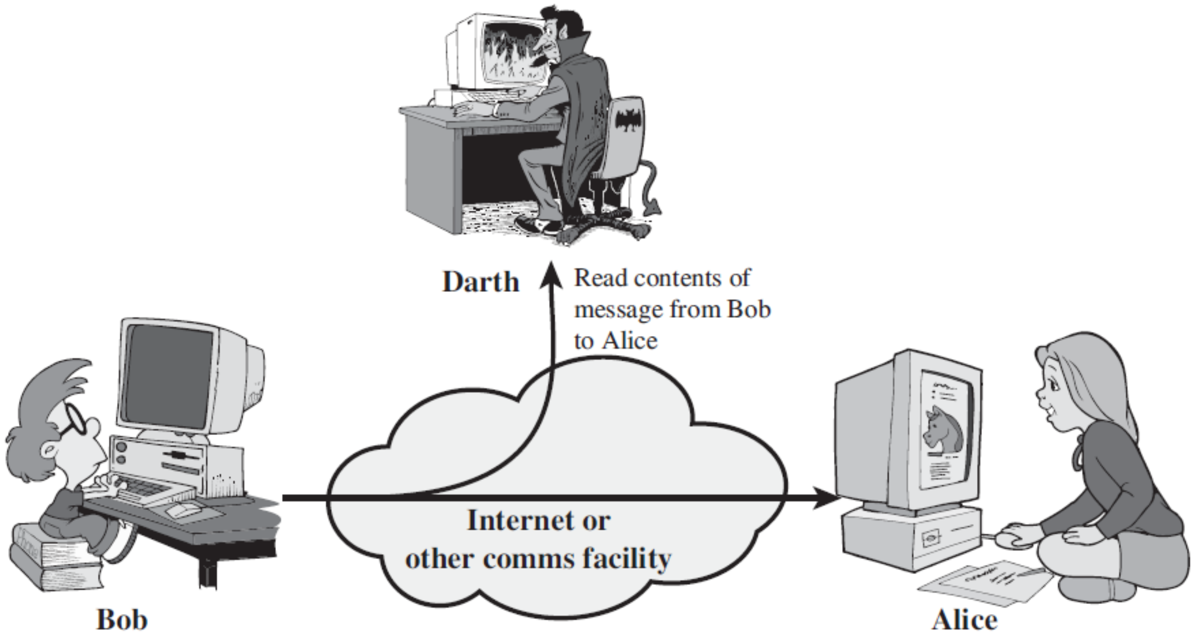
(c) Modification of messages



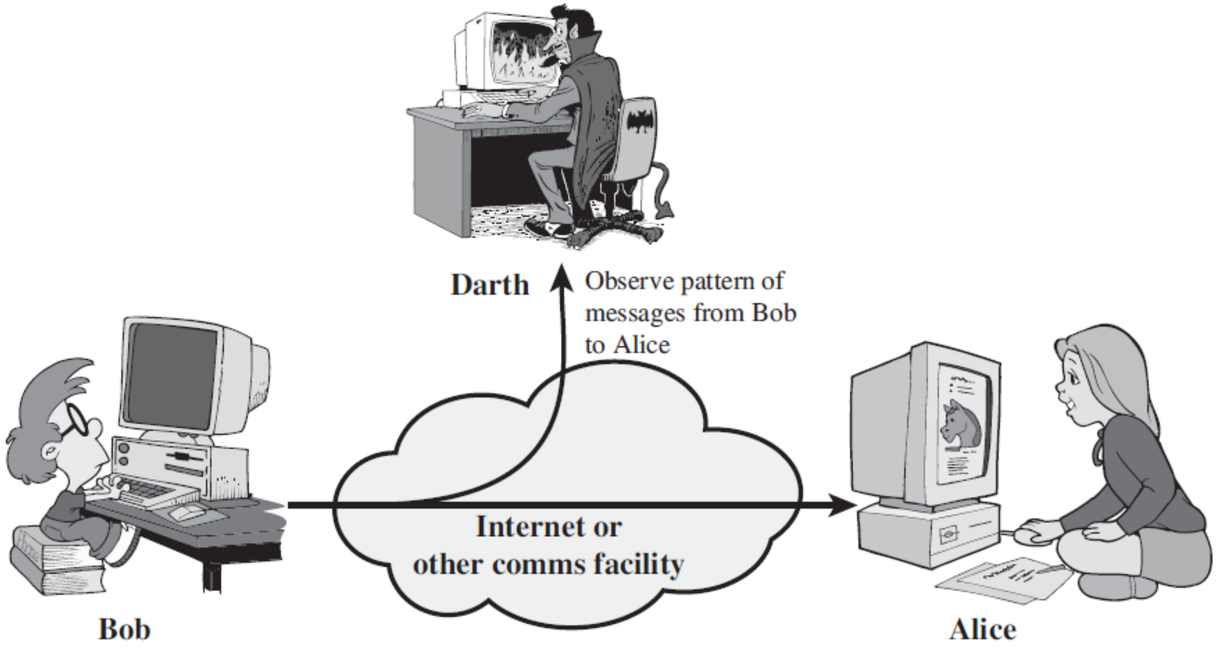
Darth disrupts service provided by server

(d) Denial of service

Passive Attacks - Illustration

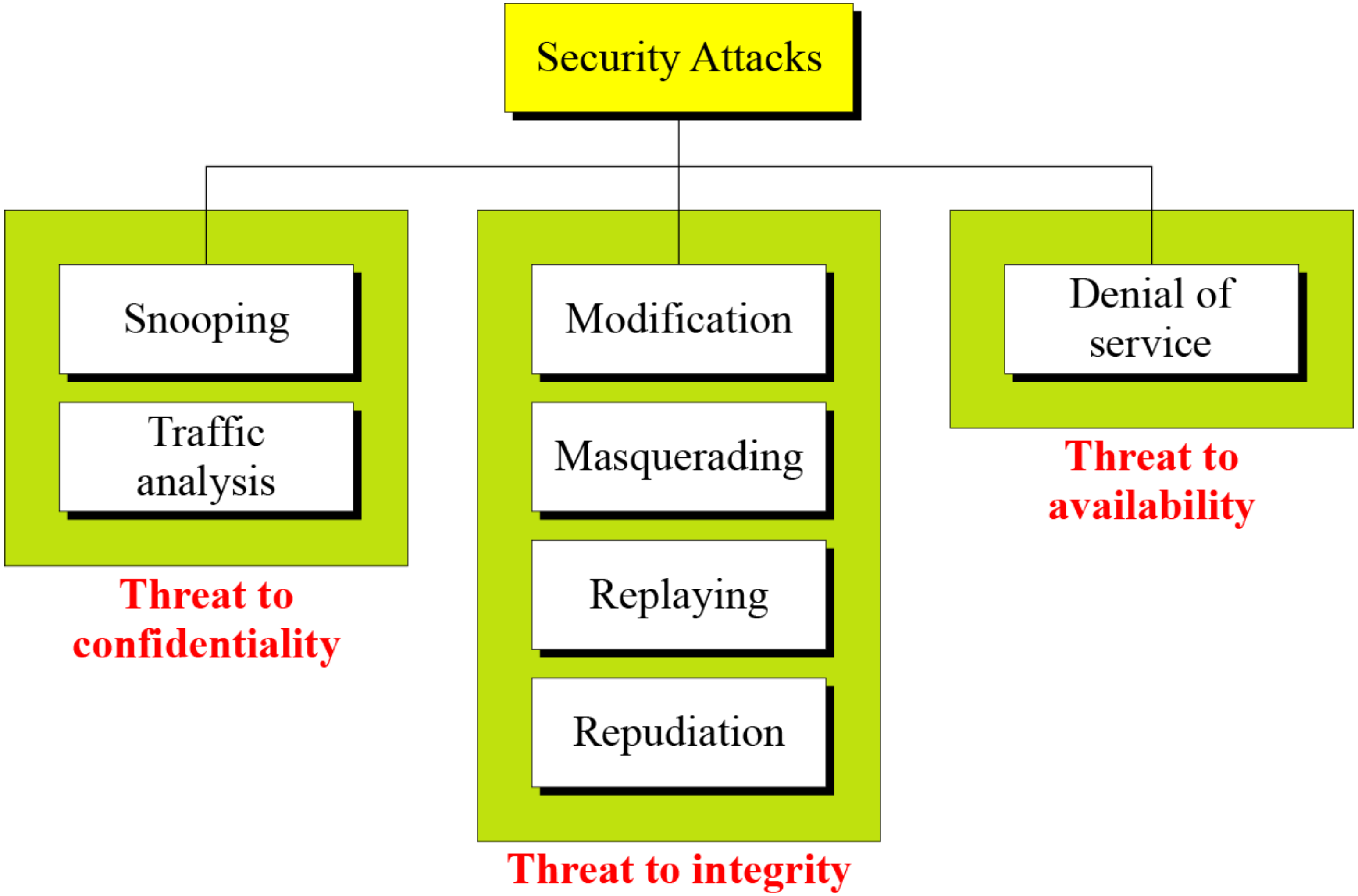

Darth
Read contents of message from Bob to Alice
Internet or other comms facility
Bob
Alice

(a) Release of message contents

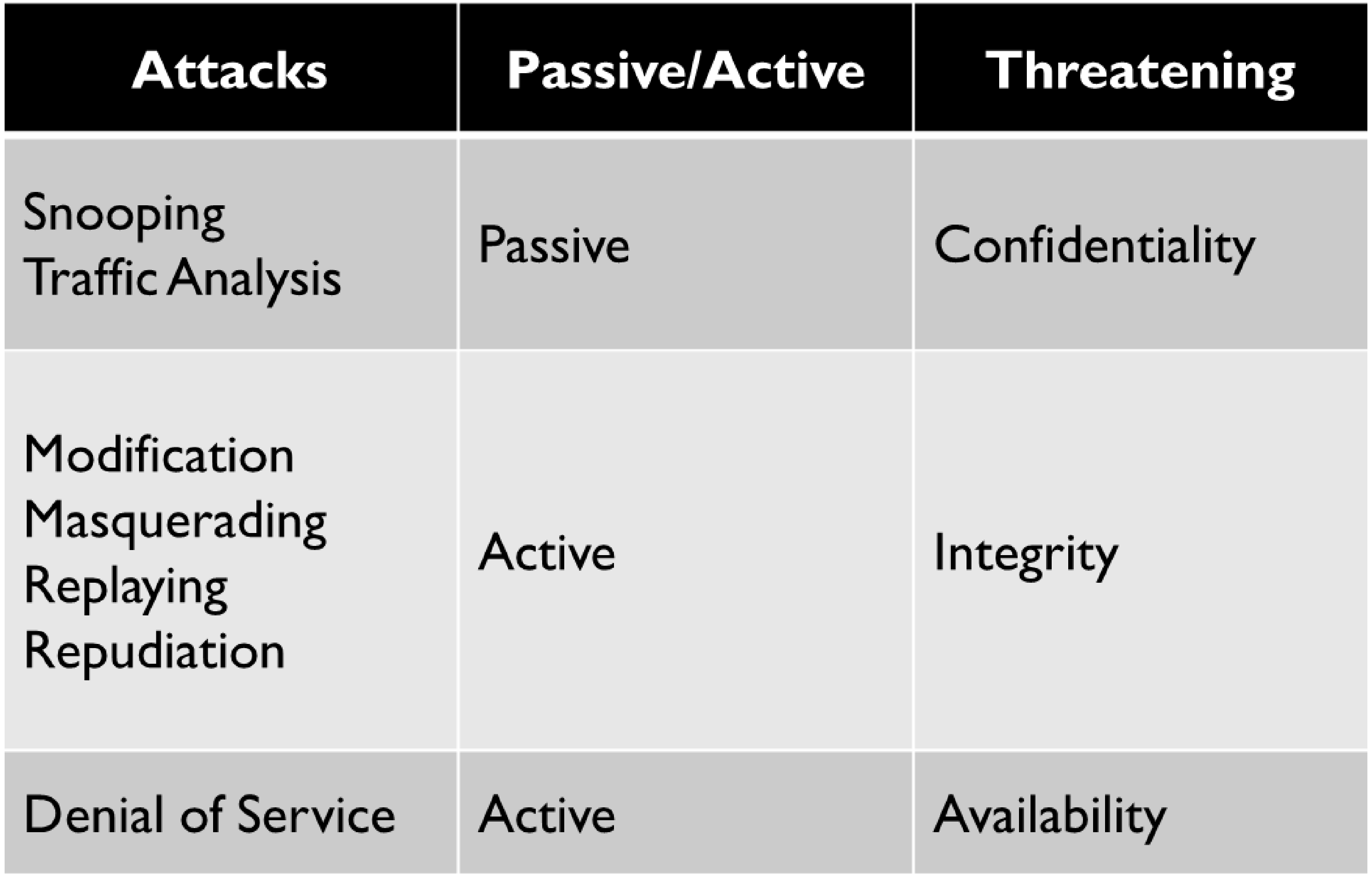

Darth
Observe pattern of messages from Bob to Alice
Internet or other comms facility
Bob
Alice

(b) Traffic analysis

Taxonomy of Service Attacks



Summary of Attacks



Attacks	Passive/Active	Threatening
Snooping Traffic Analysis	Passive	Confidentiality
Modification Masquerading Replaying Repudiation	Active	Integrity
Denial of Service	Active	Availability

Common Types of Attacks

1. Malware
2. Phishing
3. Man-in-the-Middle (MitM) Attacks
4. Denial-of-Service (DOS) Attack
5. SQL Injections
6. Zero-day Exploit
7. Password Attack
8. Cross-site Scripting
9. Rootkits
10. Internet of Things (IoT) Attacks
11. DNS Tunneling

1.Malware

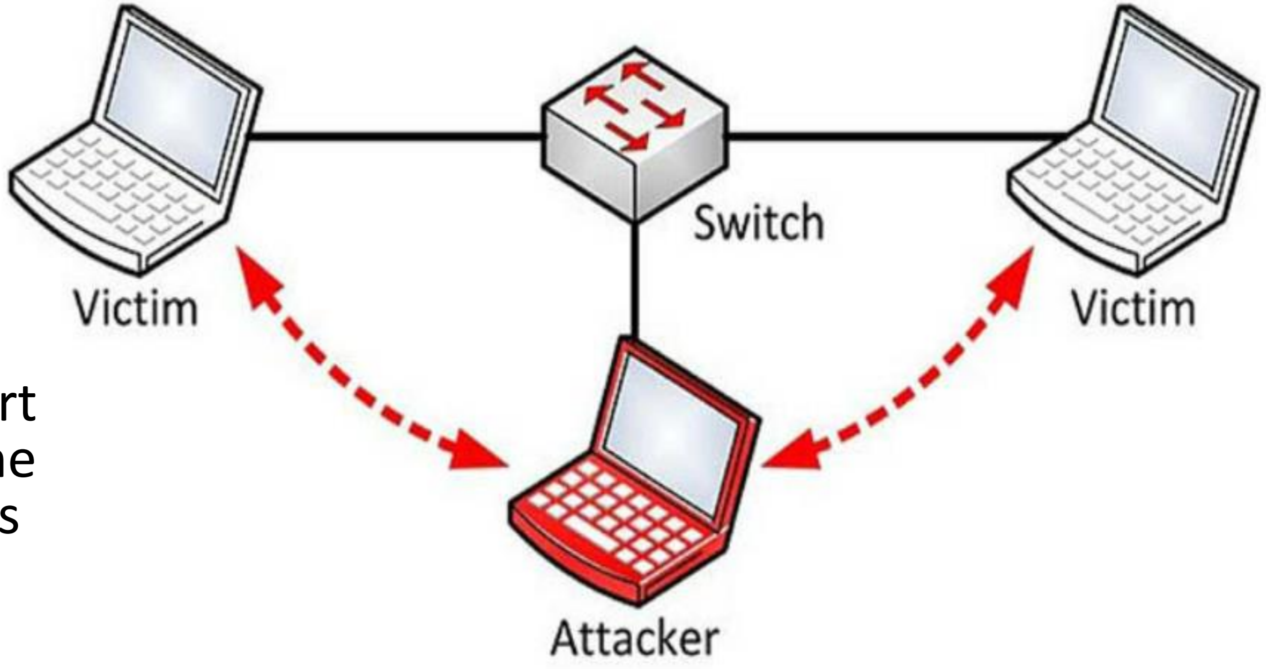
- Malware uses a vulnerability to breach a network -> install malicious software
  - **Deny access to the critical components of the network**
  - **Obtain information by retrieving data from the hard drive**
  - **Disrupt the system or even render it inoperable**
- Common Types of Malware
  - **Viruses**
    - **Infect applications** attaching themselves to the initialization sequence
    - Replicates itself, infecting other code in the computer system
  - **Trojans**
    - **Program hiding inside a useful program with malicious purposes**
    - Trojan doesn’t replicate itself and it is commonly used to establish a **backdoor** to be exploited
  - **Worms**
    - Self-contained programs that **propagate across networks and computers**
    - installed through email attachments, sending a copy of themselves to every contact in the infected computer email list
    - DoS Attack
  - **Ransomware**
    - Denies access to the victim data, **threatening to publish or delete it unless a ransom is paid**.
  - **Spyware**
    - Program installed to **collect information about users, their systems or browsing habits,** sending the data to a remote user.

2. Phishing

- Phishing is the **practice of sending fraudulent communications** that appear to come from a **reputable source**, usually through email.
- Link -> appears to be legit -> once clicked, the personal data (Authentication information is extracted)
- Types
  - **Email Phishing**
    - Register a **fake domain that mimics a genuine organization** and sends thousands of generic requests.
  - **Spear phishing**
    - Targeted attacks **directed at specific companies and/or individuals**. (Some information about the victim is already with them)
  - **Whaling**
    - Attacks targeting **senior executives and stakeholders within an organization**.
  - **Smishing and vishing**
    - Smishing involves criminals sending **text messages**
    - Vishing involves **telephonic conversations**
  - **Angler phishing**
    - **Fake URLs; cloned websites, posts, and tweets; and instant messaging**
  - **Pharming**
    - Capture **user credentials through a fake login landing page**

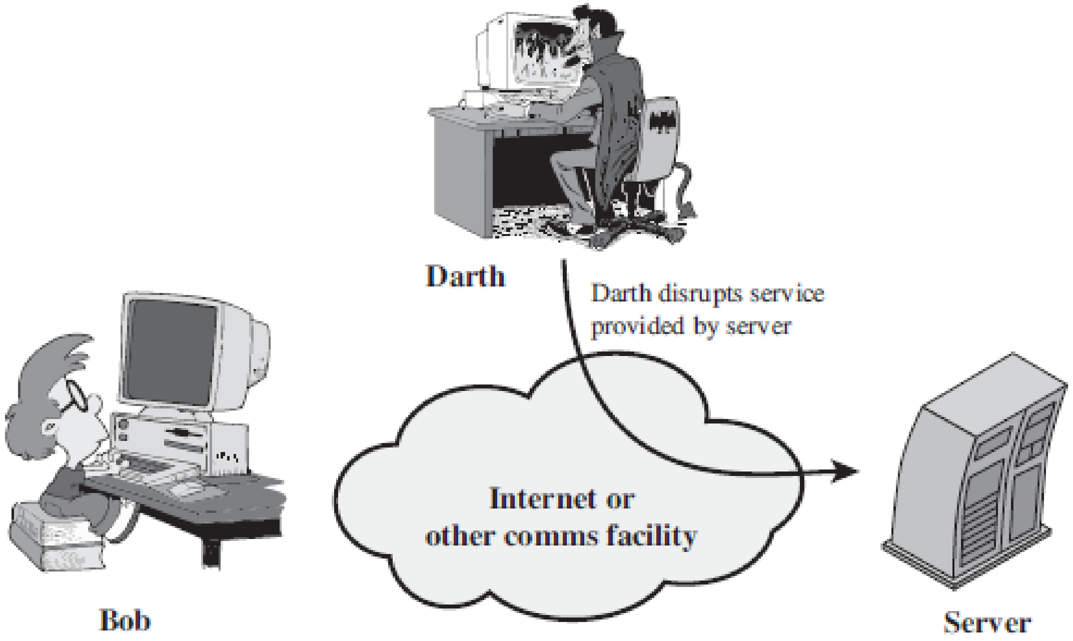
3. Man-in-the-Middle (MitM) Attacks

- Also known as **eavesdropping attacks**
- Occur when **attackers insert themselves into a two-party transaction**.
- Two common points of entry for MitM attacks:
  - On **unsecure public Wi-Fi**, attackers can insert themselves between a visitor's device and the network. Without knowing, the visitor passes all information through the attacker.
  - Once **malware has breached a device, an attacker can install software** to process all of the victim's information.



4. Denial-of-Service (DOS) Attack

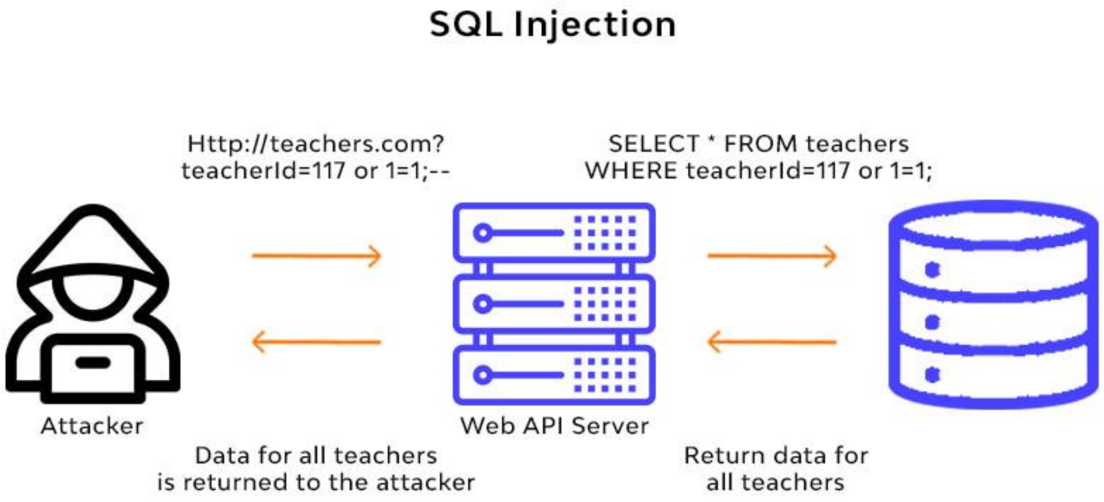
- DoS attacks work by **flooding systems, servers, and/or networks with traffic to overload resources and bandwidth**.
- The system will be unable to process and fulfill legitimate requests.
- Types
  - **Application-layer Flood**
    - An attacker simply floods the service with requests from a spoofed IP address in an attempt to slow or crash the service
  - **Distributed Denial of Service Attacks (DDoS)**
    - Requests are sent from many clients.
    - DDoS attacks often involve many "zombie" machines which send massive amounts of requests to a service to disable it.
    - Once service is disabled another attack to steal the data can be initiated.
- Other Attacks
  - TCP SYN flood attack, teardrop attack, smurf attack, ping-of-death attack, and botnets.


Darth
Darth disrupts service provided by server
Internet or other comms facility
Bob
Server

DOS Attack

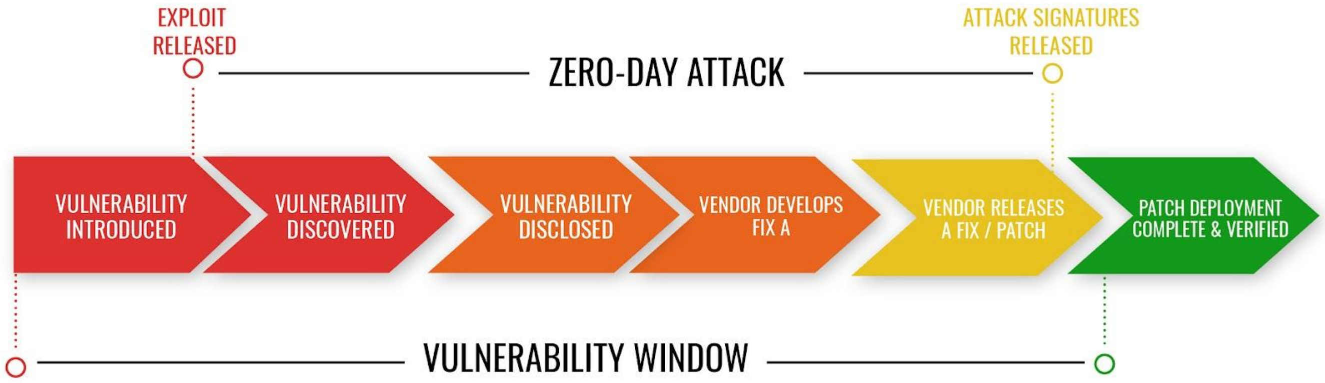
5. SQL Injections

- Attacker **inserts malicious code into a server using server query language (SQL)** forcing the **server to deliver protected information**.
- An attacker could carry out a SQL injection simply by **submitting malicious code into a vulnerable website search box.**



6.Zero-day exploit

- A zero-day exploit **hits after a network vulnerability is announced but before a patch or solution is implemented**.
- Attackers target the disclosed vulnerability during this window of time.

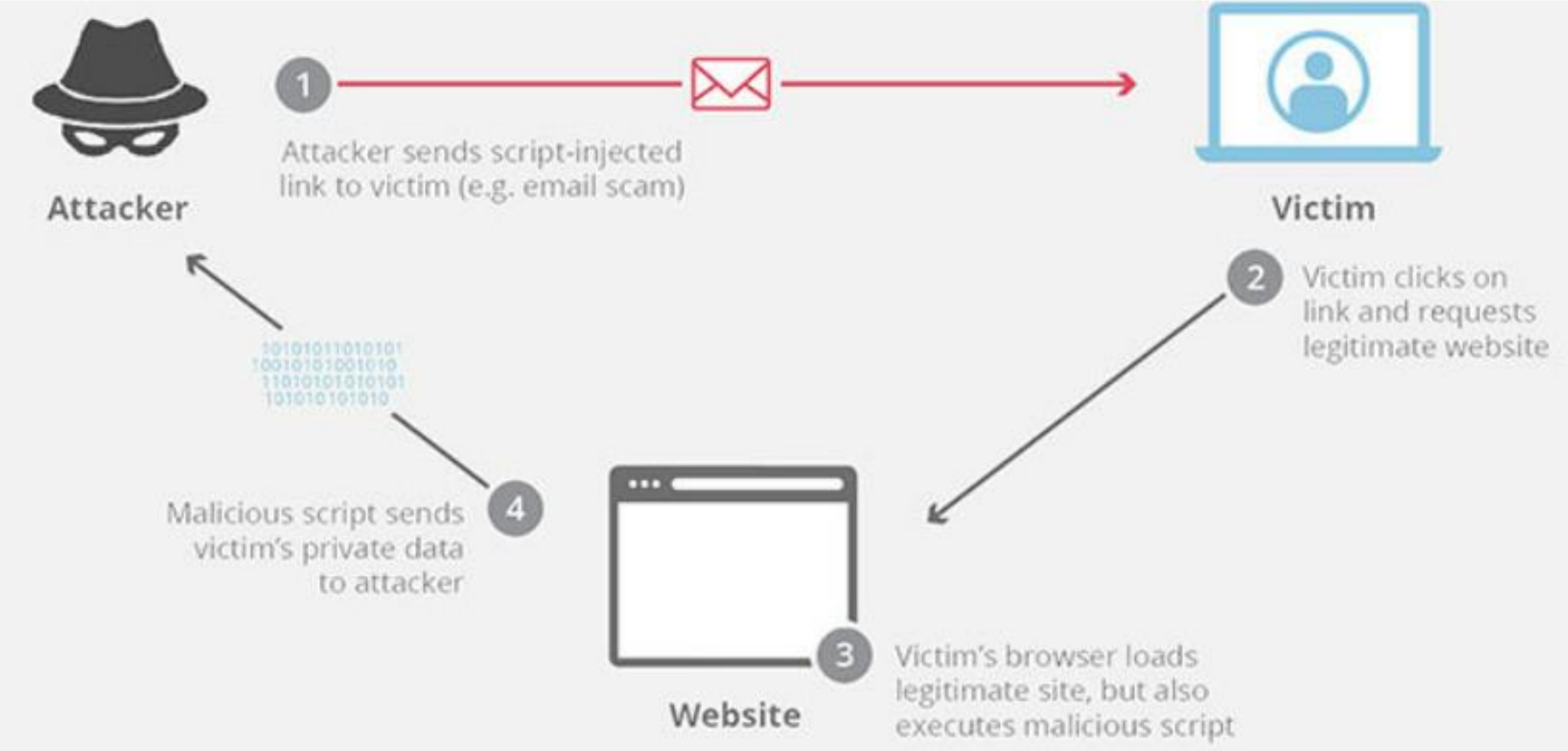


7. Password Attack

- By accessing a person’s password, an **attacker can gain entry to confidential or critical data and systems, including the ability to manipulate** and control said data/systems.
- Password attackers use a myriad of methods to identify an individual password,
  - Social engineering
  - Gaining access to a password database
  - Testing the network connection to obtain unencrypted passwords
  - Simply by guessing
  - Brute-force attack – All possible combinations
  - Dictionary attack - When the attacker uses a list of common passwords to attempt to gain access to a user’s computer and network.

8. Cross-site Scripting

- A cross-site scripting attack sends **malicious scripts into content from reliable websites**.
- The malicious code joins the dynamic content that is sent to the victim's browser.
- Usually, this malicious code consists of Javascript code executed by the victim's browser, but can include Flash, HTML, and XSS.



Attacker
1
Attacker sends script-injected link to victim (e.g. email scam)
Victim
2
Victim clicks on link and requests legitimate website
3
Victim's browser loads legitimate site, but also executes malicious script
Website
4
Malicious script sends victim's private data to attacker

9.Rootkits

- Rootkits are **installed inside legitimate software**, where they can gain remote control and administration-level access over a system.
- The **attacker then uses the rootkit to steal passwords, keys, credentials**, and retrieve critical data.

10. IOT Attacks

- The **interconnectedness of things makes it possible for attackers to breach an entry point** and use it as a gate to exploit other devices in the network.

11. DNS Tunneling

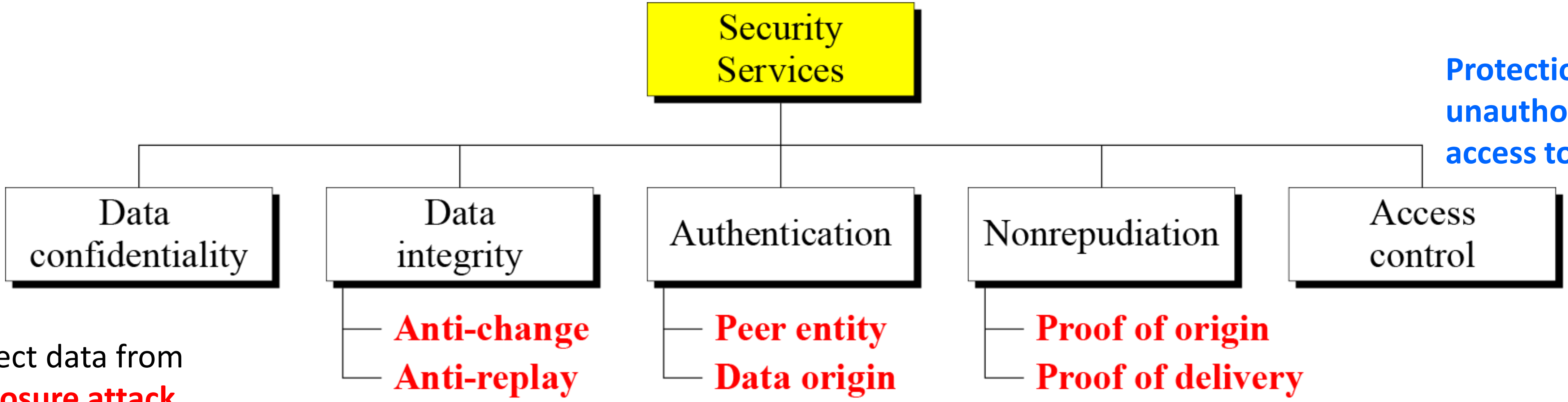
- DNS is like a phonebook for the internet, helping to translate between IP addresses and domain names.
- DNS tunneling takes advantage of this fact by using **DNS requests to implement a command and control channel for malware.**
- **Inbound DNS traffic can carry commands** to the malware, while **outbound traffic can exfiltrate sensitive data** or provide responses to the malware operator’s requests.

Security Services

- **International Telecommunication Union-Telecommunication Standardization Sector** (ITU-T) provides security services and mechanism for implementing them.
- Security services and mechanisms are closely related because a mechanism or combination of mechanisms are used to provide a service.
- A mechanism can be used in one or more services.
- Types
  - **X.800**
    - Service provided by a **protocol layer of communication**
  - **RFC 2828**
    - Communication service that is provided by a system to give a **specific kind of protection to system resources.**

Security Services (Contd…)

- **ITU-T (X.800)** has defined five services related to the security goals and attacks we defined in the previous sections.



Security Services

- Data confidentiality
- Data integrity
  - **Anti-change**
  - **Anti-replay**
- Authentication
  - **Peer entity**
  - **Data origin**
- Nonrepudiation
  - **Proof of origin**
  - **Proof of delivery**
- Access control

Protection against unauthorized access to data.

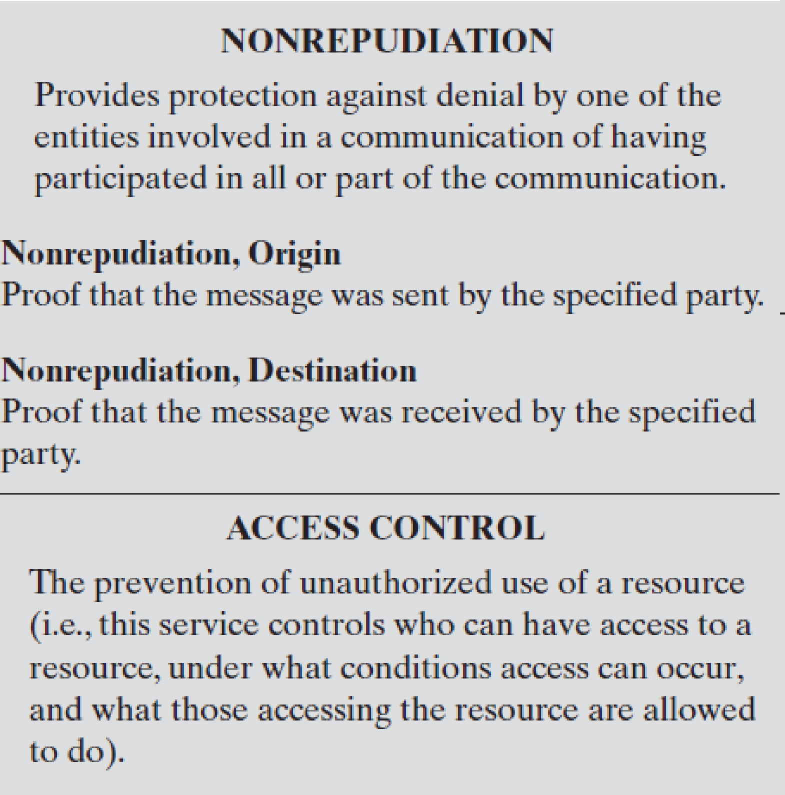
- Protect data from **disclosure attack, snooping and traffic analysis attack.**
- Defined by X.800 - encompasses **confidentiality of the whole message or part of a message.**

- Designed **to protect data from modification, insertion, deletion, and replaying.**
- Protect the **whole message or part of the message.**

- **Peer entity authentication:** Authentication of the sender or receiver -connection establishment.
- **Data origin authentication:** Authenticates the source of the data

Proof of the origin:
The receiver of the data can later **prove the identity of the sender** if denied.
Proof of delivery:
Sender of data can later prove that data were **delivered to the intended recipient.**

Security Services (Contd…)



NONREPUDIATION

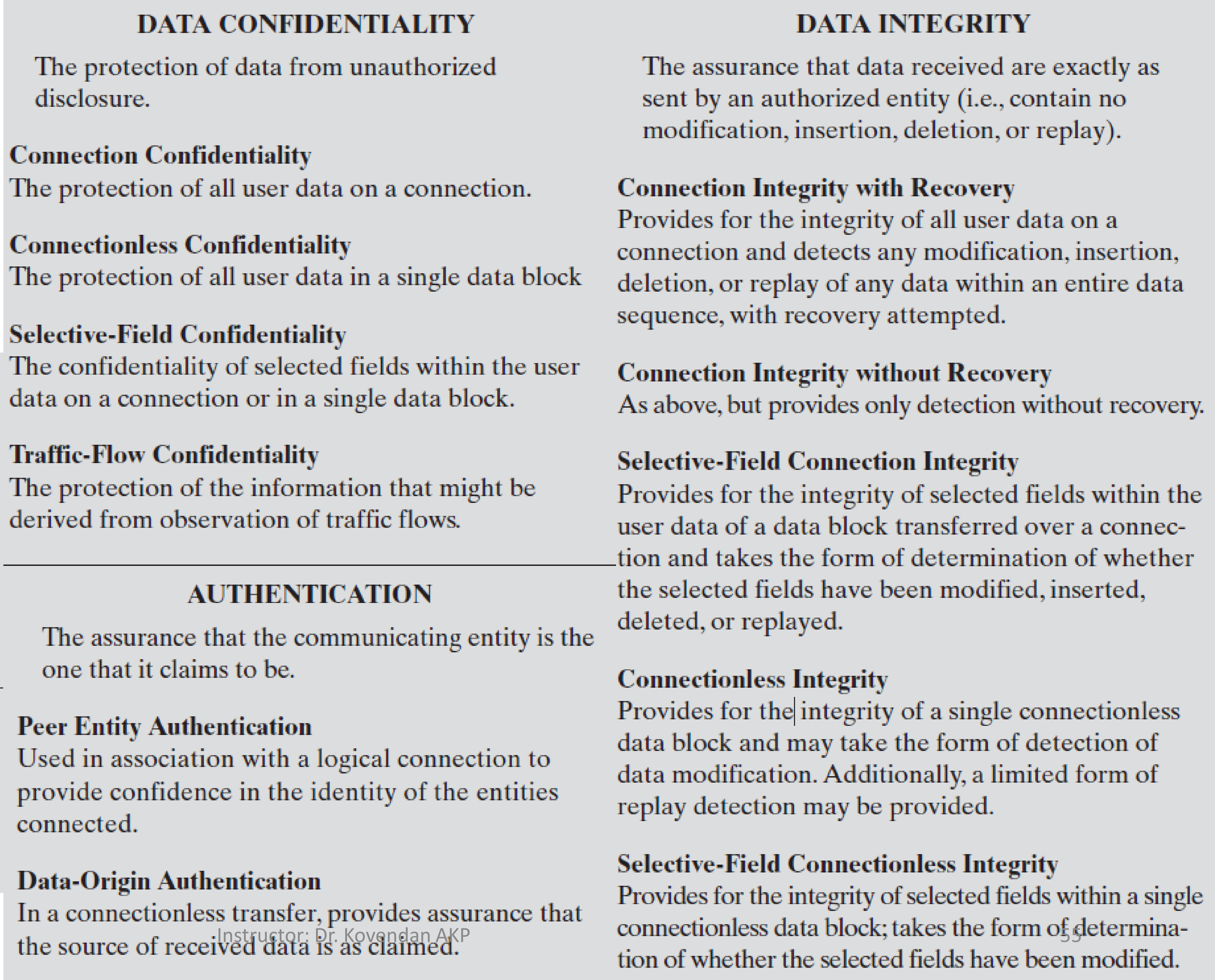
Provides protection against denial by one of the entities involved in a communication of having participated in all or part of the communication.

Nonrepudiation, Origin
Proof that the message was sent by the specified party.

Nonrepudiation, Destination
Proof that the message was received by the specified party.

ACCESS CONTROL

The prevention of unauthorized use of a resource (i.e., this service controls who can have access to a resource, under what conditions access can occur, and what those accessing the resource are allowed to do).



DATA CONFIDENTIALITY

The protection of data from unauthorized disclosure.

Connection Confidentiality
The protection of all user data on a connection.

Connectionless Confidentiality
The protection of all user data in a single data block

Selective-Field Confidentiality
The confidentiality of selected fields within the user data on a connection or in a single data block.

Traffic-Flow Confidentiality
The protection of the information that might be derived from observation of traffic flows.

AUTHENTICATION

The assurance that the communicating entity is the one that it claims to be.

Peer Entity Authentication
Used in association with a logical connection to provide confidence in the identity of the entities connected.

Data-Origin Authentication
In a connectionless transfer, provides assurance that the source of received data is as claimed.

DATA INTEGRITY

The assurance that data received are exactly as sent by an authorized entity (i.e., contain no modification, insertion, deletion, or replay).

Connection Integrity with Recovery
Provides for the integrity of all user data on a connection and detects any modification, insertion, deletion, or replay of any data within an entire data sequence, with recovery attempted.

Connection Integrity without Recovery
As above, but provides only detection without recovery.

Selective-Field Connection Integrity
Provides for the integrity of selected fields within the user data of a data block transferred over a connection and takes the form of determination of whether the selected fields have been modified, inserted, deleted, or replayed.

Connectionless Integrity
Provides for the integrity of a single connectionless data block and may take the form of detection of data modification. Additionally, a limited form of replay detection may be provided.

Selective-Field Connectionless Integrity
Provides for the integrity of selected fields within a single connectionless data block; takes the form of determination of whether the selected fields have been modified.

Security Mechanism

Incorporated into the **appropriate protocol layer** in order to provide some of the OSI security services.

Security Mechanism

Mechanisms that are **not specific to any particular OSI security service or protocol layer.**

Specific Security Mechanism

Encipherment: Mathematical algorithms to transform data.

Digital Signature: To prove the source and integrity of the data unit and protect against forgery

Access Control: Enforce access rights to resources.

Data Integrity: Assure the integrity of a data unit or stream of data units.

Authentication Exchange: Ensure the identity of an entity by means of information exchange.

Traffic Padding: The insertion of bits into gaps in a data stream.

Routing Control: Selection of particular physically secure routes.

Notarization: Trusted third party to assure certain properties of a data exchange

Pervasive Security Mechanism

Trusted Functionality: That which is perceived to be correct with respect to some criteria.

Security Label: Designates the security attributes of tha resource.

Event Detection: Detection of security-relevant events.

Security Audit Trail: Data collected and potentially used to facilitate a security audit, which is an independent review and examination of system records and activities.

Security Recovery: Deals with requests from mechanisms, such as event handling and management functions, and takes recovery actions.

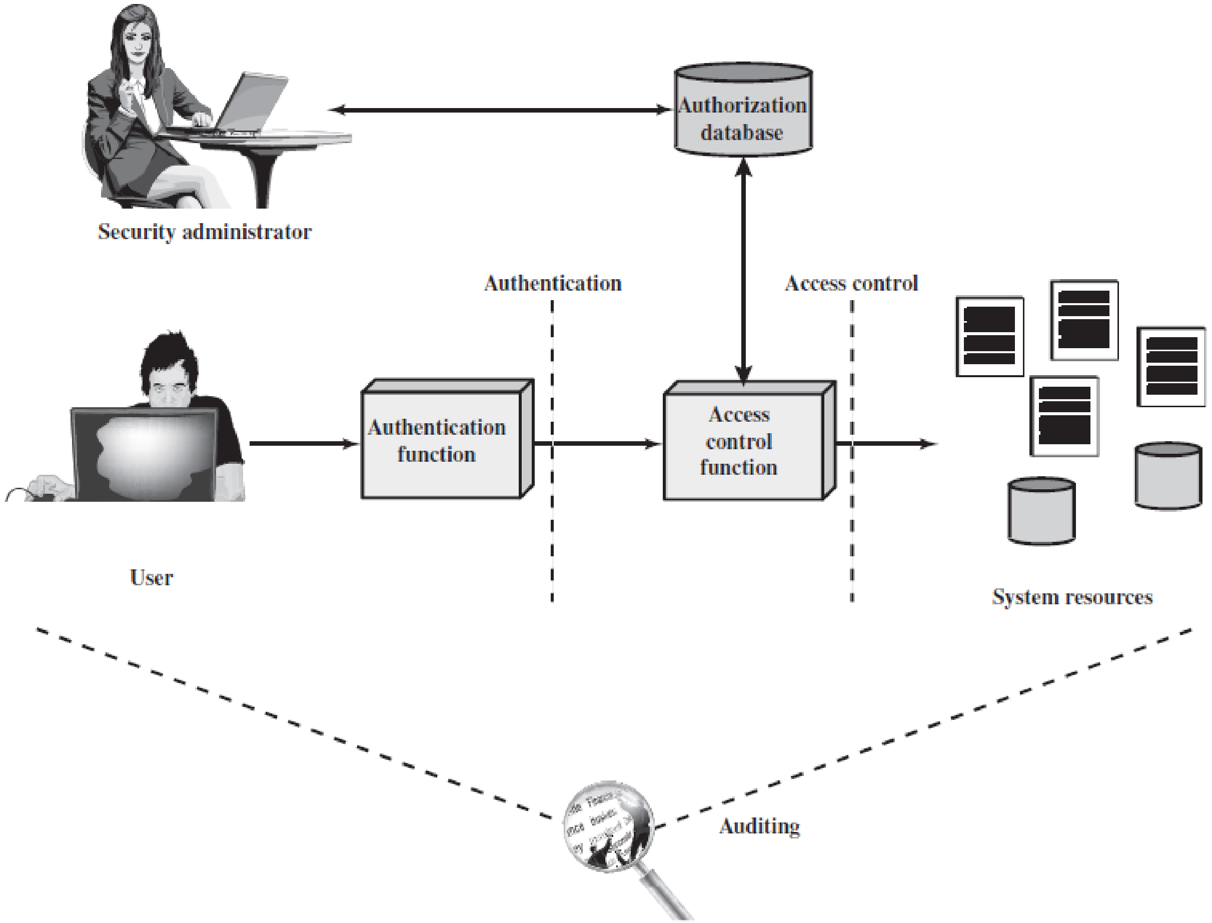
Access Control

- **NIST IR 7298**
  - Defines access control as the **process of granting or denying specific requests** to:
  - (1) **Obtain and use information** and related information processing services; and
  - (2) **Enter specific physical facilities**.
- **RFC 4949**
  - Defines access control as **a process by which use of system resources is regulated according to a security policy and is permitted only by authorized entities** (users, programs, processes, or other systems) according to that policy.

Access Control Principles

Access Control Context

- **Authentication:**
  - **Verification that the credentials** of a user or other system entity are **valid.**
- **Authorization:**
  - The **granting of a right or permission to a system entity** to access a system resource.
  - This function determines who is trusted for a given purpose.
- **Audit:**
  - An **independent review and examination of system records** and activities in order to test for adequacy of system controls.



Access Control Principles

Access Control Policies

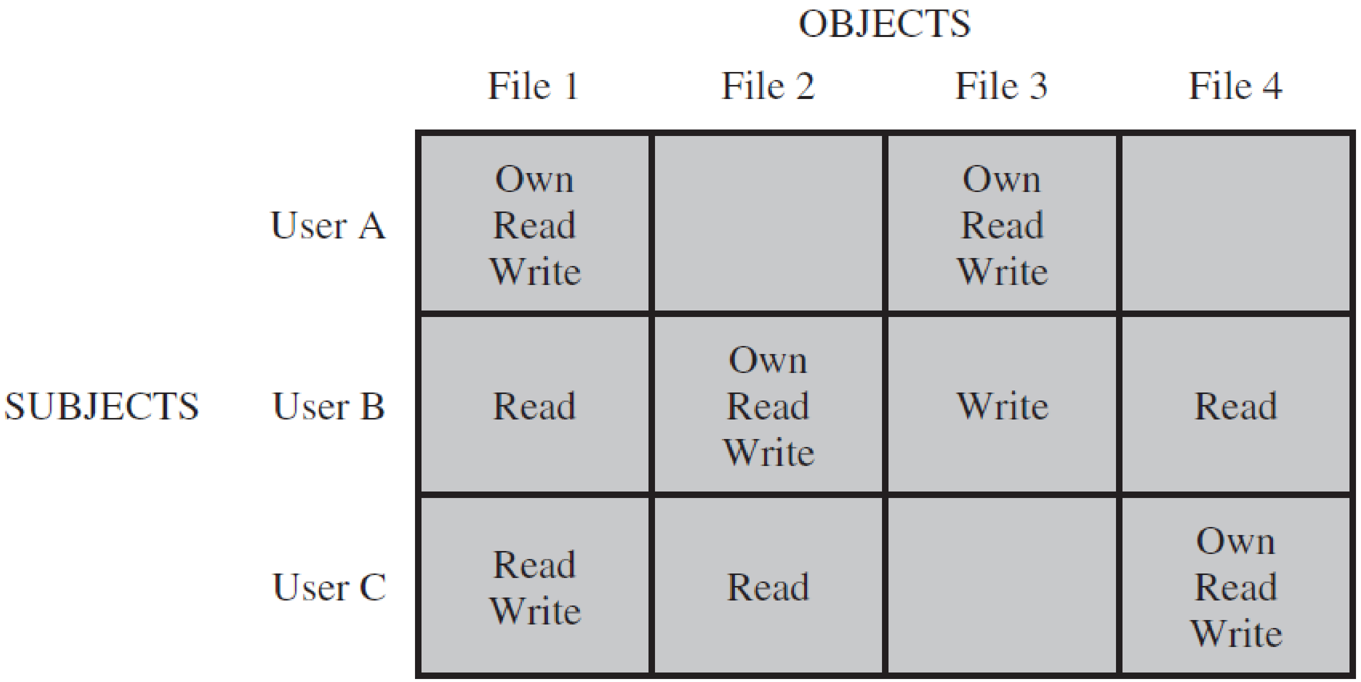
- What types of access are permitted, under what circumstances, and by whom.
- **Discretionary access control (DAC)**
  - Controls access **based on the identity of the requestor and on access rules**.
- **Mandatory access control (MAC)**
  - Controls access based on **comparing security labels** (which indicate how sensitive or critical system resources are) **with security clearances** (which indicate system entities are eligible to access certain resources).
- **Role-based access control (RBAC)**
  - Controls access based on the **roles that users have within the system** and on rules stating what accesses are allowed to users in given roles.
- **Attribute-based access control (ABAC)**
  - Controls access based on **attributes of the user, the resource to be accessed, and current environmental conditions.**

Basic Elements of Access Control

- **Subject**
  - **Entity capable of accessing objects**
  - Three classes of subject
    - **Owner:** Creator of a resource – Administrator
    - **Group:** Collection of members who are assigned with access rights.
    - **World:** The least amount of access is granted to users who are able to access the system but are not included in the categories owner and group.
- **Object**
  - **Resource to which access is controlled.**
- **Access Right**
  - **The way in which a subject may access an object.**
  - **Read; Write; Execute; Delete; Create; Search**

Discretionary Access Control

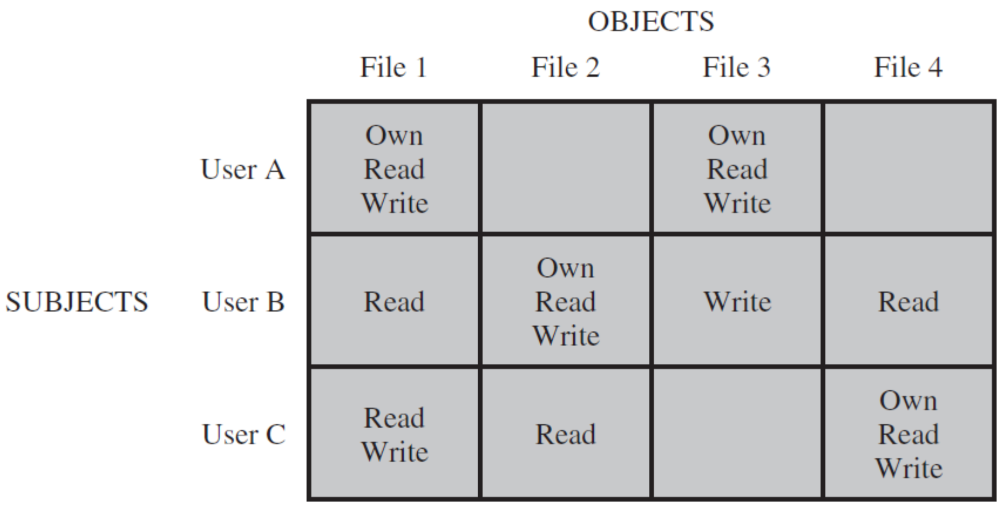
- Controls access **based on the identity of the requestor** and on access rules.
- **General approach to DAC - Access Matrix**
    - 1st Dimension - Identified subjects that may attempt data access to the resources
    - 2nd Dimension - The objects that may be accessed



		OBJECTS			
		File 1	File 2	File 3	File 4
SUBJECTS	User A	Own Read Write		Own Read Write	
	User B	Read	Own Read Write	Write	Read
	User C	Read Write	Read		Own Read Write

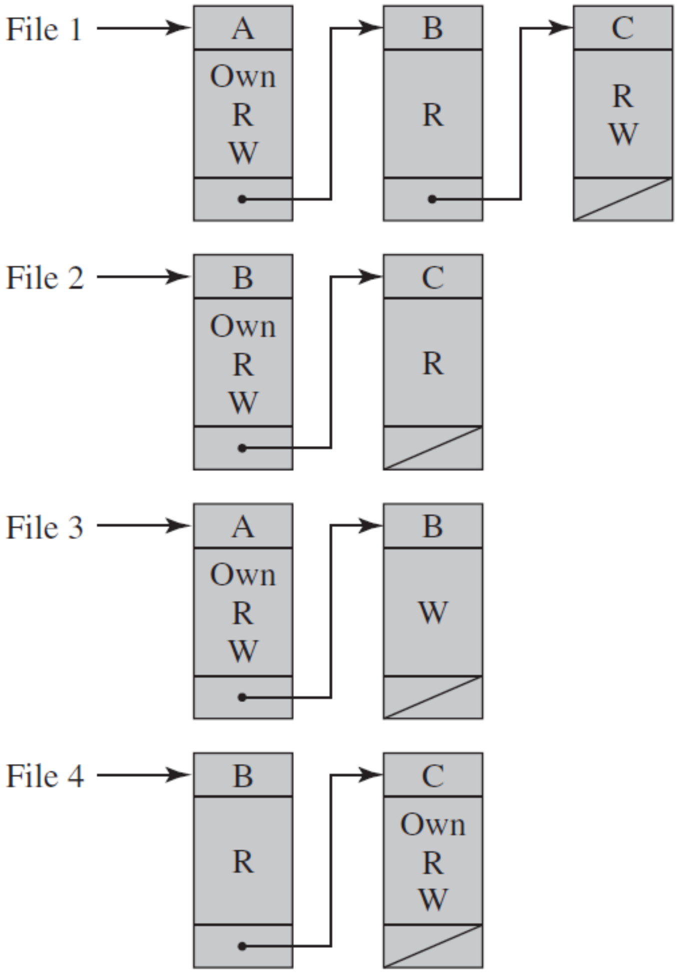
(a) Access matrix

Decomposing Access Matrix (by Column)



		OBJECTS			
		File 1	File 2	File 3	File 4
SUBJECTS	User A	Own Read Write		Own Read Write	
	User B	Read	Own Read Write	Write	Read
	User C	Read Write	Read		Own Read Write

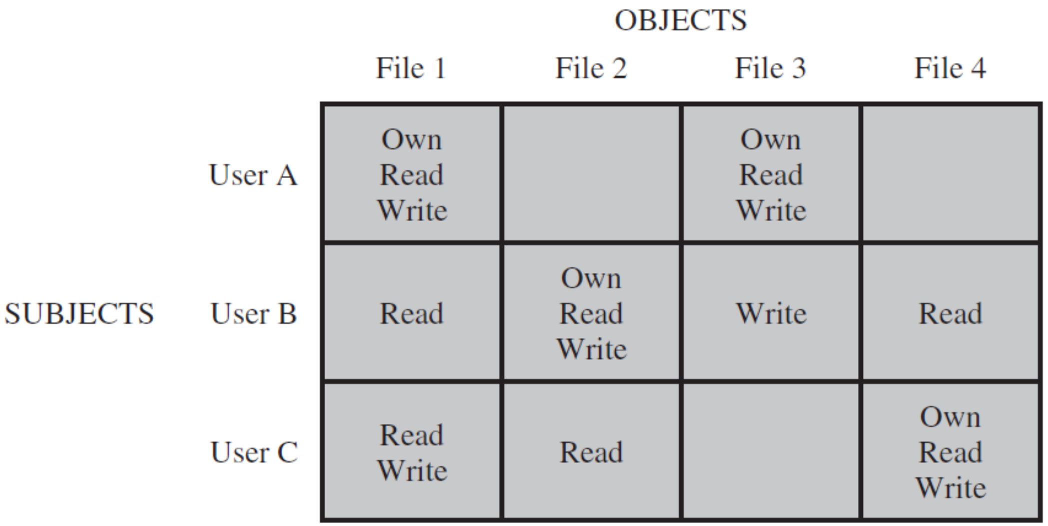
(a) Access matrix



(b) Access control lists for files of part (a)

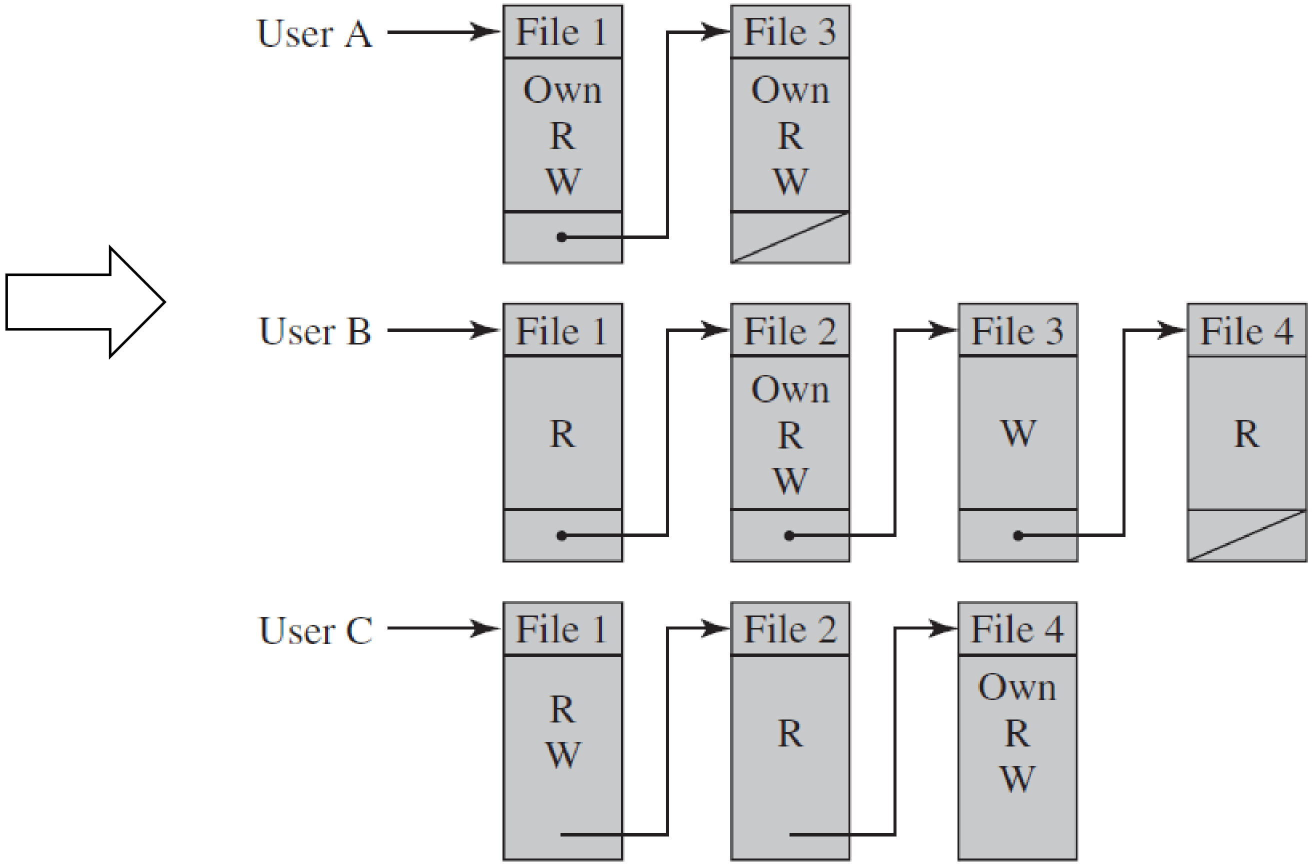
- For each object, an ACL lists users and their permitted access rights.
- The ACL may contain a default, or public, entry.
- The default set of rights should always follow the rule of least privilege or read-only access, whichever is applicable.
- ACLs are convenient, because each **ACL provides the information for a given resource.**
- However, this data structure is **not convenient for determining the access rights available to a specific user.**

Decomposing Access Matrix (by Row)



		OBJECTS			
		File 1	File 2	File 3	File 4
SUBJECTS	User A	Own Read Write		Own Read Write	
	User B	Read	Own Read Write	Write	Read
	User C	Read Write	Read		Own Read Write

(a) Access matrix



- A **capability ticket** specifies authorized objects and operations for a particular user.
  - **Integrity of a ticket must be protected.**
  - **Ticket must be unforgeable.**
- Each user has a number of tickets and may be authorized to loan or give them to others.
- Because tickets may be **dispersed around the system, they present a greater security problem** than access control lists.
- OS takes control of these tickets.

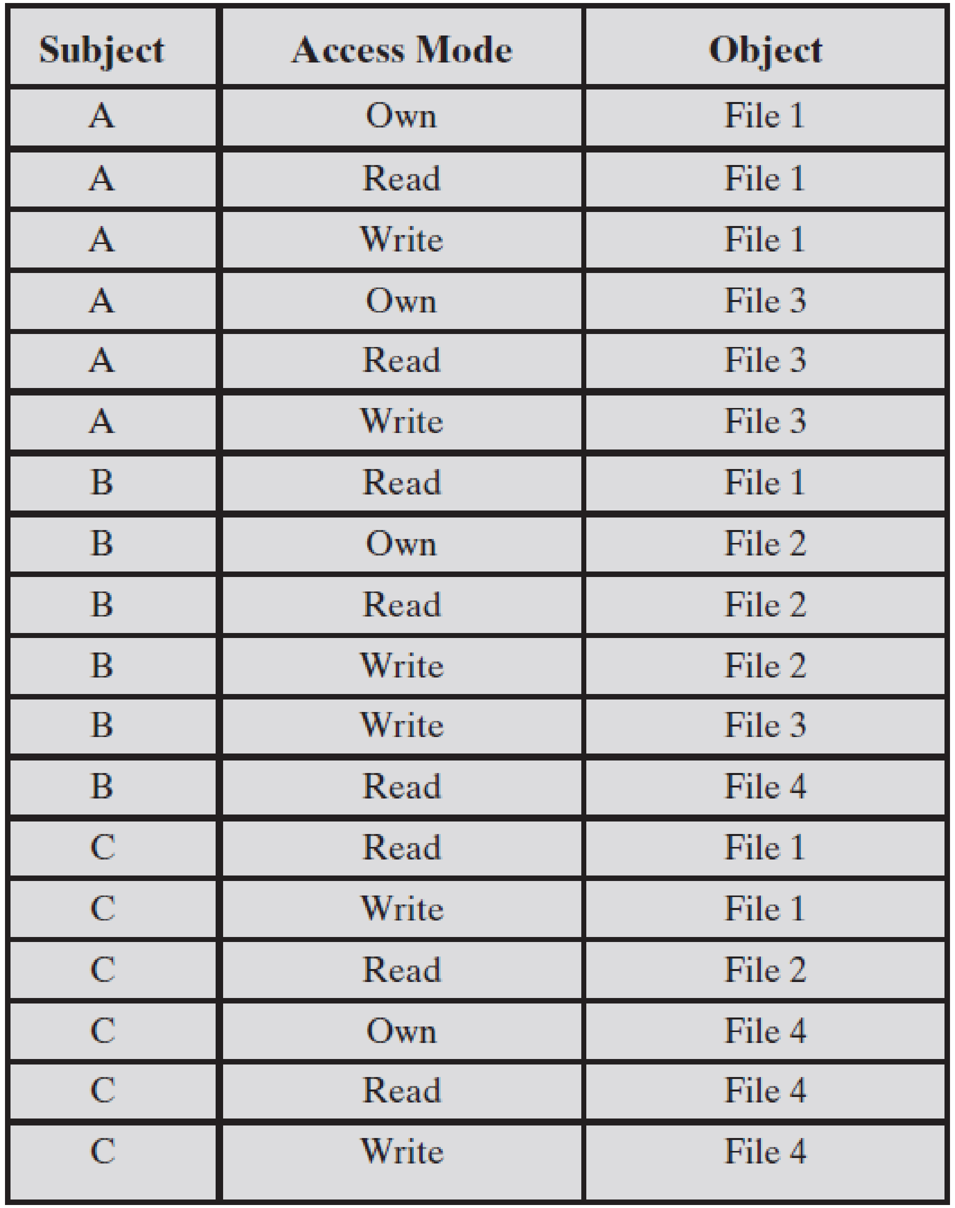
Token

This could be a **large random password, or a cryptographic message authentication code**.

This value is verified by the relevant resource whenever access is requested.

Authorization Table

- **Data structure that is not sparse**, like the access matrix, but is more convenient than either ACLs or capability lists.
- **One row for one access right of one subject to one resource.**
- **Sorting** or accessing the table by **subject** is equivalent to a **capability list.**
- **Sorting** or accessing the table by **object** is equivalent to an **ACL**.



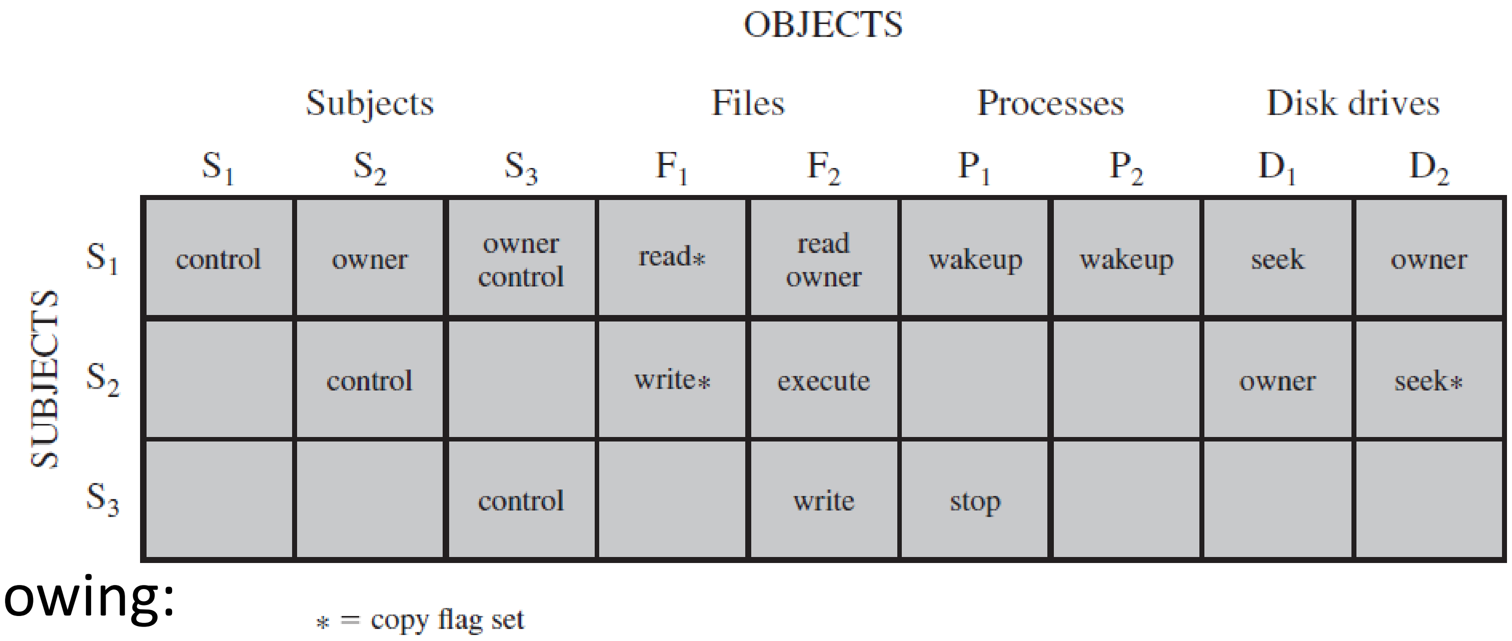
Subject	Access Mode	Object
A	Own	File 1
A	Read	File 1
A	Write	File 1
A	Own	File 3
A	Read	File 3
A	Write	File 3
B	Read	File 1
B	Own	File 2
B	Read	File 2
B	Write	File 2
B	Write	File 3
B	Read	File 4
C	Read	File 1
C	Write	File 1
C	Read	File 2
C	Own	File 4
C	Read	File 4
C	Write	File 4

Access Control Model - DAC

- The model **assumes a set of subjects, a set of objects, and a set of rules that govern the access of subjects to objects**.
- Let us define the **protection state of a system to be the set of information, at a given point in time**, that specifies the access rights for each subject with respect to each object.
- Three requirements
  - **Representing the protection state**
  - **Enforcing access rights**
  - **Allowing subjects to alter the protection state in certain ways.**

Representing the Protection State

- We extend the universe of objects in the access control matrix to include the following:
  - **Processes**
    - Access rights include the ability to **delete a process, stop (block), and wake up a process**.
  - **Devices**
    - Access rights include the ability to **read/write** the device, to control its operation (e.g., a **disk seek**), and to **block/unblock the device for use**.
  - **Memory locations or regions**
    - Access rights include the ability to **read/write certain regions of memory** that are protected such that the default is to disallow access.
  - **Subjects**
    - Access rights with respect to a subject have to do with the **ability to grant or delete access rights of that subject to other objects**

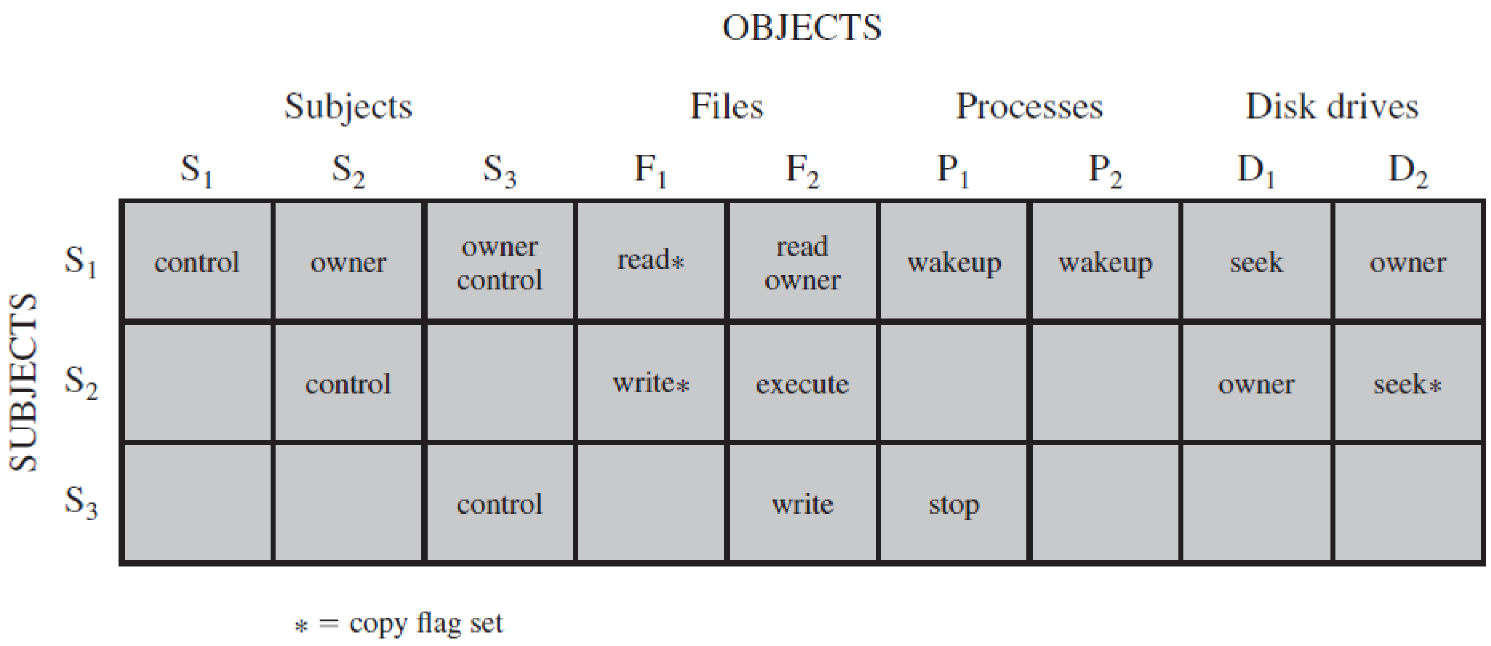


OBJECTS

		Subjects			Files		Processes		Disk drives	
		S_1	S_2	S_3	F_1	F_2	P_1	P_2	D_1	D_2
SUBJECTS	S_1	control	owner	owner control	read*	read owner	wakeup	wakeup	seek	owner
	S_2		control		write*	execute			owner	seek*
	S_3			control		write	stop			

* = copy flag set

- For an access control matrix A, each **entry A[S, X]** contains strings, called **access attributes**, that specify the access rights of subject S to object X.
  - S1 may read file F1, because ‘read’ appears in A[S1, F1].



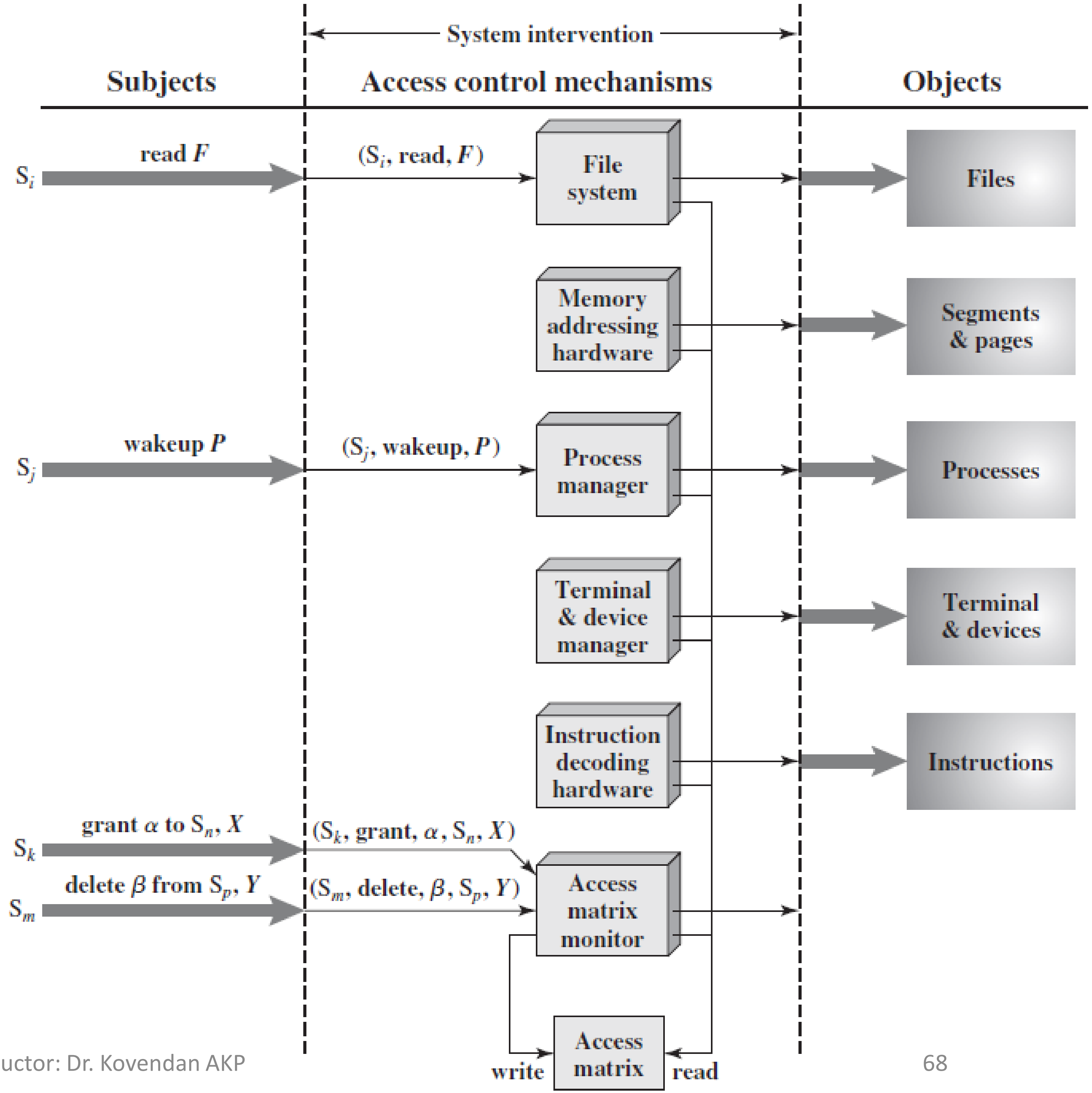
		OBJECTS								
		Subjects			Files		Processes		Disk drives	
		S_1	S_2	S_3	F_1	F_2	P_1	P_2	D_1	D_2
SUBJECTS	S_1	control	owner	owner control	read*	read owner	wakeup	wakeup	seek	owner
	S_2		control		write*	execute			owner	seek*
	S_3			control		write	stop			

* = copy flag set

- Separate access control module is associated with each type of object.
- Every access by a subject to an object is **mediated by the controller for that object**, and that the controller’s decision is based on the current contents of the matrix.
- An access attempt triggers the following steps:

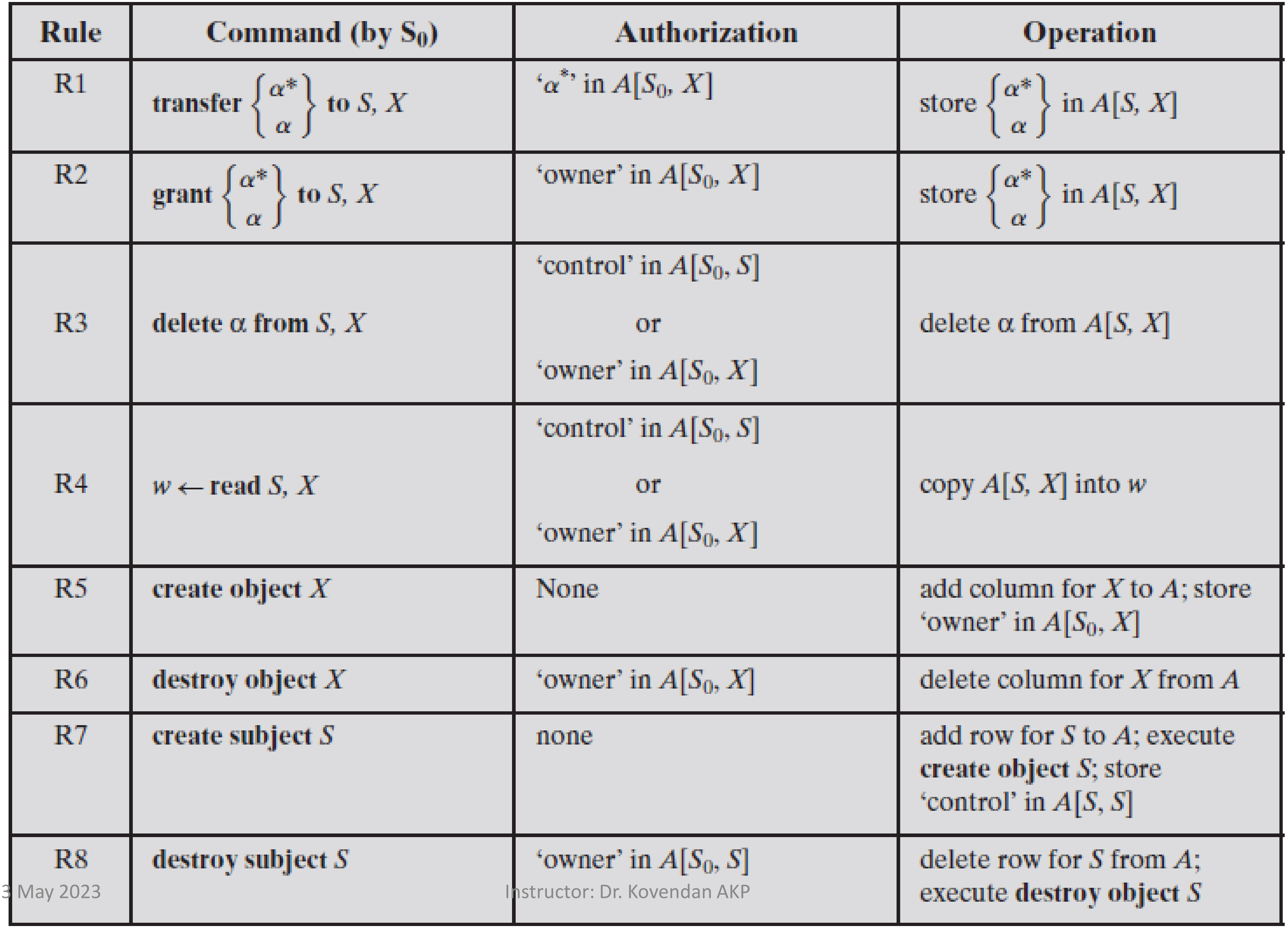
1. **A subject S0 issues a request of type α for object X.**
2. **The request causes the system to generate a message of the form (S0, α, X) to the controller for X.**
3. **The controller interrogates the access matrix A to determine if α is in A[S0, X].**
    1. **If so, the access is allowed;**
    2. **if not, the access is denied and a protection violation occurs.**

- The violation should trigger a warning and appropriate action.



Modification of Access Matrix by a Subject

- In addition, certain subjects have the authority to **make specific changes to the access matrix.**
- A request to modify the access matrix is **treated as an access to the matrix, with the individual entries in the matrix treated as objects.**
- Such accesses are mediated by an **access matrix controller**, which controls updates to the matrix.
- The model also includes a **set of rules that govern modifications** to the access matrix.
- For this purpose, we introduce the access rights **‘owner’ and ‘control’ and the concept of a copy flag**.
- Rule table is illustrated in the next slide.



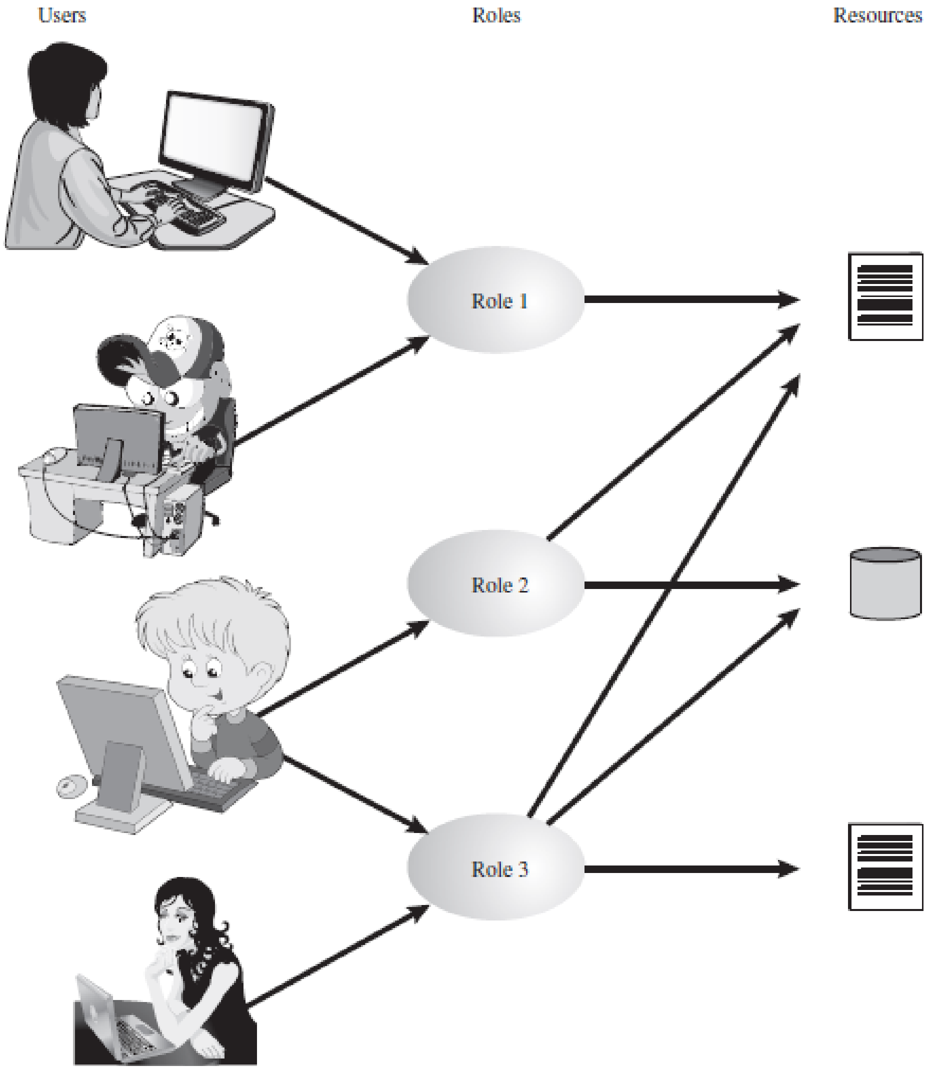
Rule	Command (by S_0)	Authorization	Operation
R1	**transfer** $\left\{\begin{matrix}\alpha^*\\ \alpha\end{matrix}\right\}$ **to** S, X	'α^*' in $A[S_0, X]$	store $\left\{\begin{matrix}\alpha^*\\ \alpha\end{matrix}\right\}$ in $A[S, X]$
R2	**grant** $\left\{\begin{matrix}\alpha^*\\ \alpha\end{matrix}\right\}$ **to** S, X	'owner' in $A[S_0, X]$	store $\left\{\begin{matrix}\alpha^*\\ \alpha\end{matrix}\right\}$ in $A[S, X]$
R3	**delete** α **from** S, X	'control' in $A[S_0, S]$ or 'owner' in $A[S_0, X]$	delete α from $A[S, X]$
R4	$w \leftarrow$ **read** S, X	'control' in $A[S_0, S]$ or 'owner' in $A[S_0, X]$	copy $A[S, X]$ into w
R5	**create object** X	None	add column for X to A; store 'owner' in $A[S_0, X]$
R6	**destroy object** X	'owner' in $A[S_0, X]$	delete column for X from A
R7	**create subject** S	none	add row for S to A; execute **create object** S; store 'control' in $A[S, S]$
R8	**destroy subject** S	'owner' in $A[S_0, S]$	delete row for S from A; execute **destroy object** S

Protection Domains

- The access control matrix model that we have discussed so far associates a set of capabilities with a user.
- A more **general and more flexible approach, proposed is to associate capabilities with protection domains**.
- A protection domain is a set of objects together with access rights to those objects.
- In terms of the access matrix, **a row defines a protection domain**.
- Any processes spawned by the user have access rights defined by the same protection domain.
- Eg: In Unix,
  - User Mode
  - Kernel Mode

Role based Access Control

- RBAC is based on the roles that users assume in a system rather than the user's identity.
- In turn, **users are assigned to different roles, either statically or dynamically**, according to their responsibilities.
- **Each role will have specific access rights to one or more resources.**
- **Access matrix representation** to depict the key elements of an RBAC system



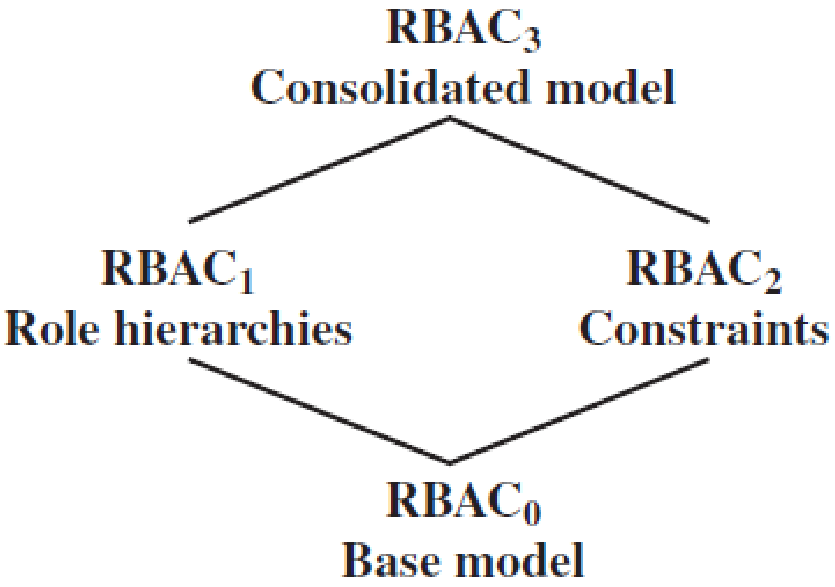
Access Matrix Representation

	R_1	R_2	• • •	R_n
U_1	✕			
U_2	✕			
U_3		✕		✕
U_4				✕
U_5				✕
U_6				✕
• • •				
U_m	✕			

		OBJECTS								
		R_1	R_2	R_n	F_1	F_2	P_1	P_2	D_1	D_2
ROLES	R_1	control	owner	owner control	read *	read owner	wakeup	wakeup	seek	owner
	R_2		control		write *	execute			owner	seek *
	• • •									
	R_n			control		write	stop			

RBAC Reference Models

- This family consists of four models that are related to each other.
- **RBAC0 contains the minimum functionality for an RBAC system.**
- **RBAC1 includes the RBAC0** functionality and **adds role hierarchies**, which enable one role to inherit permissions from another role.
- **RBAC2 includes RBAC0** and **adds constraints**, which restrict the ways in which the components of a RBAC system may be configured.
- **RBAC3 contains the functionality of RBAC0, RBAC1, and RBAC2**.



RBAC$_3$
Consolidated model

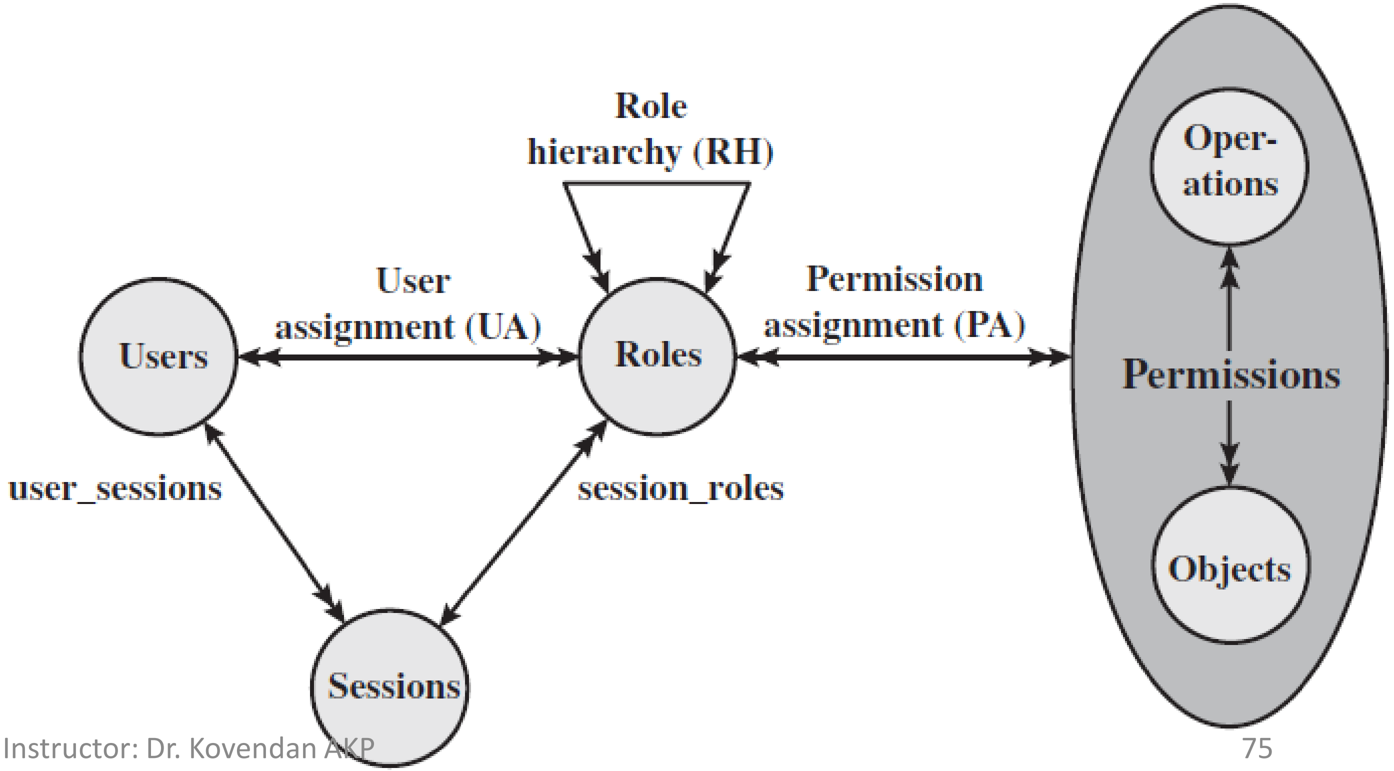
RBAC$_1$
Role hierarchies

RBAC$_2$
Constraints

RBAC$_0$
Base model

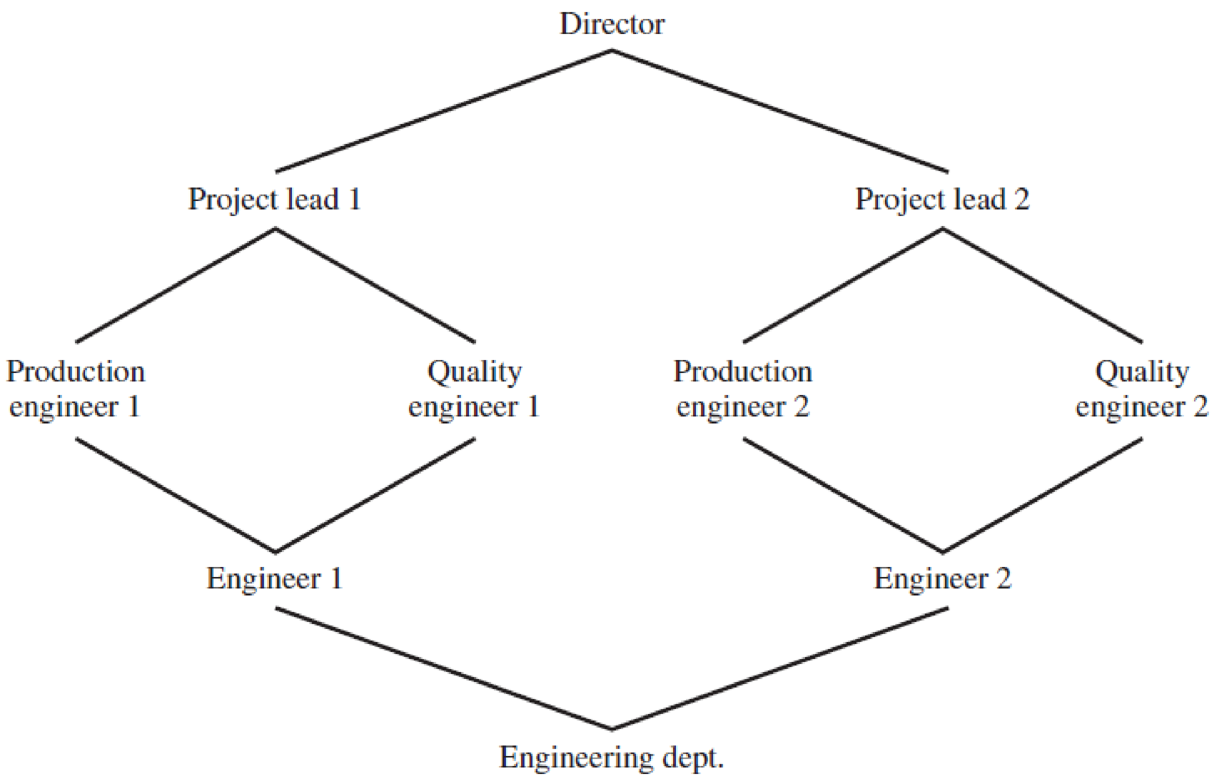
RBAC0

- Without the role hierarchy and constraints
- Four types of entities in an RBAC0 system
  - User
  - Role
  - Permission
  - Session



RBAC1 - Hierarchies

- Role hierarchies provide a means of **reflecting the hierarchical structure of roles in an organization.**
- Job functions with **greater responsibility have greater authority to access resources**.
- A **subordinate job function may have a subset of the access rights** of the superior job function.
- A line between two roles implies that the upper role includes all of the access rights of the lower role, as well as other access rights not available to the lower role.

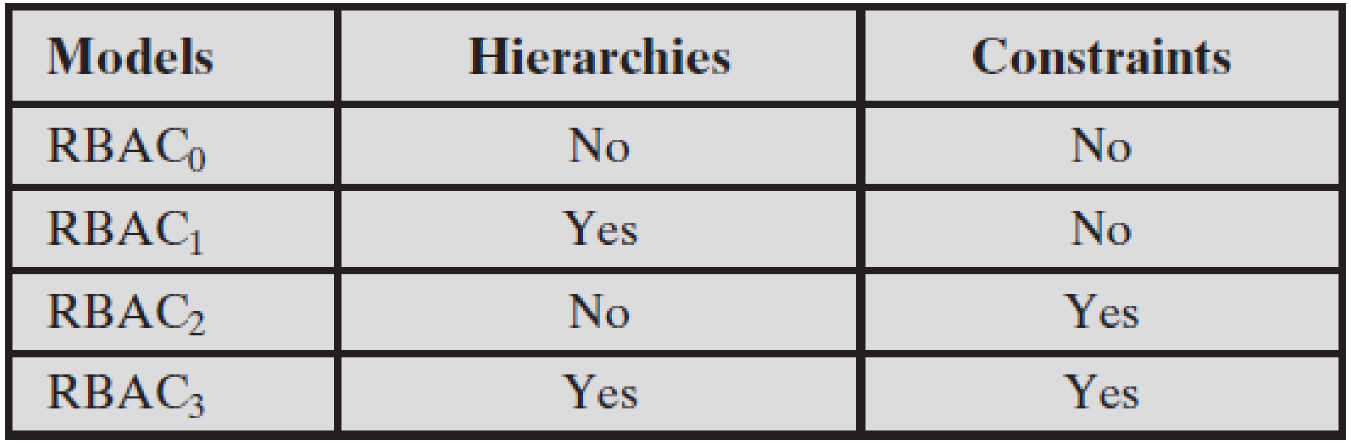


RBAC2 - Constraints

- Adapting RBAC to the specifics of administrative and security policies in an organization.
- A constraint is a defined relationship among roles or a condition related to roles.
- Constraints:
  - **Mutually exclusive roles**
    - Roles such that a user can be assigned to only one role in the set
    - A user can only be assigned to one role in the set (either during a session or statically).
    - Any permission (access right) can be granted to only one role in the set.
  - **Cardinality**
    - Setting a maximum number with respect to roles
  - **Prerequisite roles**
    - Dictates that a user can only be assigned to a particular role if it is already assigned to some other specified role.

RBAC3 – Hierarchy + Constraints

- Combination of RBAC0 + RBAC1 + RBAC2



Models	Hierarchies	Constraints
RBAC$_0$	No	No
RBAC$_1$	Yes	No
RBAC$_2$	No	Yes
RBAC$_3$	Yes	Yes

Attribute based Access Control

- An ABAC model can define authorizations that express conditions on properties of both the resource and the subject.
  - For example,
  - **Consider a configuration in which each resource has an attribute that identifies the subject that created the resource.**
  - **Then, a single access rule can specify the ownership privilege for all the creators of every resource.**
- Three key elements to an ABAC model
  - **Attributes**
    - Which are defined for entities in a configuration
  - **Policy model**
  - **Architecture model**
    - Enforce access control

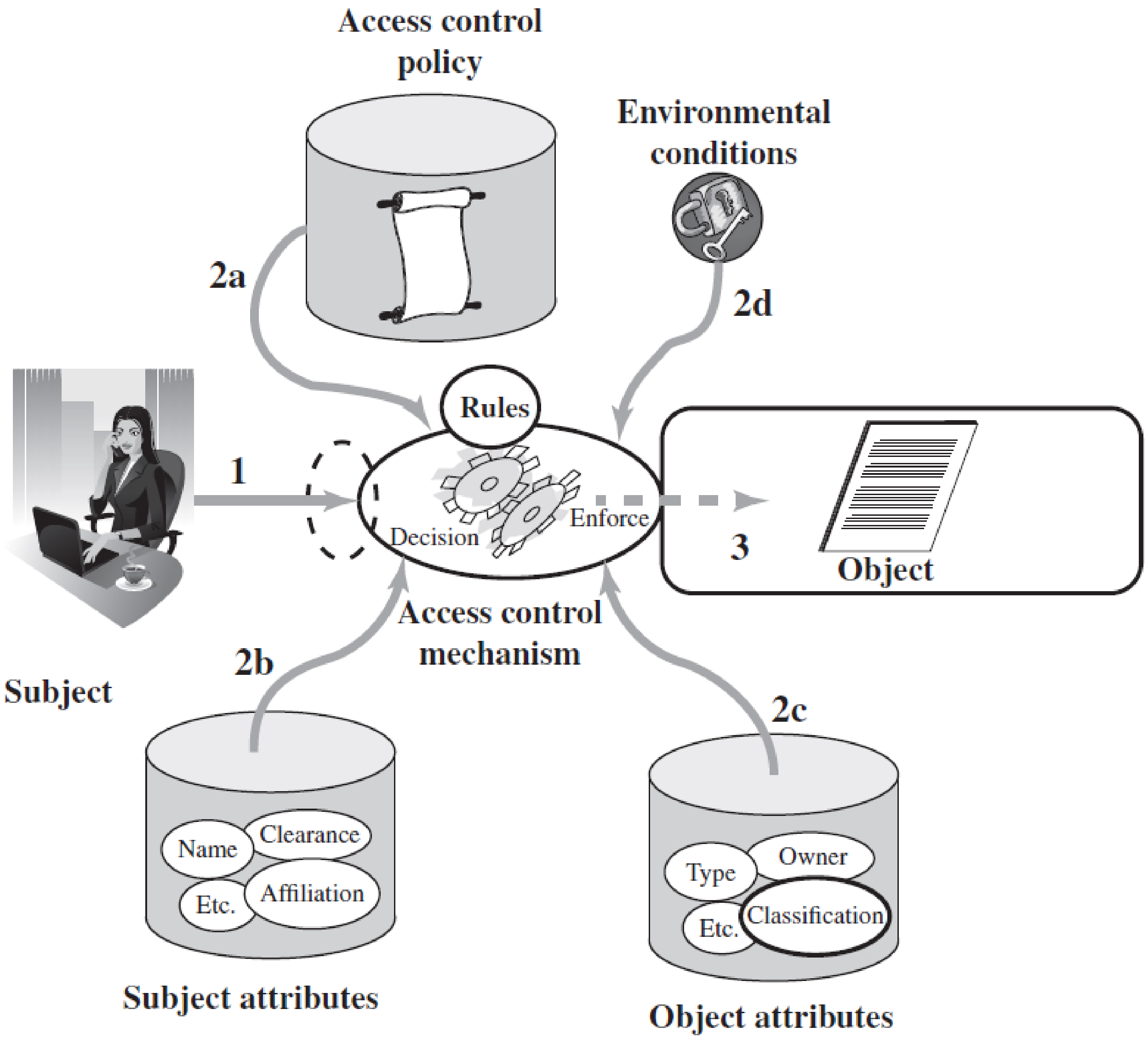
Attributes

- Attributes are characteristics that define specific aspects of the subject, object, environment conditions, and/or requested operations that are predefined and preassigned by an authority.
  - **Subject Attributes**
    - Define the identity and characteristics of the subject.
    - Eg: Identifier, name, organization, job title
  - **Object Attributes**
    - Objects have attributes that can be leveraged to make access control decisions.
    - A Microsoft Word document, for example, may have attributes such as title, subject, date, and author.
  - **Environmental Attributes**
    - The operational, technical, and even situational environment or context in which the information access occurs.
    - Current date and time, the current virus/hacker activities, and the network's security level.
- ABAC relies upon the evaluation of attributes of the subject, attributes of the object, and a formal relationship or access control rule defining the allowable operations for subject object attribute combinations in a given environment.
- ABAC systems are capable of enforcing DAC, RBAC, and MAC concepts.

ABAC Logical Architecture

- An access by a subject to an object proceeds according to the following steps:

1. A subject requests access to an object. This request is routed to an access control mechanism.
2. The access control mechanism is governed by a set of rules (2a) that are defined by a preconfigured access control policy.
3. Based on these rules, the access control mechanism assesses the attributes of the subject (2b), object (2c), and current environmental conditions (2d) to determine authorization.
4. The access control mechanism grants the subject access to the object if access is authorized and denies access if it is not authorized.



ABAC Policies

- A policy is a set of rules and relationships that govern allowable behavior within an organization, based on the privileges of subjects and how resources or objects are to be protected under which environment conditions.
- Privileges represent the authorized behavior of a subject.
- Other terms that are commonly used instead of privileges are rights, authorizations, and entitlements.
- **Policy is typically written from the perspective of the object that needs protecting and the privileges available to subjects.**

Firewall

- A firewall is a **network security device that monitors the network's data traffic.**
- It **permits or blocks data packets based on a set of security rules**.
- Firewalls can be **software, hardware, or cloud-based**, with each type of firewall having its own unique pros and cons.
- The primary goal of a firewall is to **block malicious traffic requests and data packets while allowing legitimate traffic through.**

The Need for Firewalls

- The modern organization includes the following,
  - **Centralized data processing system**, with a central mainframe supporting a number of directly connected terminals.
  - **Local area networks (LANs)** interconnecting PCs and terminals to each other and the mainframe.
  - Premises network, **consisting of a number of LANs, interconnecting PCs**, servers, and perhaps a mainframe or two.
  - Enterprise-wide network, consisting of **multiple, geographically distributed premises networks interconnected by a private wide area network (WAN).**
  - **Internet connectivity**, in which the various premises networks all hook into the Internet and may or may not also be connected by a private WAN.
- **This creates a threat to the organization.**

Firewall Design Goals

- **All traffic from inside to outside, and vice versa, must pass through the firewall.** This is achieved by physically blocking all access to the local network except via the firewall.
- **Only authorized traffic, as defined by the local security policy**, will be allowed to pass.
- **The firewall itself is immune to penetration**. This implies the use of a hardened system with a secured operating system.
- Trusted computer systems are suitable for hosting a firewall and often required in government applications.

Firewall Characteristics

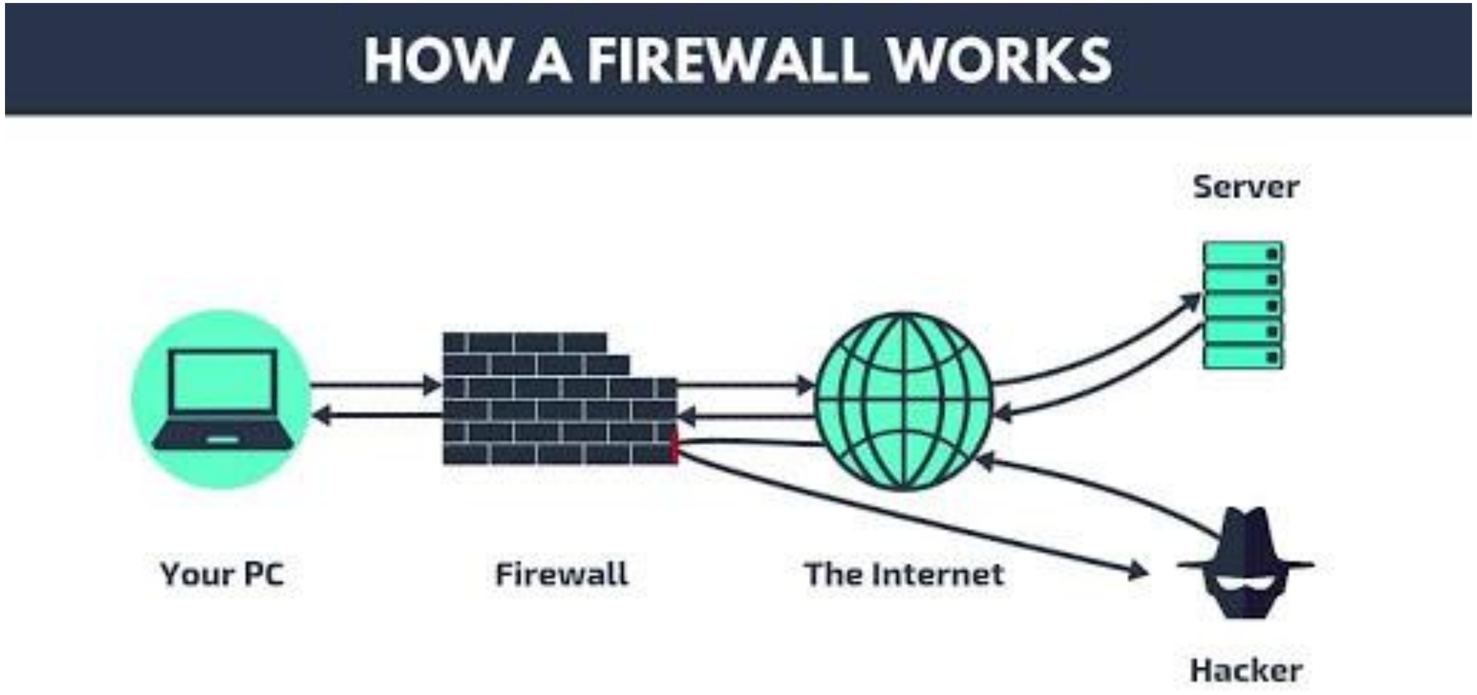
- Controls access based on the **source or destination addresses** and port numbers, direction of flow being inbound or outbound, and other network and transport layer characteristics.
- Controls access on the basis of **authorized application protocol data**.
- Controls access based on the **users identity**.
- Controls access based on considerations such as the **time or request**.

Capabilities of a Firewall

- A firewall defines a **single choke point that attempts to keep unauthorized users out** of the protected network.
- A firewall provides a **location for monitoring security-related events**.
- **Audits and alarms** can be implemented on the firewall system.
- A firewall can serve as the **platform for IPSec**.

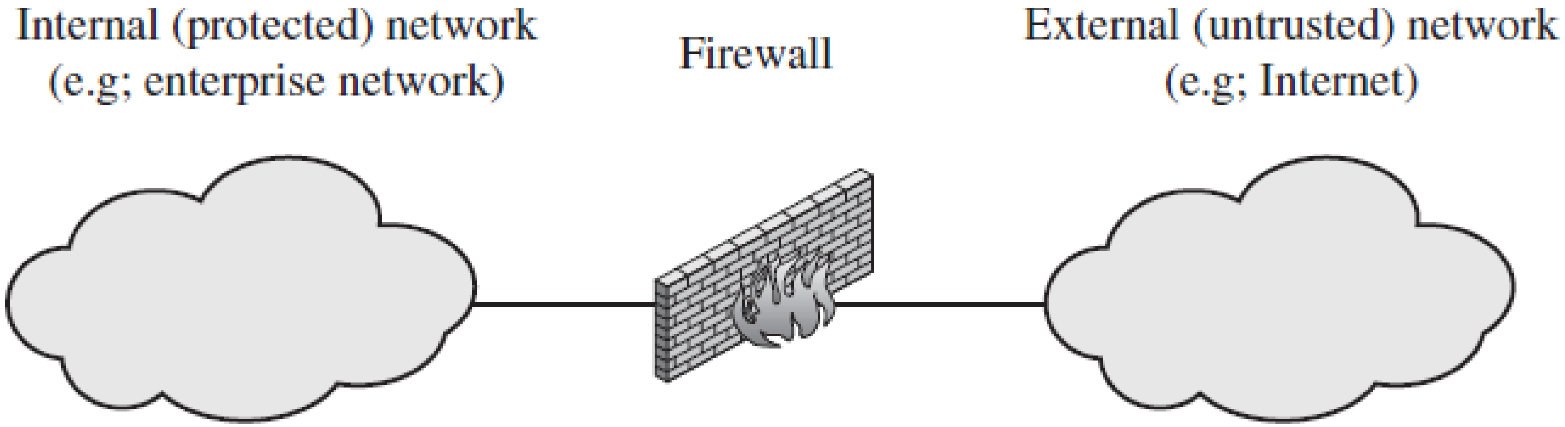
How does a firewall work?

- Firewalls carefully **analyze incoming traffic based on pre-established rules** and filter traffic coming from suspicious sources to prevent attacks.
- **Firewalls guard traffic at a computer's entry point, called ports**, which is where information is exchanged with external devices.



Types of Firewall

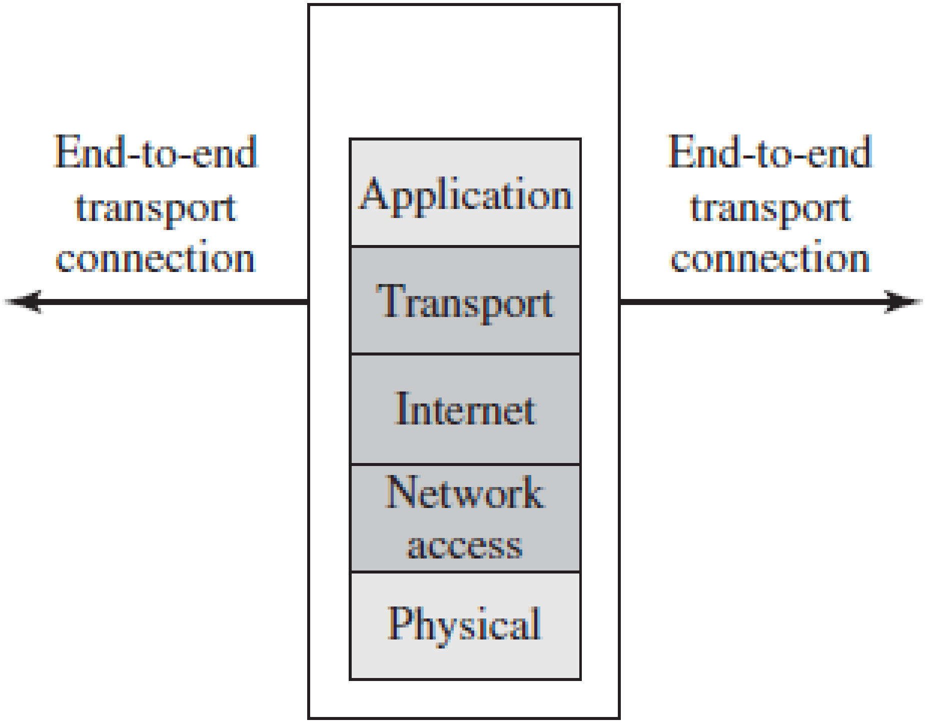
- It can operate as a **positive filter**, allowing to pass only packets that meet specific criteria, or as a **negative filter**, rejecting any packet that meets certain criteria.
- Types of Firewall
  - Packet-filtering firewalls
  - Circuit-level firewalls/gateways
  - Stateful inspection firewalls
  - Application-level gateways (a.k.a. proxy firewalls)
  - Next-gen firewalls
  - Software firewalls
  - Hardware firewalls
  - Cloud firewalls


Internal (protected) network
(e.g; enterprise network)
Firewall
External (untrusted) network
(e.g; Internet)

(a) General model

Packet filtering firewalls

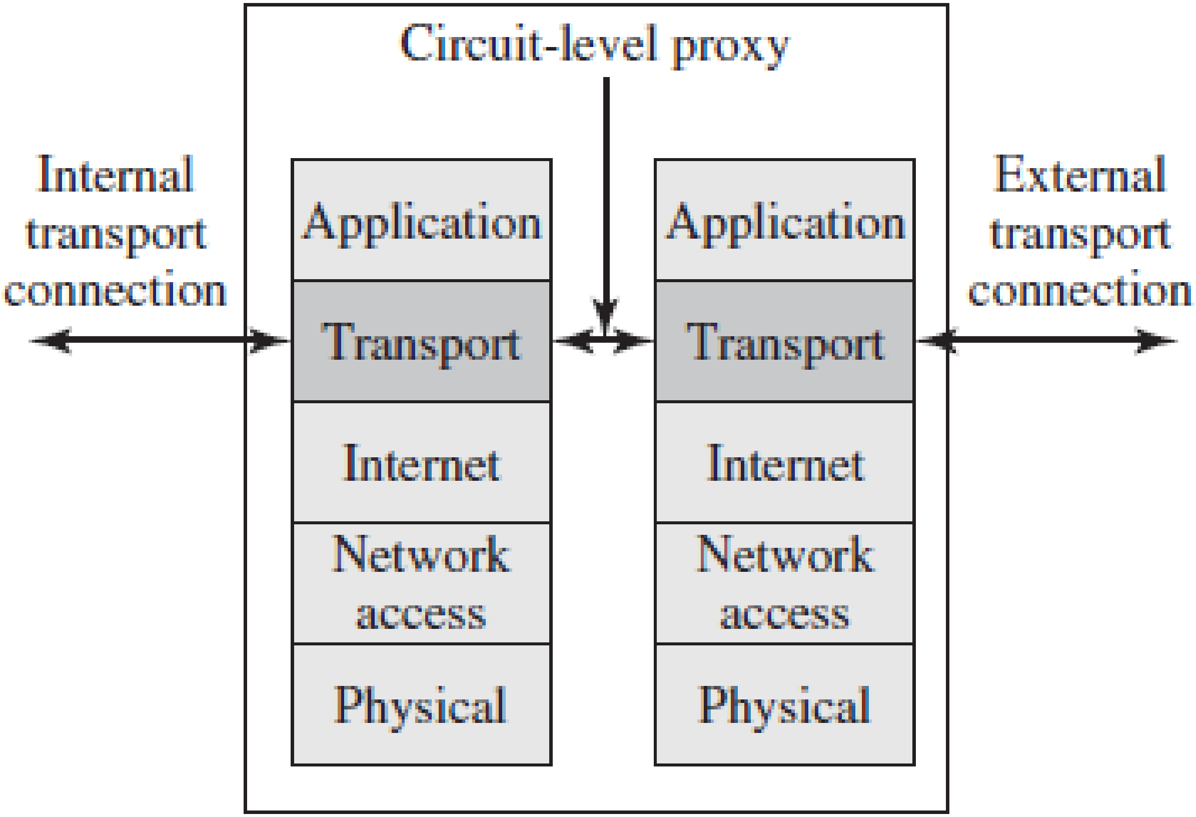
- It is the most “basic” and oldest type of firewall architecture.
- It basically creates a **checkpoint at a traffic router or switch**.
- The firewall performs a simple check of the data packets coming through the router
- inspecting information such as the destination and origination **IP address, packet type, port number, and other surface-level information.**
- **These firewalls aren’t very resource-intensive thereby, impact on system performance and are relatively simple.**
- **However, they’re also relatively easy to bypass compared to firewalls with more robust inspection capabilities.**


End-to-end transport connection
Application
Transport
Internet
Network access
Physical
End-to-end transport connection

(b) Packet filtering firewall

Circuit-Level Gateways

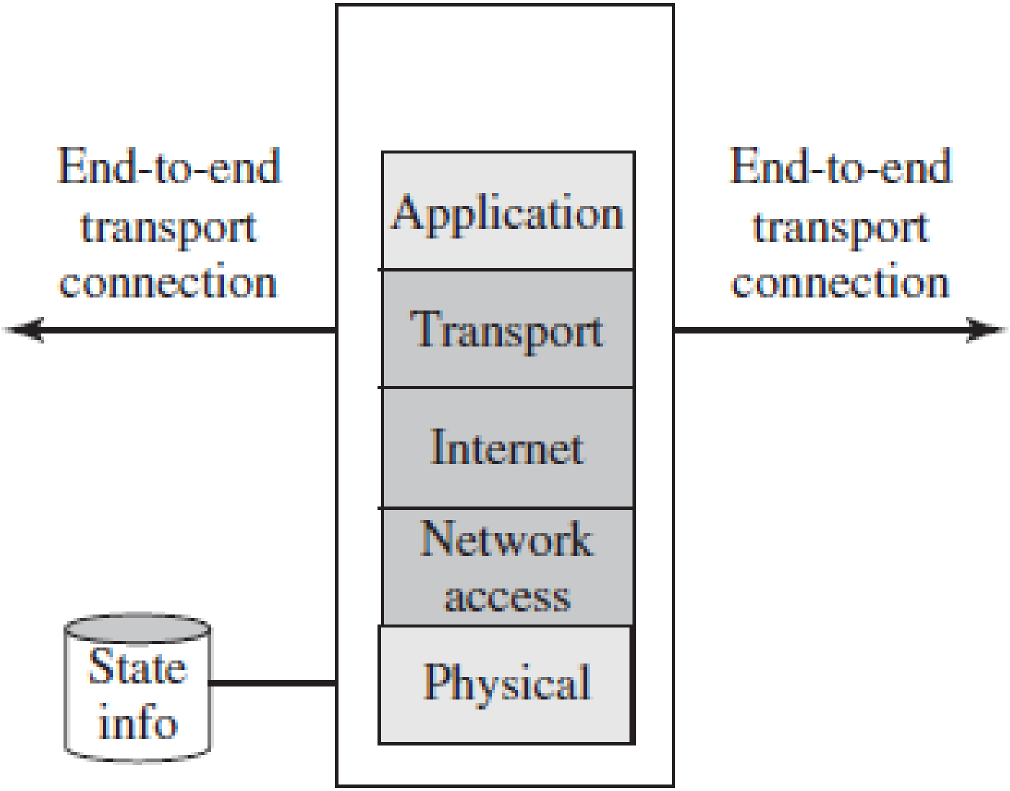
- Another simplistic firewall which works quickly and easily approve or deny traffic without consuming significant computing resources.
- **Circuit-level gateways work by verifying the transmission control protocol (TCP) handshake.**
- This TCP handshake check is designed to make sure that the **session the packet is from is legitimate.**
- While extremely resource-efficient, these **firewalls do not check the packet itself.**
- **So, if a packet held malware, but had the right TCP handshake, it would pass right through.**
- This is why circuit-level gateways are not enough to protect your business by themselves.



(e) Circuit-level proxy firewall

Stateful Inspection Firewalls

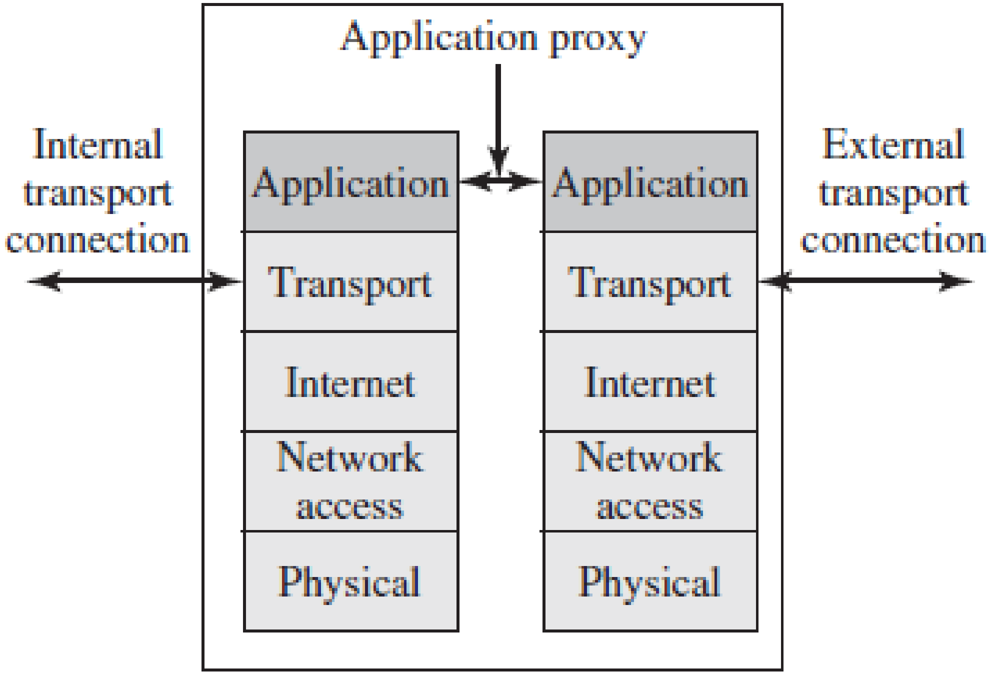
- State-aware devices, on the other hand, not only examine each packet, but also **keep track of whether or not that packet is part of an established TCP or other network session**.
- These firewalls **combine both packet inspection technology and TCP handshake verification.**
- This **offers more security** than either packet filtering or circuit monitoring alone.
- However, these firewalls do put more of a strain on computing resources and network performance.


End-to-end transport connection
Application
Transport
Internet
Network access
Physical
End-to-end transport connection
State info

(c) Stateful inspection firewall

Proxy Firewalls (Application-Level Gateways)

- Proxy firewalls operate at the **application layer to filter incoming traffic**.
- These firewalls are delivered via a cloud-based solution or another proxy device.
- Rather than letting traffic connect directly, **the proxy firewall first establishes a connection to the source of the traffic and inspects the incoming data packet.**
- Also perform **deep-layer packet inspections**, checking the actual contents of the information packet to **verify that it contains no malware**.
- Once the check is complete, and the packet is approved to connect to the destination, the proxy sends it off.



(d) Application proxy firewall

Next-Generation Firewalls

- Many of the most recently-released firewall products are being touted as “next-generation” architectures.
- Some common features of next-generation firewall architectures include **deep-packet inspection, TCP handshake checks, and surface-level packet inspection.**
- Next-generation firewalls may include other technologies as well, such as **Intrusion Prevention Systems (IPSs)** that work to automatically stop attacks against your network.
- The issue is that there is no one definition of a next-generation firewall

Software Firewalls

- Software firewalls include any type of **firewall that is installed on a local device rather than a separate piece of hardware**
- The big benefit of a software firewall is that it's highly useful for **creating defense in depth by isolating individual network endpoints from one another.**
- However, maintaining individual software firewalls on different devices can be difficult and time-consuming.
- Furthermore, **not every device on a network may be compatible with a single software firewall.**

Hardware Firewalls

- Hardware firewalls use a **physical appliance that acts in a manner similar to a traffic router**.
- Act as a perimeter security by making sure **malicious traffic is intercepted before the company's network endpoints**.
- The major weakness of a hardware-based firewall, however, is that it is often **easy for insider attacks to bypass them**.

Cloud Firewalls

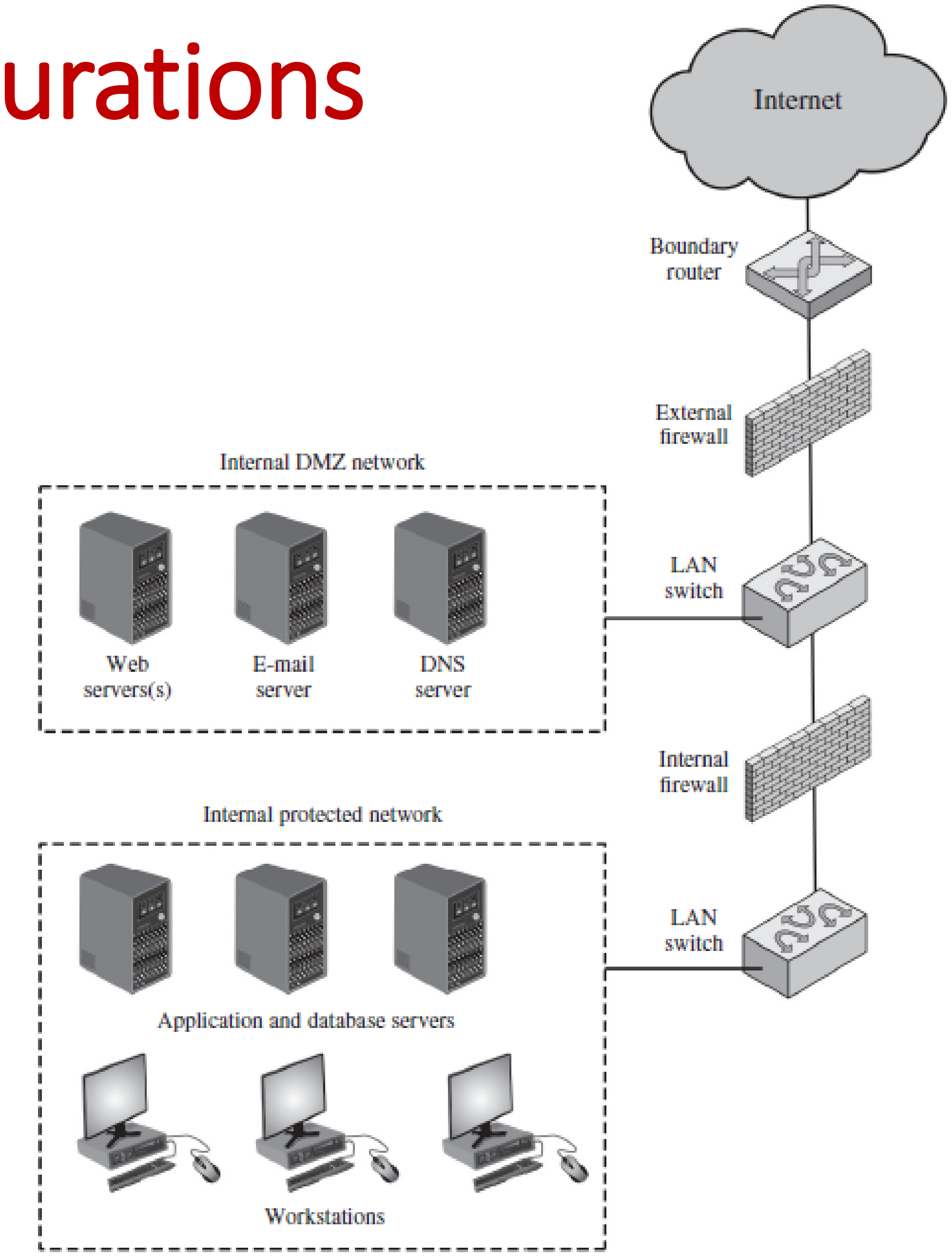
- Whenever a cloud solution is used to deliver a firewall, it can be called a cloud firewall, or **firewall-as-a-service** (FaaS).
- Cloud firewalls are considered synonymous with proxy firewalls by many, since a cloud server is often used in a proxy firewall setup.
- **The big benefit of having cloud-based firewalls is that they are very easy to scale with your organization.**
- As your needs grow, you can add additional capacity to the cloud server to filter larger traffic loads.
- Cloud firewalls, like hardware firewalls, excel at **perimeter security**.

Firewall Basing

- It is common to **base a firewall on a stand-alone machine running a common operating system, such as UNIX or Linux.**
- Other Basing points
  - **Bastion Host**
    - A bastion host is a **system identified by the firewall administrator as a critical strong point** in the network's security.
  - **Host-Based Firewalls**
    - A host-based firewall is a **software module used to secure an individual host**.
    - Such modules are available in many operating systems or can be provided as an add- on package.
  - **Personal Firewall**
    - The personal firewall is a **software module on the personal computer**.

Firewall Location and Configurations
1. DMZ Networks

- DMZ (demilitarized zone) network
  - Recourses that are accessed externally but need additional protection
- External firewall provides a basic level of protection.
- Internal firewalls serve three purposes,
  - The internal firewall adds more stringent filtering capability.
  - Two-way protection with respect to the DMZ
  - Multiple internal firewalls can be used to protect portions of the internal network from each other.



Firewall Location and Configurations
2. Virtual Private Networks

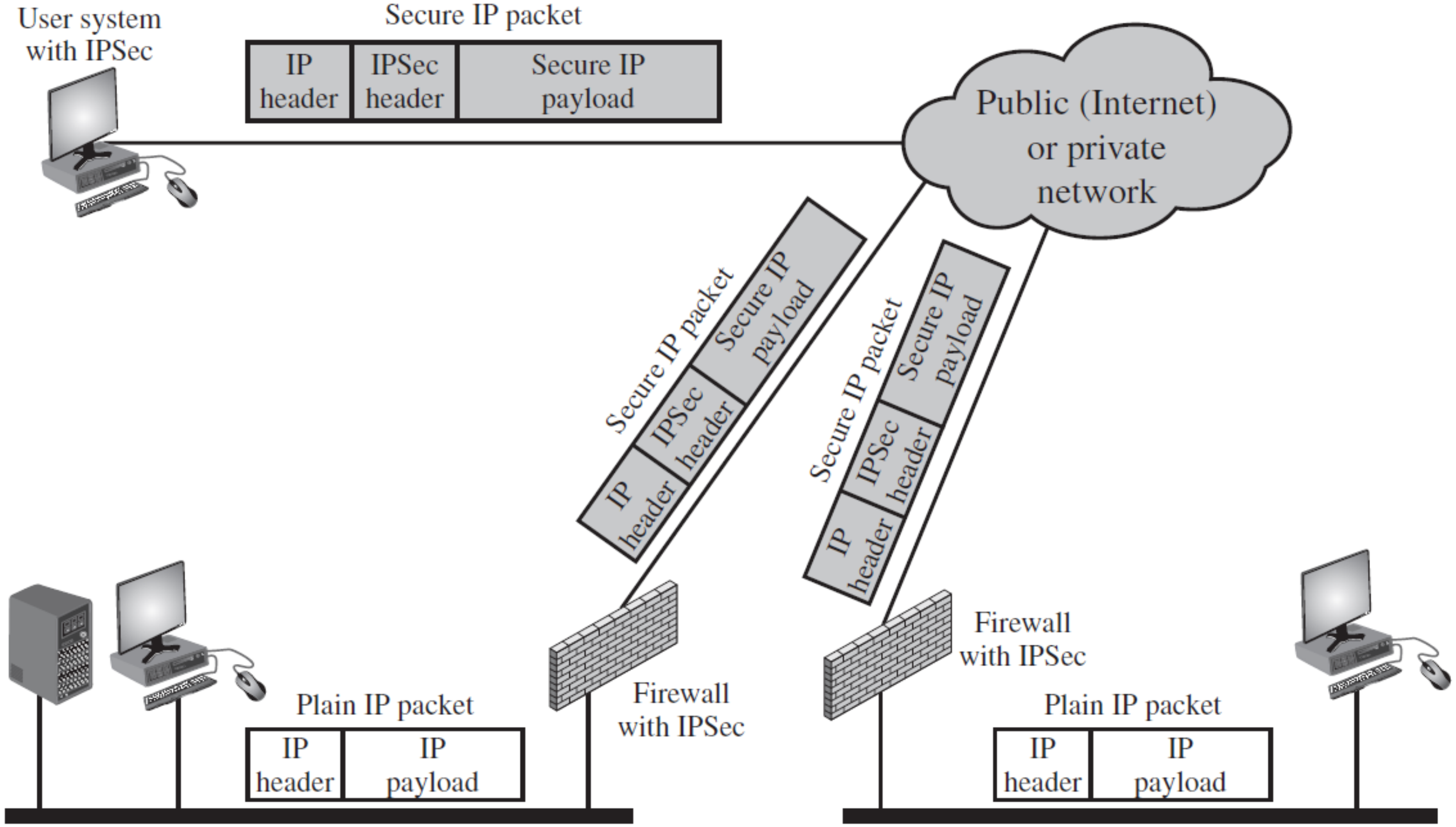


Figure 9.3 A VPN Security Scenario

Firewall Location and Configurations
3. Distributed Firewalls

- A distributed firewall configuration involves stand-alone firewall devices plus host based firewalls working together under a central administrative control.

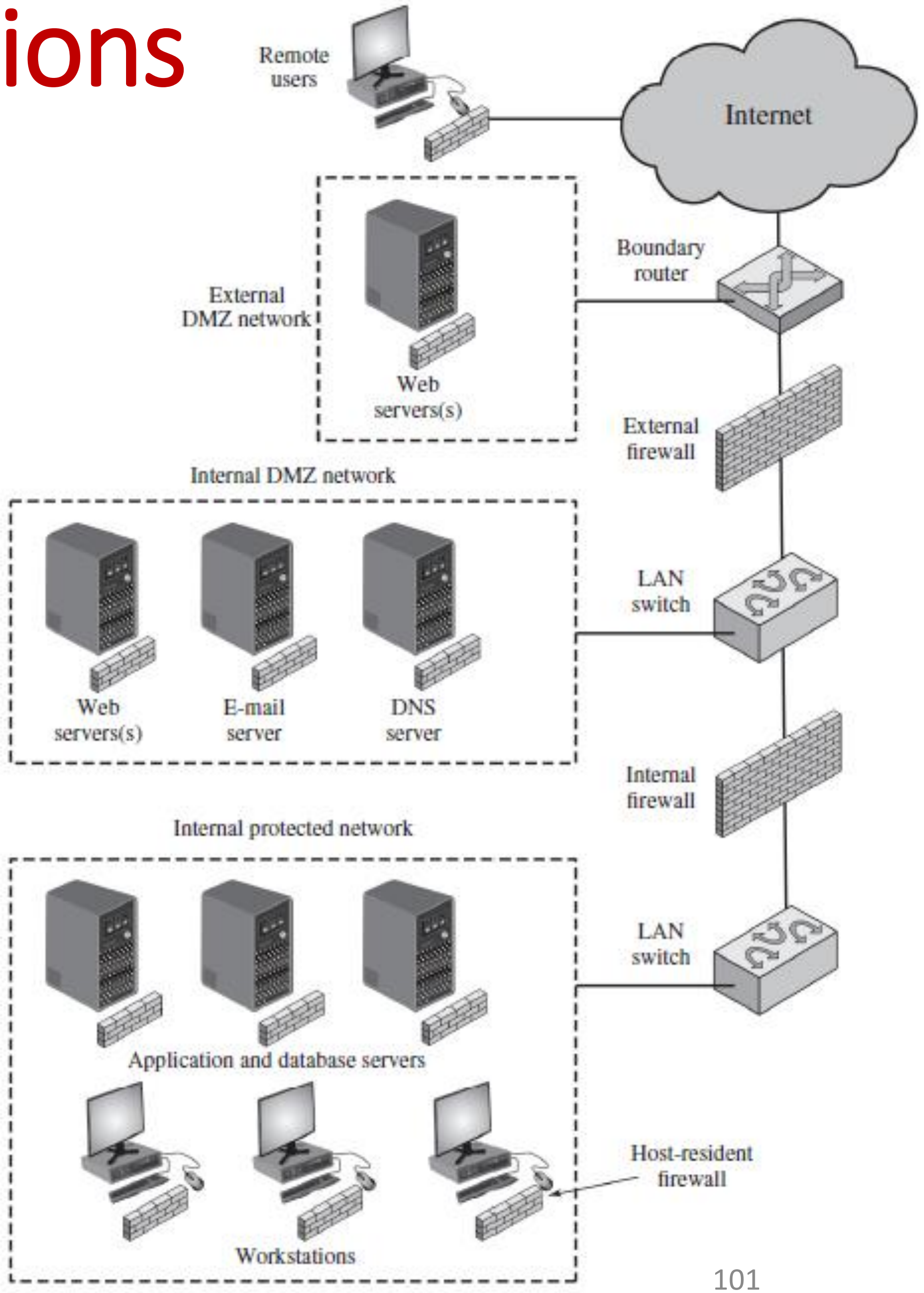


Figure 9.4 **Example Distributed Firewall Configuration**

Limitations of Firewall

- The firewall cannot protect against **attacks that bypass the firewall**.
- The firewall may **not protect fully against internal threats**, such as a disgruntled employee.
- An **internal firewall that separates portions of an enterprise network cannot guard against wireless communications** between local systems on different sides of the internal firewall.
- A laptop, PDA, or portable storage device may be used and **infected outside the corporate network** and then attached and used internally.