

BITS Pilani
Pilani | Dubai | Goa | Hyderabad

Cyber Security

Introduction – Part-1

Dr. Ramakrishna Dantu

Associate Professor, BITS Pilani

Disclaimer and Acknowledgement



- The content for these slides has been obtained from books and various other source on the Internet
- I here by acknowledge all the contributors for their material and inputs.
- I have provided source information wherever necessary
- I have added and modified the content to suit the requirements of the course

Cyber Security - Introduction



Agenda

- Computer Security Concepts
- Threats, Attacks, and Assets
- Security Functional Requirements
- Fundamental Security Design Principles
- Attack Surfaces and Attack Trees
- Computer Security Strategy
- Standards





Computer Security Concepts

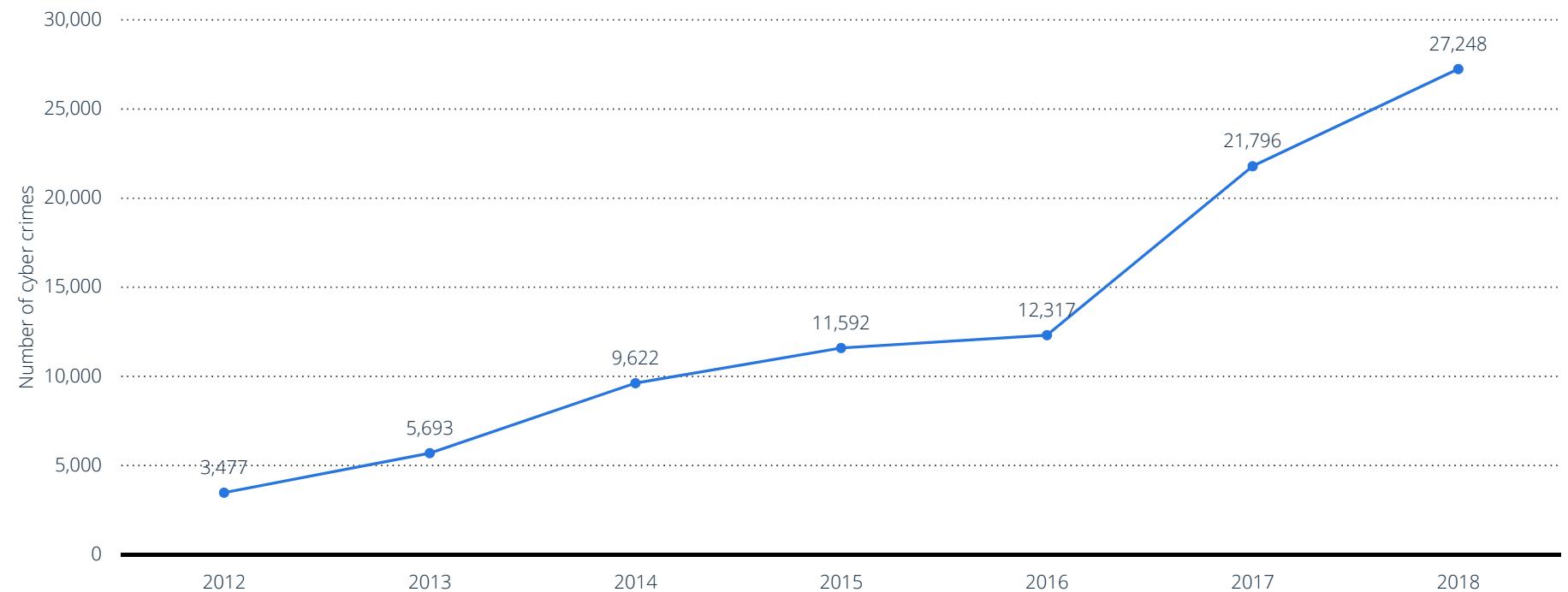


Some Facts

A circular university crest featuring a blue border with the text 'शोनं परमं बलम्' and a central emblem.

Total number of cyber crimes reported across India from 2012 to 2018

Total number of cyber crimes reported in India 2018



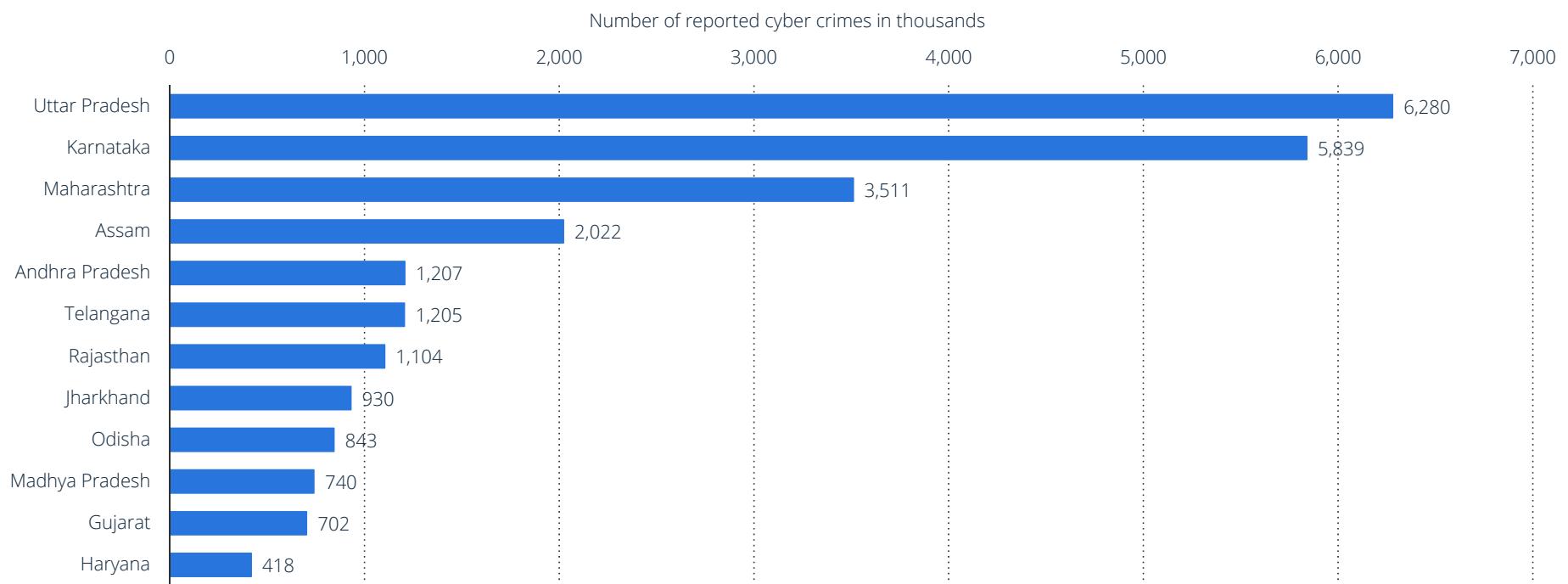
Note: India; 2012 to 2018

Further information regarding this statistic can be found on [page 31](#).

Source(s): NCRB (India); [ID 309435](#)

Number of cyber crimes reported across India in 2018, by leading state (in 1,000s)

Number of cyber crimes reported in India 2018 by leading state



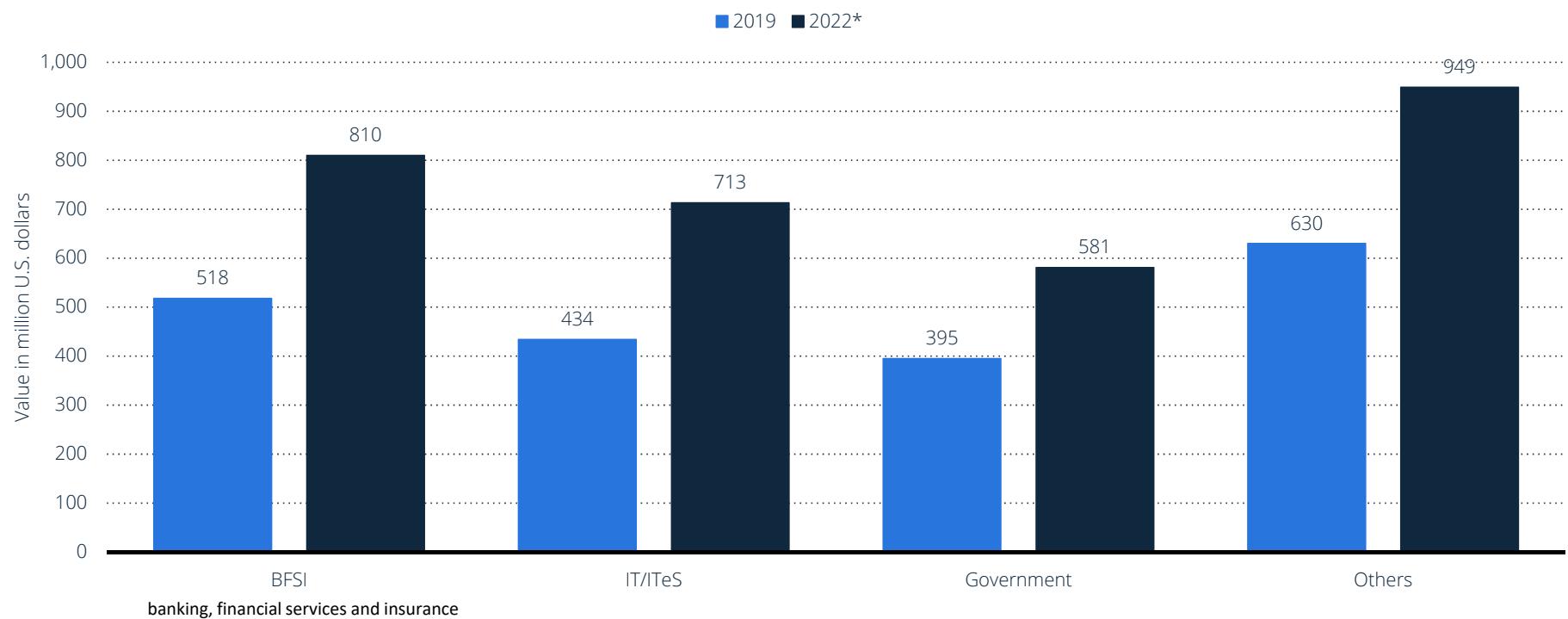
Note: India; 2018

Further information regarding this statistic can be found on [page 32](#).

Source(s): NCRB (India); [ID 1097071](#)

Value of expenditure towards cyber security in India in 2019 and 2022, by sector (in million U.S. dollars)

Value of expenditure towards cyber security India 2019-2022 by sector



Note: India; 2019

Further information regarding this statistic can be found on [page 33](#).

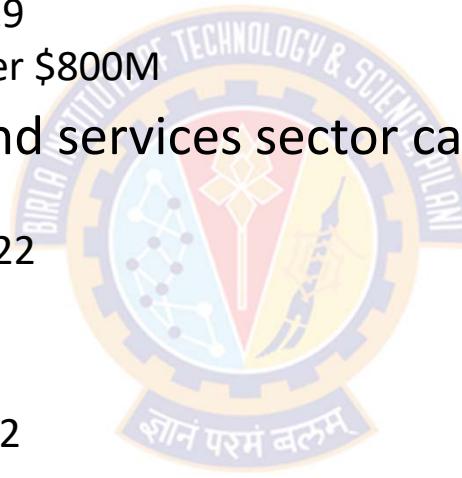
Source(s): PwC; DSCI; [ID 1099728](#)

Some Facts



Cyber Security Expenditure in India: 2019-2022

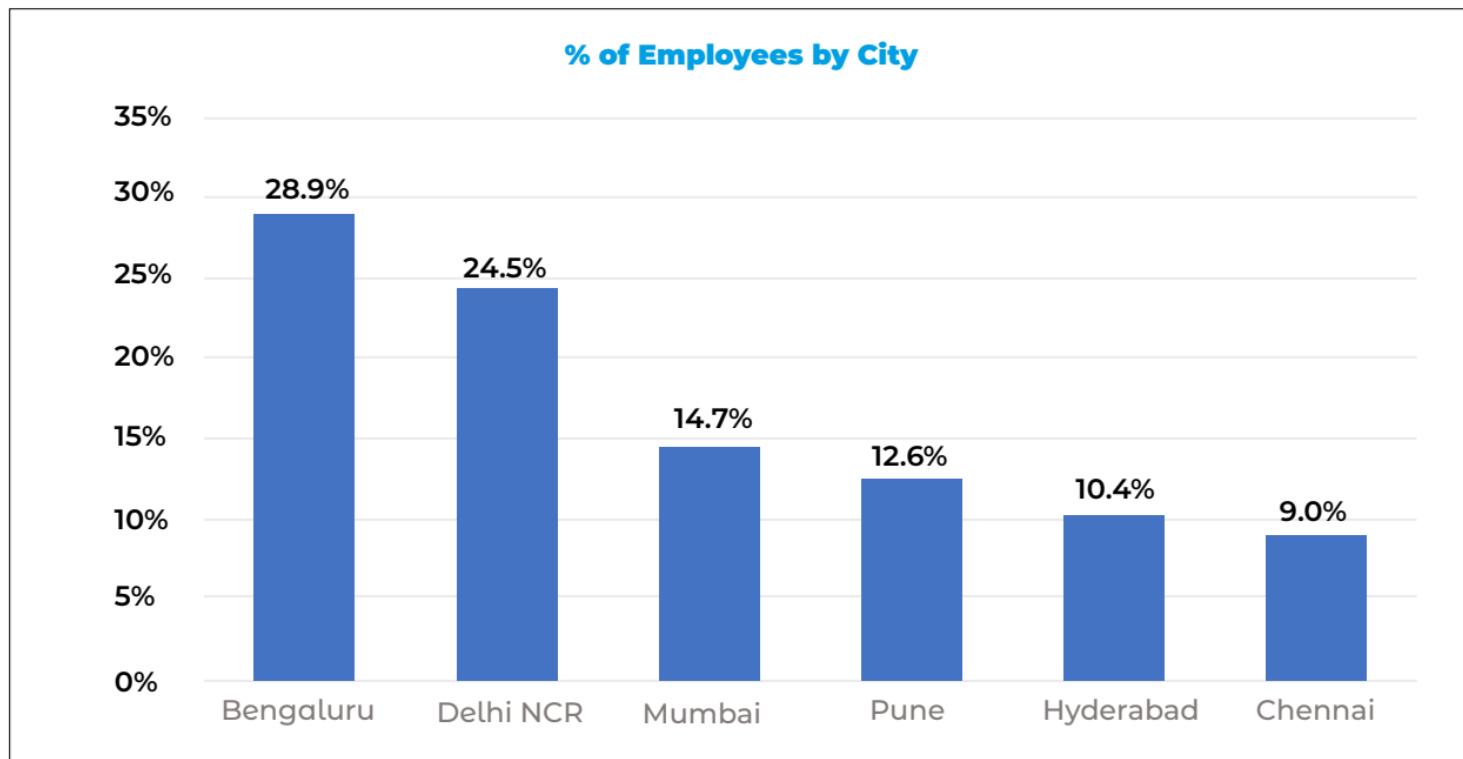
- India's BFSI sector had the highest expenditure on cyber security
 - Over 500 million U.S. dollars in 2019
 - By 2022, this is estimated to go over \$800M
- The information technology and services sector came second
 - Over \$430M in 2019
 - Estimated to go over \$700M by 2022
- Government sector
 - Close to \$400M in 2019
 - Expected to go over \$500M by 2022
- Other businesses collective expenditure
 - Over \$600 Million in 2019
 - It was estimated that these expenses would reach a billion dollars by 2022



Some Facts



Cyber Security Employee Distribution: 2020

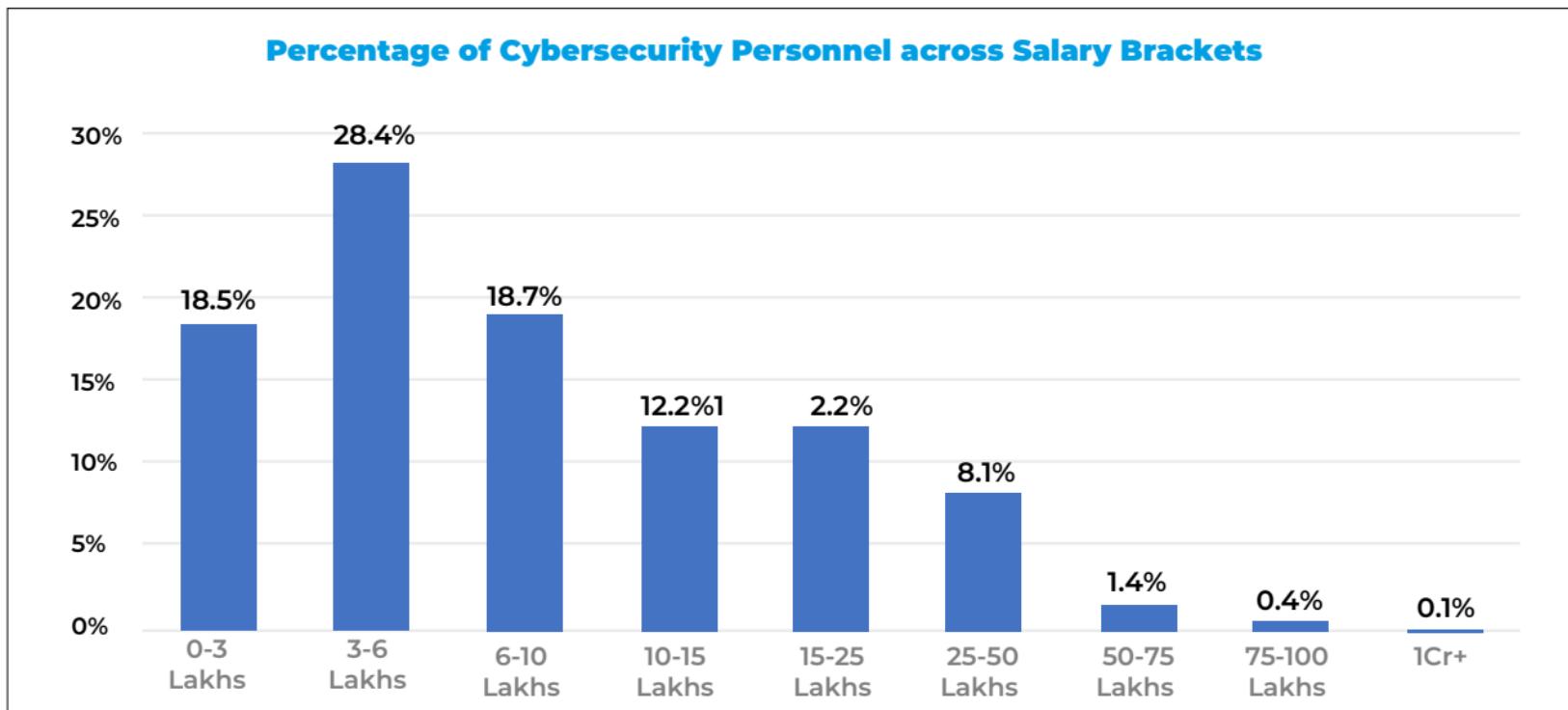


Source: State of Cyber Security in India by Jigsaw Academy & AIMResearch

Some Facts



Cyber Security Personnel Salary Brackets: 2020

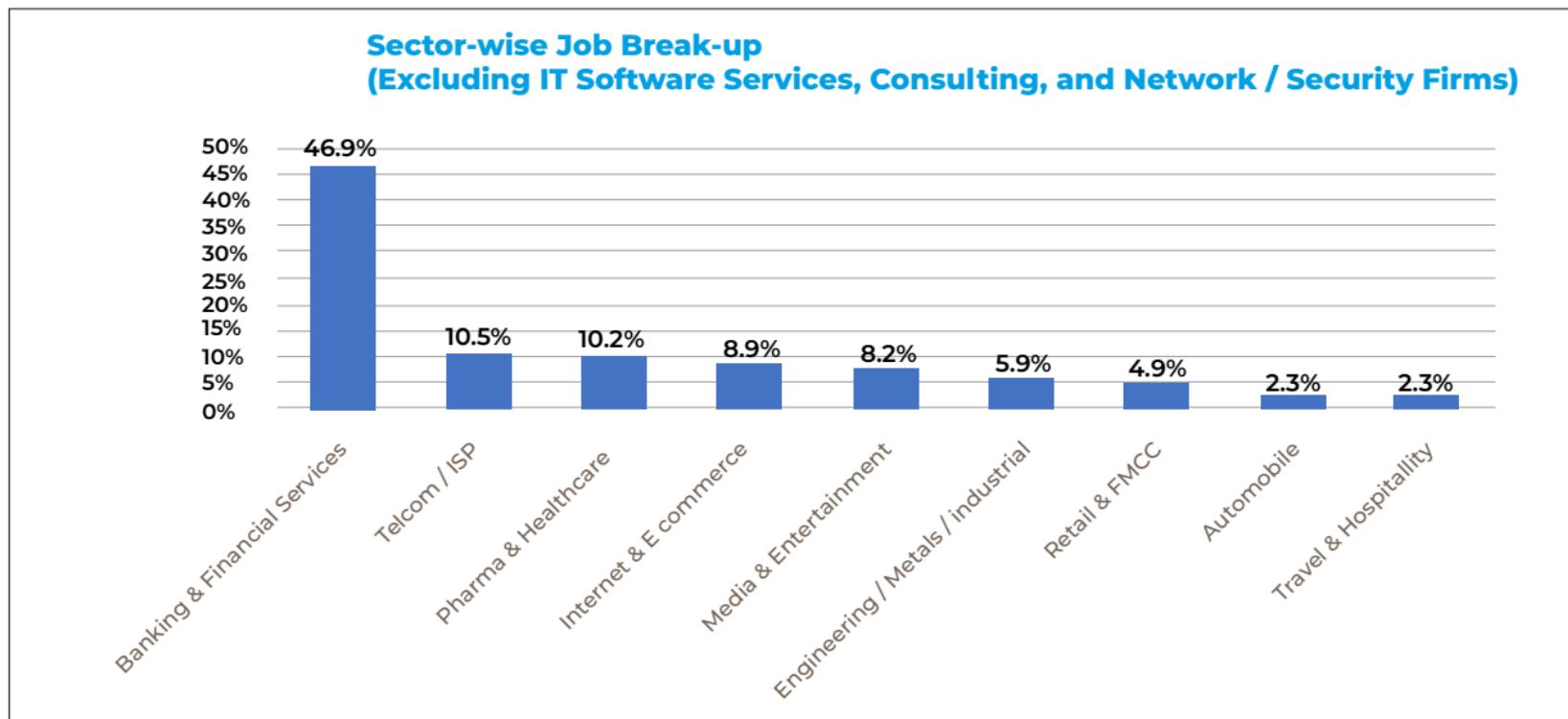


Source: State of Cyber Security in India by Jigsaw Academy & AIMResearch

Some Facts



Cyber Security Sector-wise Job Break-up: 2020



Source: State of Cyber Security in India by Jigsaw Academy & AIMResearch



A Definition of Computer Security

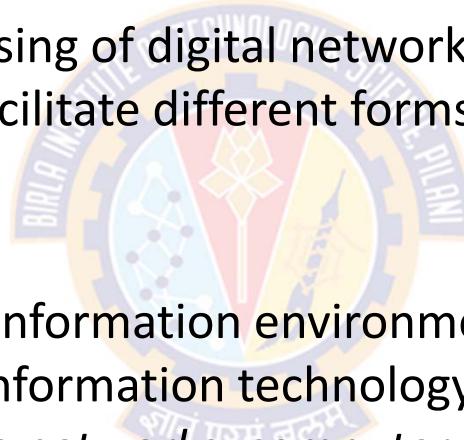


Computer Security Concepts



What is Cyber Space?

- Cyberspace refers to:
 - "An interactive space comprising of digital networks that collect, store, and manipulate information to facilitate different forms of communication"
-- Brian Walker
 - "A global domain within the information environment consisting of the interdependent network of information technology infrastructures, including the Internet, telecommunications networks, computer systems, and embedded processors and controllers."
-- NITI Aayog

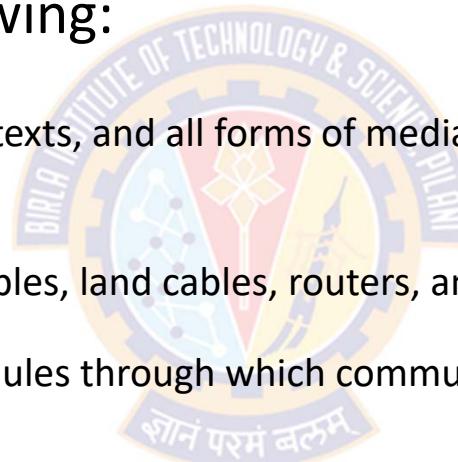


Computer Security Concepts



What is Cyber Space?

- Based on the above definitions, cyberspace is a multi-layered platform that is made up of the following:
 - Information
 - Includes financial transactions, texts, and all forms of media and social media posts, etc., stored in various places.
 - Physical foundations
 - Include satellites, submarine cables, land cables, routers, and anything else that provides a pathway for communication
 - These are the transmission modules through which communication is permitted
 - People
 - Include producers and consumers of information shared in cyberspace
 - Logical building blocks
 - These are the operating systems, applications, and web browsers that allow us to interact with the physical foundations and access information online



Computer Security Concepts



What is Cyber Security?

- "the **practice of defending** computers, servers, mobile devices, electronic systems, networks, and data from malicious attacks."
 - <https://www.kaspersky.co.in/resource-center/definitions/what-is-cyber-security>
- "**techniques of protecting** computers, networks, programs and data from unauthorized access or attacks that are aimed for exploitation."
 - <https://economictimes.indiatimes.com/definition/cyber-security>
- "the **practice of protecting** systems, networks, and programs from digital attacks."
 - https://www.cisco.com/c/en_in/products/security/what-is-cybersecurity.html
- "the **protection** of computer systems and networks from the theft of or damage to their hardware, software, or electronic data, as well as from the disruption or misdirection of the services they provide."
 - https://en.wikipedia.org/wiki/Computer_security
- "the body of technologies, processes, and practices designed to **protect** networks, devices, programs, and data from attack, damage, or unauthorized access."
 - <https://digitalguardian.com/blog/what-cyber-security>

Computer Security Concepts



What is Cyber Security?

- Data Security Council of India (DSCI)
 - A non-profit industry body on data protection in India, setup by NASSCOM®
 - Is committed to making the cyberspace **safe, secure** and **trusted** by establishing **best practices, standards and initiatives** in cyber security and privacy.
- According to DSCI, the term "cyber security" refers to three things:
 - A set of **technical** and **non-technical** activities and measures taken to protect **computers, computer networks, related hardware** and **devices software**, and the information they contain and communicate, including **software** and **data**, from all threats, including threats to the **national security**
 - The **degree of protection** resulting from the application of these activities and measures
 - The associated field of **professional endeavor**, including **research** and **analysis**, aimed at implementing and those activities and improving their quality.

Computer Security Concepts



A Definition of Computer Security

- The National Institute of Standards and Technology (NIST)
 - Is a physical sciences laboratory and a non-regulatory agency of the United States Department of Commerce
 - Is responsible for establishing technology, standards, and metrics to be applied to the science and technology industries
- The NIST Computer Security Handbook [NIST95] defines computer security as:
 - *"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)"*

Computer Security Concepts



A Definition of Computer Security

- This definition introduces three key elements of Computer Security:
 - Confidentiality
 - Integrity
 - Availability
- Referred as
the CIA Triad
- 
- The logo of Birla Institute of Technology & Science, Pilani, featuring a circular emblem with a central torch and the text "BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI" around the top and "शोनं परमं बलम्" at the bottom.

Computer Security Concepts



Key objectives of Computer Security

- **Confidentiality** covers two related concepts:

- **Data confidentiality:**

- Assures that private or confidential information is not made available or disclosed to unauthorized individuals
 - Example:
 - SSNs and other personal information must remain confidential to prevent identity theft
 - Passwords must remain confidential to protect systems and accounts.

- **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.
 - Example:
 - The Family Educational Rights and Privacy Act (FERPA) is a federal law enacted in 1974 that protects the privacy of student education records
 - The Health Insurance Portability and Accountability Act of 1996 (HIPAA) is a federal law that protects patient health information from being disclosed without the patient's consent or knowledge

Computer Security Concepts



Key objectives of Computer Security

- **Integrity** covers two related concepts:

- **Data integrity:**

- Assures that information and programs are changed only in a specified and authorized manner.
 - E.g., a user updates data fields with wrong data (phone number, address, name, etc.)

- **System integrity:**

- Assures that a system performs its intended function in an unimpaired manner, free from deliberate or inadvertent unauthorized manipulation of the system.
 - E.g., a bug in an application attempts to delete the wrong record.
 - E.g., a vending machine dispenses a wrong item for a certain choice pressed

- **Availability:**

- Assures that systems are available and work promptly and service is not denied to authorized users

Computer Security Concepts



Side Bar

- NIST has developed several standards called Federal Information Processing Standards (FIPS)
- FIPS 199 is a US Federal Government standard that establishes security categories of information systems used by the Federal Government
- FIPS 199 and FIPS 200 are mandatory security standards as required by FISMA
 - Federal Information Security Management Act of 2002
- FIPS 199 requires Federal agencies to assess their information systems in each of the categories of confidentiality, integrity and availability
 - The agencies have to rate each system as low, moderate or high impact in each category
 - The most severe rating from any category becomes the information system's overall security categorization
- FIPS 200 talks about minimum security requirements for Federal Information and Information Systems

Computer Security Concepts



Key objectives of Computer Security

- FIPS 199 provides requirements and the definition of a loss of security in each category

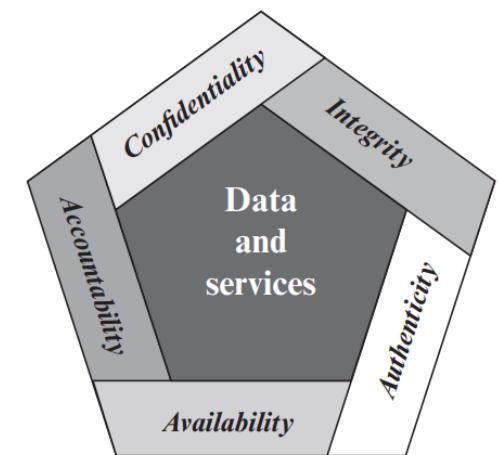
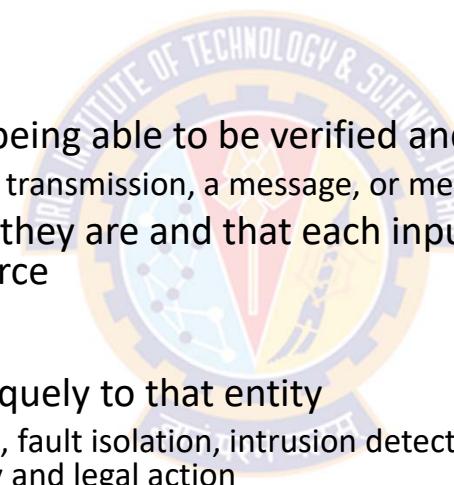
Category	Requirement	Definition of a loss of security
Confidentiality	<ul style="list-style-type: none">• Preserving authorized restrictions on information access and disclosure• Includes means for protecting personal privacy and proprietary information	<ul style="list-style-type: none">• A loss of confidentiality is the unauthorized disclosure of information
Integrity:	<ul style="list-style-type: none">• Guarding against improper modification or destruction of information• Includes ensuring information nonrepudiation and authenticity	<ul style="list-style-type: none">• A loss of integrity is the unauthorized modification or destruction of information
Availability:	<ul style="list-style-type: none">• Ensuring timely and reliable access to and use of information.	<ul style="list-style-type: none">• A loss of availability is the disruption of access to or use of information or an Information System

Computer Security Concepts



Key objectives of Computer Security

- Security experts add two additional objectives to CIA to present a complete picture
- **Authenticity:**
 - The property of being genuine and being able to be verified and trusted
 - Infuses confidence in the validity of a transmission, a message, or message originator
 - Verifies that users are who they say they are and that each input arriving at the system came from a trusted source
- **Accountability:**
 - Actions of an entity to be traced uniquely to that entity
 - Supports nonrepudiation, deterrence, fault isolation, intrusion detection and prevention, and after-action recovery and legal action
 - A security breach should be traceable to a responsible party
 - Systems must keep records of the activities to permit forensic analysis to trace security breaches or to aid in transaction disputes



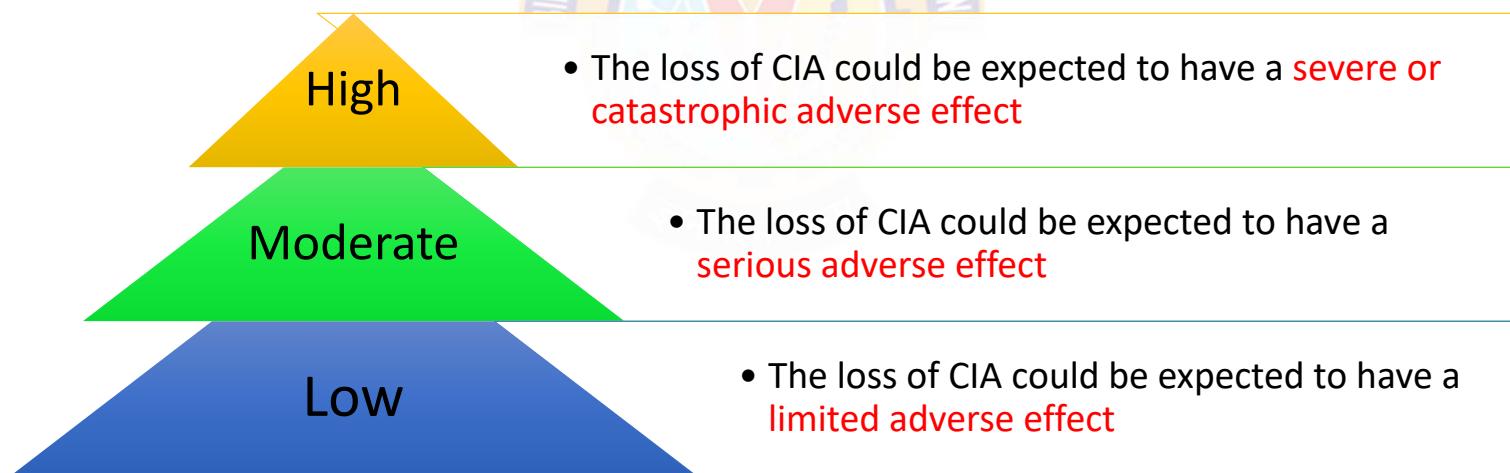
Essential Network and Computer Security Requirements

Computer Security Concepts



CIA Triad – Levels of effects due to breach of security

- Breach of security results in a loss of C, I or A
- FIPS PUB 199 defines three levels of effects on **organizational operations**, **organizational assets**, and **individuals** should there be a breach of security



Computer Security Concepts



Damages due to the loss of CIA Triad

Effect on	Breach of Security		
	Low	Moderate	High
Overall effect on organizational operations, assets, and individuals	Limited adverse effect	Serious adverse effect	Severe or catastrophic adverse effect
Extent and duration of degradation in mission capability	Minor	Significant	Severe
Organization is able to perform its primary functions	Yes, but the effectiveness of the functions is noticeably reduced	Yes, but effectiveness of the functions is significantly reduced	Not able to perform one or more of its primary functions
Organizational assets	Minor damage	Significant damage	Major damage
Financial loss	Minor	Significant	Major
Individuals	Minor harm	Significant harm	Severe or catastrophic harm
Loss of life or serious, life-threatening injuries	Not applicable	None	Yes

Computer Security Concepts



Loss to CIA Triad – Confidentiality – Example

Confidentiality	Example	Protected by	Accessibility
High	Student grade information	In the US, the release of such information is regulated by the Family Educational Rights and Privacy Act (FERPA)	<ul style="list-style-type: none">Grade information should only be available to students, their parents, and employees that require the information to do their job
Moderate	Student enrollment information	Also covered by FERPA	<ul style="list-style-type: none">This information is seen by more people on a daily basisIs less likely to be targeted than grade informationResults in less damage if disclosed
Low	Directory information	Not covered by FERPA	<ul style="list-style-type: none">E.g., lists of students or faculty or departmental listsThis information is typically freely available to the public and published on a school's Web site.

Computer Security Concepts



Loss to CIA Triad – Integrity – Example

Integrity	Example	Details
High	Patient Allergy Information	<ul style="list-style-type: none">The doctor should be able to trust that the information is correct and currentNow suppose that a nurse who is authorized to access this information deliberately falsifies the data to cause harm to the hospitalThe database needs to be restored to a trusted basis quicklyIt should be possible to trace the error back to the person responsibleInaccurate information could result in serious harm or death to a patient and expose the hospital to massive liability
Moderate	Web site	<ul style="list-style-type: none">Offers a forum to registered users to discuss specific topicsEither a registered user or a hacker could falsify some entries or deface the Web siteIf the forum exists only for the enjoyment of the users, brings in little or no advertising revenue, and is not used for something important such as research, then potential damage is not severeThe Web master may experience some data, financial, and time loss
Low	Anonymous online poll	<ul style="list-style-type: none">Many Web sites (E.g., news organizations), run polls for their users with very few safeguardsHowever, the inaccuracy and unscientific nature of such polls is well understood.

Computer Security Concepts



Loss to CIA Triad – Availability – Example

Availability	Example	Details
High	A system that provides authentication services for critical systems, applications, and devices	<ul style="list-style-type: none">An interruption of service results in the inability for<ul style="list-style-type: none">customers to access computing resourcesstaff to access the resources they need to perform critical tasks.The loss of service results into a large financial loss in lost employee productivity and potential customer loss.
Moderate	A public Web site for a university	<ul style="list-style-type: none">The Web site provides information for current and prospective students and donorsSuch a site is not a critical component of the university's information system, but its unavailability will cause some embarrassment
Low	Online telephone directory lookup application	<ul style="list-style-type: none">The temporary loss of the application may be an annoyance, butThere are other ways to access the information, such as a hardcopy directory or the operator



Challenges in Computer Security



Challenges in Computer Security

1. Computer security is not as simple as we might think
2. Constantly think about potential attacks on the security features
3. Procedures used to provide particular services are often counterintuitive
4. Physical and logical placement needs to be determined
5. No single protocol or algorithm
6. Computer security is a perpetual battle of wits between a perpetrator and the designer
7. Perceptions of no benefit from security investment
8. Security requires regular and constant monitoring
9. Security is too often an afterthought
10. Strong security viewed as an impediment

Computer Security Concepts



Challenges in Computer Security

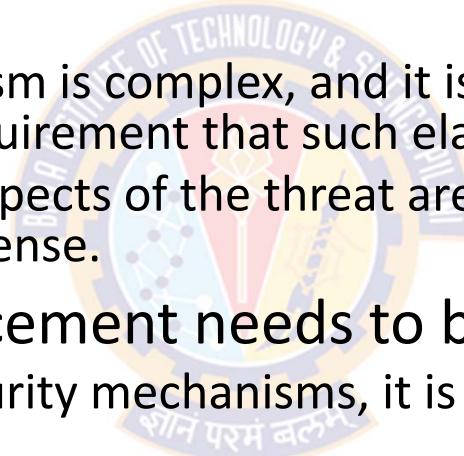
- 1) Computer security is not simple
 - The computer security requirements appear to be straightforward
 - For example, most of the major requirements for security services can be given self-explanatory one-word labels:
 - confidentiality, authentication, nonrepudiation, integrity
 - But the mechanisms used to meet those requirements can be quite complex, and understanding them may involve rather subtle reasoning
- 2) Potential attacks on security features
 - In developing a particular security mechanism or algorithm, one must always consider potential attacks on those security features
 - Most of the successful attacks are designed by looking at the problem in a completely different way, therefore exploiting an unexpected weakness in the mechanism.

Computer Security Concepts



Challenges in Computer Security

- 3) Procedures used to provide particular services are often counterintuitive
 - Typically, a security mechanism is complex, and 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.
- 4) Physical and logical placement needs to be determined
 - Having designed various security mechanisms, it is necessary to decide where to use them
 - Physical placement
 - E.g., at what points in a network are certain security mechanisms needed
 - Logical placement
 - E.g., at what layer or layers of an architecture such as TCP/IP should mechanisms be placed



Computer Security Concepts



Challenges in Computer Security

- 5) No single protocol or algorithm
 - Security mechanisms typically involve more than a particular algorithm or protocol
 - Security mechanisms also require that participants be in possession of some secret information (e.g., an encryption key)
 - This creates additional questions of creation, distribution, monitoring, and protection of that secret information
 - The behavior of communications protocols may complicate the task of developing the security mechanism
 - For example
 - If the proper functioning of the security mechanism requires setting time limits on the transit time of a message from sender to receiver, then any unpredictable delays (due to network and communication protocols) may render such time limits meaningless

Computer Security Concepts



Challenges in Computer Security

- 6) Computer security is a perpetual battle of wits between a perpetrator and the designer
 - Perpetrator – the one who tries to find holes
 - Designer – the one who tries to close them
 - Attackers only need to find a single weakness, while the designer must find and eliminate all weaknesses to achieve perfect security
- 7) Perceptions of no benefit from security investment
 - There is a natural tendency on the part of users and system managers to perceive little benefit from security investment until a security failure occurs

Computer Security Concepts



Challenges in Computer Security

- 8) Security requires regular and constant monitoring
 - Constantly monitoring security would be difficult in today's short-term, overloaded environment
 - Think of security forces guarding our national borders 24/7
- 9) Security is too often an afterthought
 - Many times, security is incorporated into the system after the design is complete, rather than being an integral part of the design process
- 10) Strong security is viewed as an impediment
 - Many users, including security admins view strong security as an obstruction to smooth operation of an IS or information use



Terminology

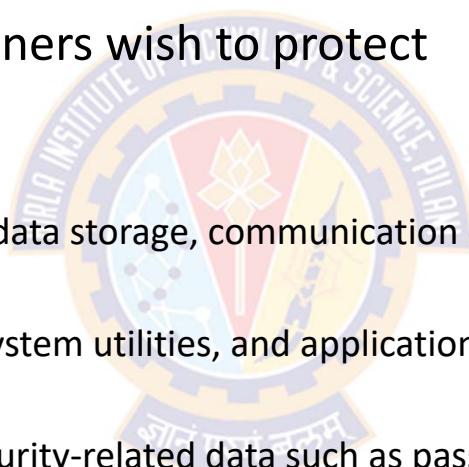
शोनं परमं बलम्

Computer Security Concepts



Terminology

- Asset
 - Something that users and owners wish to protect
 - Can be categorized as:
 - Hardware
 - Includes computer systems, data storage, communication devices
 - Software
 - Includes operating system, system utilities, and application software
 - Data
 - Includes files, databases, security-related data such as passwords
 - Networks and Communication Facilities
 - Includes local and wide area network communication networks, bridges, routers, etc.



Computer Security Concepts



Terminology

- **Vulnerability**
 - Weakness in an information system, system security procedures, or internal controls that could be exploited by a threat source
- **General categories of vulnerabilities of assets (system resources)**
 - Leaky system (Confidentiality issue)
 - E.g., someone who should not have access to information through network obtains such access
 - A weakness in a firewall that lets hackers get into a computer network
 - Corrupted system (Integrity issue)
 - The system does wrong things or gives wrong answers
 - E.g., A malicious macro in a Word document inserts the word "not" after some random instances of the word "is"
 - Unavailable or slow system (Availability issue)
 - Using the system or network becomes impossible or impractical

Computer Security Concepts



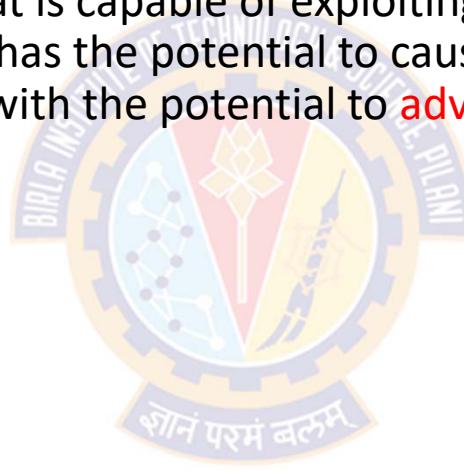
Terminology

- Threat

- A threat is a possible danger that is capable of exploiting a vulnerability
- It is a set of circumstances that has the potential to cause loss or harm
- It is any **circumstance or event** with the potential to **adversely impact**:
 - organizational operations
 - organizational assets
 - individuals
 - other organizations, or
 - the Nation

using an ICT via

- unauthorized access
- destruction
- disclosure
- modification of information, and/or
- denial of service



Computer Security Concepts



Terminology

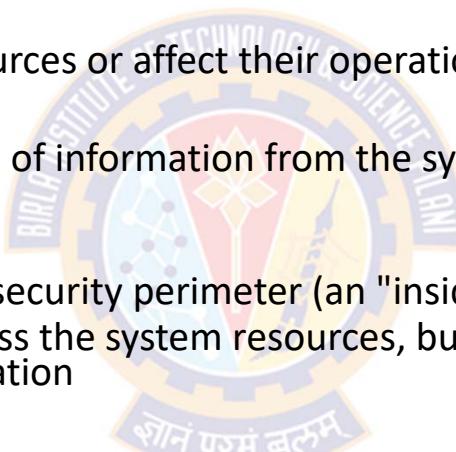
- Attack
 - An attack is a threat that is carried out (threat action)
 - An intelligent act that is a deliberate attempt to evade security services and violate the security policy of a system
 - Any kind of malicious activity that attempts to collect, disrupt, deny, degrade, or destroy information system resources or the information itself
 - A successful attack can lead to violation of security, or threat consequence
- Adversary (Threat agent)
 - An individual, group, organization, or government that conducts or has the intent to conduct detrimental activities
 - An agent carrying out the attack is referred to as an attacker or threat agent

Computer Security Concepts



Terminology

- 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
 - Inside attack
 - Initiated by an entity inside the security perimeter (an "insider")
 - The insider is authorized to access the system resources, but uses them in a way not approved by those who granted the authorization
 - Outside attack
 - Initiated from outside the perimeter, by an unauthorized or illegitimate user of the system (an "outsider")
 - On the Internet, outside attackers range from amateur pranksters to organized criminals, international terrorists, and hostile governments



Computer Security Concepts



Terminology

- Countermeasure

- A device or technique that is used to:
 - prevent a particular type of attack from succeeding
 - impair the operational effectiveness of undesirable or adversarial activity, or
 - prevent espionage, sabotage, theft, or unauthorized access to or use of sensitive information or information systems
- 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
 - When prevention is not possible, or fails in some instance, the goal is to detect the attack then recover from the effects of the attack

Computer Security Concepts



Terminology

- Risk

- An expectation of loss expressed as the probability that a particular threat will exploit a particular vulnerability with a particular harmful result.
- A measure of the extent to which an entity is threatened by a potential circumstance or event, and typically a function of
 - 1) the likelihood of occurrence
 - 2) the adverse impacts that would arise if the circumstance or event occurs

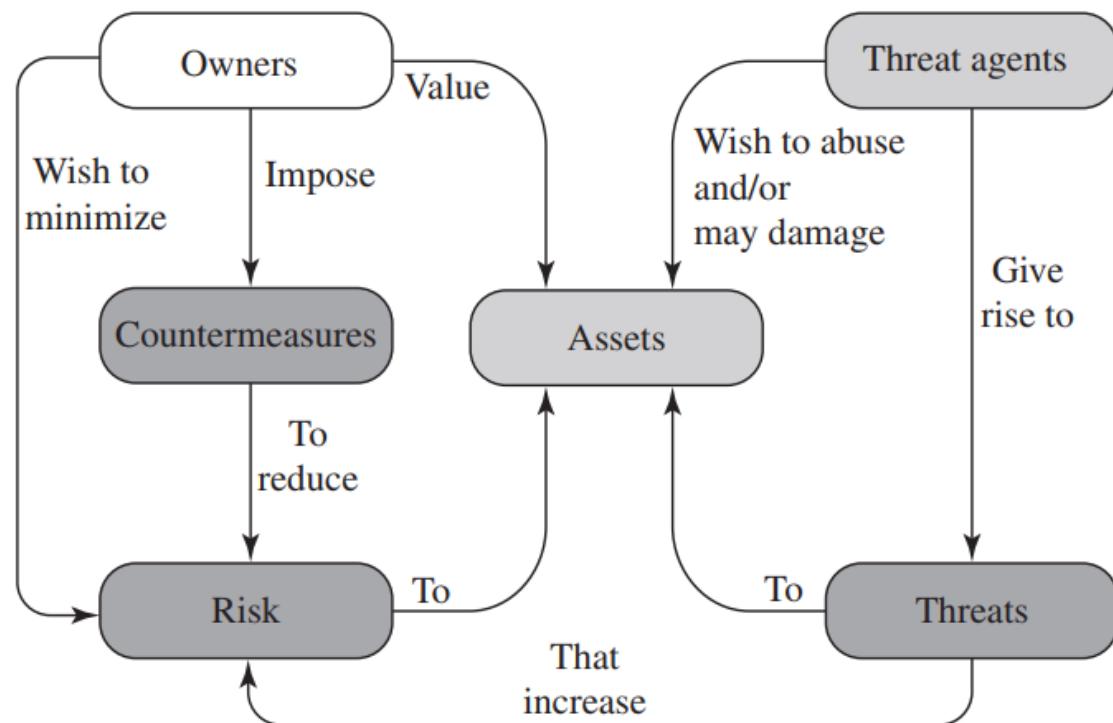
- Security Policy

- 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
- It defines and constrains the activities of a data processing facility in order to maintain a condition of security for systems and data
- Example

Computer Security Concepts



Security Concepts and Relationships





BITS Pilani
Pilani | Dubai | Goa | Hyderabad

Cyber Security

Introduction – Part-1

Dr. Ramakrishna Dantu

Associate Professor, BITS Pilani

Disclaimer and Acknowledgement



- The content for these slides has been obtained from books and various other source on the Internet
- I here by acknowledge all the contributors for their material and inputs.
- I have provided source information wherever necessary
- I have added and modified the content to suit the requirements of the course

Cyber Security - Introduction



Agenda

- Computer Security Concepts
- Threats, Attacks, and Assets
- Security Functional Requirements
- Fundamental Security Design Principles
- Attack Surfaces and Attack Trees
- Computer Security Strategy
- Standards





Threats, Attacks, and Assets



Threats & Attacks



Threats & Attacks



Threat Consequences

- Threat consequence is a security violation that results from a threat action
- Types of threat consequences and corresponding attacks that result in each of these consequences

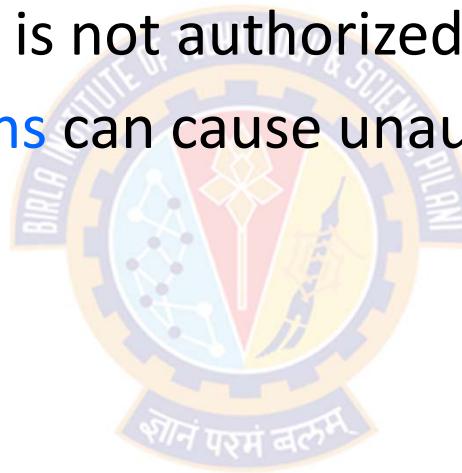
Threat Consequence	CIA Component	Type of Threat Action
Unauthorized Disclosure	Is a threat to confidentiality	Exposure; Interception; Inference; Intrusion
Deception	Is a threat to system or data integrity	Masquerade; Falsification; Repudiation
Disruption	Is a threat to availability or system integrity	Incapacitation; Corruption; Obstruction
Usurpation	Is a threat to system integrity	Misappropriation; Misuse



Threats & Attacks

Unauthorized Disclosure

- A circumstance or event whereby an entity gains access to the asset (data) for which the entity is not authorized
- The following **threat actions** can cause unauthorized disclosure:
 - Exposure
 - Interception
 - Inference
 - Intrusion





Threats & Attacks

Unauthorized Disclosure

- Exposure
 - A threat action whereby sensitive data is **directly released** to an unauthorized entity
 - Involves exposing confidential and sensitive information to an outsider
 - This attack results in an entity gaining unauthorized access of sensitive data
 - This can be **deliberate**
 - E.g., when an insider intentionally releases credit card numbers to an outsider
 - This can also be **an error** resulting from humans, hardware, or software error,
 - E.g., universities accidentally posting student confidential information on the Web
- Intrusion
 - A threat action whereby an unauthorized entity gains access to sensitive data by **circumventing** or **bypassing** a system's security protections

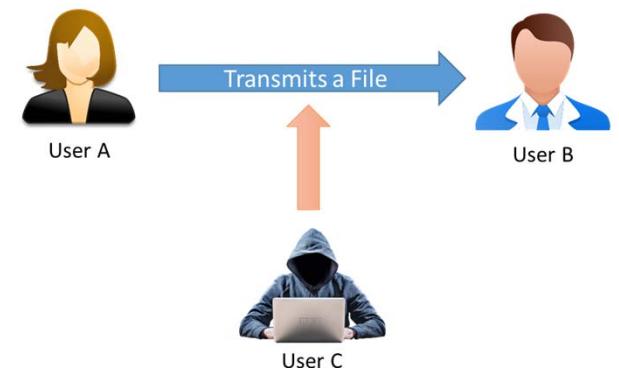
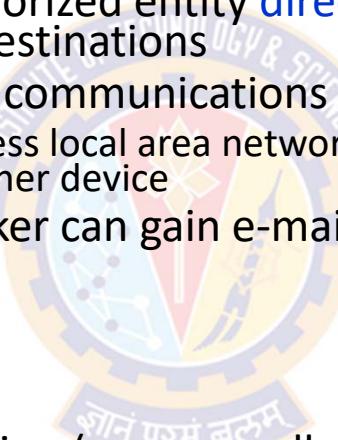
Threats & Attacks



Unauthorized Disclosure

- Interception
 - A threat action whereby an unauthorized entity directly accesses sensitive data travelling between authorized sources and destinations
 - A common attack in the context of communications
 - E.g., Any device attached to a wireless local area network (LAN) or a broadcast Ethernet can receive a copy of packets intended for another device
 - On the Internet, a determined hacker can gain e-mail access and other data transfers

- Scenario
 - User A transmits a file to user B
 - The file contains sensitive information (e.g., payroll records) that is to be protected from disclosure
 - User C, who is not authorized to read the file, is able to monitor the transmission and capture a copy of the file during its transmission.



Threats & Attacks



Unauthorized Disclosure

- Inference
 - A threat action whereby an unauthorized entity **indirectly accesses** sensitive data by reasoning from characteristics or byproducts of communications
 - E.g., **Traffic analysis**
 - An adversary is able to gain access to information from observing the pattern of traffic on a network
 - E.g., amount of traffic between pairs of hosts on the network
 - Traffic analysis is performed to **infer** from trivial information more robust information such as location of key nodes, routing structure, etc.,,
 - This is accomplished by repeated queries whose combined results enable inference
 - Once the base node is located, the attacker can accurately launch a host of attacks against the base station such as jamming, eavesdropping, etc.,.

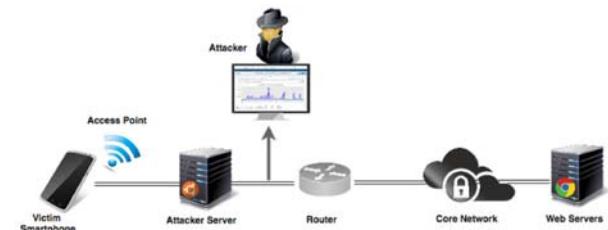


Image Source: Kausar et al., 2019, Traffic Analysis Attack for Identifying Users' Online Activities, Published in IT Professional 2019

Threats & Attacks



Unauthorized Disclosure

Threat Action	Types of Threat Actions	Description
Exposure <i>A threat action whereby sensitive data is directly released to an unauthorized entity.</i>	Deliberate Exposure	Intentional release of sensitive data to an unauthorized entity.
	Scavenging	Searching through data residue in a system to gain unauthorized knowledge of sensitive data
	Human Error	Human action or inaction that unintentionally results in an entity gaining unauthorized knowledge of sensitive data.
	Hardware/software error	System failure that results in an entity gaining unauthorized knowledge of sensitive data.

Threats & Attacks



Unauthorized Disclosure

Threat Action	Types of Threat Actions	Description
Intrusion <i>A threat action whereby an unauthorized entity gains access to sensitive data by circumventing a system's security protections.</i>	Trespass	Gaining unauthorized physical access to sensitive data by circumventing a system's protections.
	Penetration	Gaining unauthorized logical access to sensitive data by circumventing a system's protections.
	Reverse Engineering	Acquiring sensitive data by disassembling and analyzing the design of a system component.
	Cryptanalysis	Transforming encrypted data into plain text without having prior knowledge of encryption parameters or processes.



Threats & Attacks

Unauthorized Disclosure

Threat Action	Types of Threat Actions	Description
Interception <i>A threat action whereby an unauthorized entity directly accesses sensitive data travelling between authorized sources and destinations</i>	Theft	Gaining access to sensitive data by stealing a shipment of a physical medium, such as a magnetic tape or disk, that holds the data.
	Wiretapping	Monitoring and recording data that is flowing between two points in a communication system
	Emanations Analysis	Gaining direct knowledge of communicated data by monitoring and resolving a signal that is emitted by a system and that contains the data but is not intended to communicate the data.

An emanation is a form of energy or a mass of tiny particles that comes from something
E.g., Emanation of light or sound

Threats & Attacks



Unauthorized Disclosure

Threat Action	Types of Threat Actions	Description
<p>Inference</p> <p><i>A threat action whereby an unauthorized entity indirectly accesses sensitive data (but not necessarily the data contained in the communication) by reasoning from characteristics or byproducts of communications</i></p>	Traffic Analysis Signal Analysis	<p>Gaining knowledge of data by observing the characteristics of communications that carry the data.</p> <p>Gaining indirect knowledge of communicated data by monitoring and analyzing a signal that is emitted by a system and that contains the data but is not intended to communicate the data.</p>

Threats & Attacks



Deception

- A circumstance or event that may result in an authorized entity **receiving false data** and **believing it to be true**
- The following threat actions can cause deception:
 - Masquerade
 - A threat action whereby an unauthorized entity gains access to a system or performs a malicious act by posing as an authorized entity.
 - Falsification
 - A threat action whereby false data deceives an authorized entity
 - Repudiation
 - A threat action whereby an entity deceives another by falsely denying responsibility for an act.

Threats & Attacks



Deception

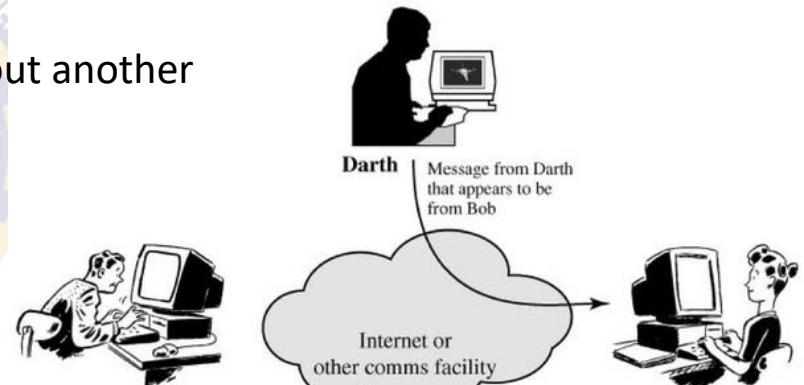
- **Masquerade**

- E.g., an attempt by an unauthorized user to gain access to a system by posing as an authorized user

- This can happen if the unauthorized user learns about another user's login ID and password

- E.g., Malicious logic such as Trojan horse

- The software performs a useful or desirable function but actually gains unauthorized access to system resources



Active Attack – Masquerade

Threats & Attacks



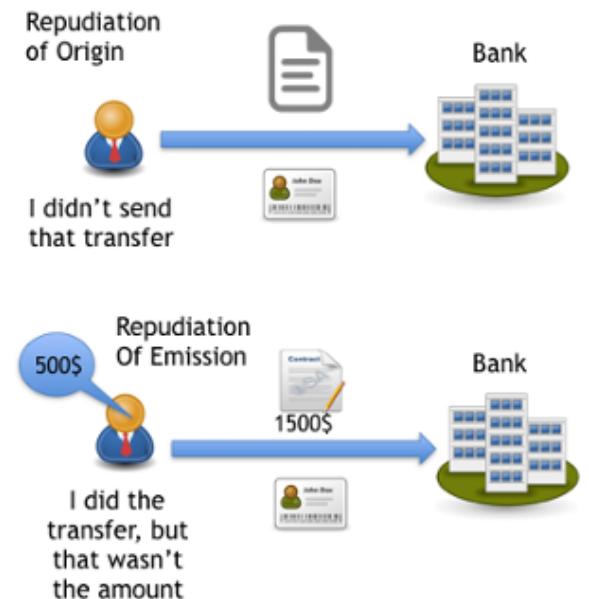
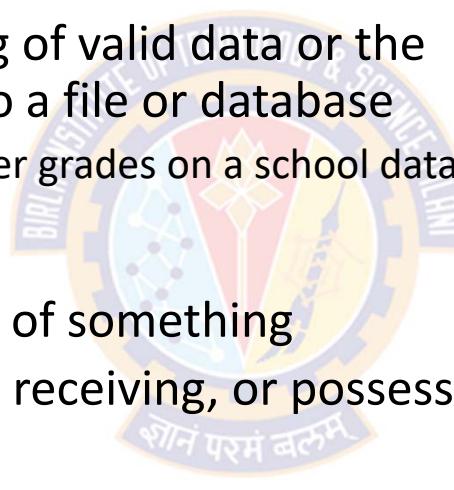
Deception

- **Falsification**

- Refers to altering or replacing of valid data or the introduction of false data into a file or database
 - E.g., a student may alter his/her grades on a school database

- **Repudiation**

- Denial of the truth or validity of something
- A user either denies sending, receiving, or possessing the data

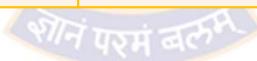


Threats & Attacks



Deception

Threat Action	Types of Threat Actions	Description
Masquerade <i>A threat action whereby an unauthorized entity gains access to a system or performs a malicious act by posing as an authorized entity.</i>	Spoof	Attempt by an unauthorized entity to gain access to a system by posing as an authorized user.
	Malicious Logic	In context of masquerade, any hardware, firmware, or software (e.g., Trojan horse) that appears to perform a useful or desirable function, but actually gains unauthorized access to system resources or tricks a user into executing other malicious logic.



Threats & Attacks



Deception

Threat Action	Types of Threat Actions	Description
Falsification <i>A threat action whereby false data deceives an authorized entity</i>	Substitution	Altering or replacing valid data with false data that serves to deceive an authorized entity.
	Insertion	Introducing false data that serves to deceive an authorized entity
Repudiation <i>A threat action whereby an entity deceives another by falsely denying responsibility for an act.</i>	False Denial of Origin	Action whereby the originator of data denies responsibility for its generation.
	False denial of receipt	Action whereby the recipient of data denies receiving and possessing the data.



Threats & Attacks

Disruption

- A circumstance or event that **interrupts or prevents** the correct operation of system services and functions.
- The following threat actions can cause disruption:
 - Incapacitation:
 - Prevents or interrupts system operation by disabling a system component.
 - Corruption:
 - Undesirably alters system operation by adversely modifying system functions or data.
 - Obstruction:
 - Interrupts delivery of system services by hindering system operations.



Threats & Attacks

Disruption

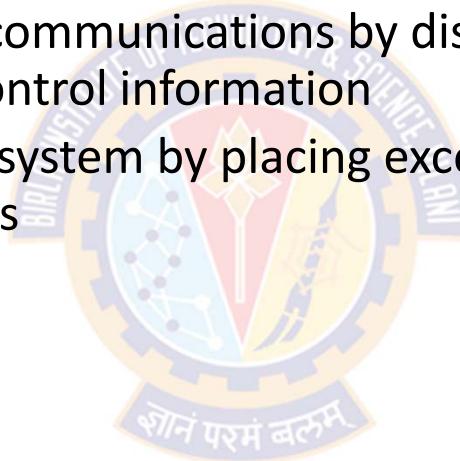
- Incapacitation (attack on system availability)
 - Could occur as a result of physical destruction or damage to system hardware
 - Trojan horses, viruses, or worms disable a system or some of its services
- Corruption (attack on system integrity)
 - Malicious software can make system resources or services function in an unintended manner
 - A user could gain unauthorized access to a system and modify some of its functions
 - E.g., user places a backdoor logic in the system to provide subsequent access to a system and its resources by other than the usual procedure

Threats & Attacks



Disruption

- Obstruction (attack on system availability)
 - One way is to interfere with communications by disabling the communication links or altering communication control information
 - Other way is to overload the system by placing excess burden on communication traffic or processing resources





Threats & Attacks

Disruption

Threat Action	Types of Threat Actions	Description
Incapacitation <i>Prevents or interrupts system operation by disabling a system component</i>	Malicious Logic	In the context of incapacitation, any hardware, firmware, or software (e.g., logic bomb) intentionally introduced into a system to destroy system functions or resources.
	Physical Destruction	Deliberate destruction of a system component to interrupt or prevent system operation.
	Human Error	Action or inaction that unintentionally disables a system component.
	Hardware or software error	Error that causes failure of a system component and leads to disruption of system operation.
	Natural disaster	Any natural disaster (e.g., fire, flood, earthquake, lightning, or wind) that disables a system component.[19]

A logic bomb is a piece of code that will set off a malicious function when specified conditions are met. For example, a programmer may hide a piece of code that starts deleting files (such as a salary database trigger), should they ever be terminated from the company.



Threats & Attacks

Disruption

Threat Action	Types of Threat Actions	Description
<p>Corruption</p> <p>A threat action that undesirably alters system operation by adversely modifying system functions or data.</p>	Tamper	In the context of corruption, deliberate alteration of a system's logic, data, or control information to interrupt or prevent correct operation of system functions.
	Malicious Logic	In the context of corruption, any hardware, firmware, or software (e.g., a computer virus) intentionally introduced into a system to modify system functions or data.
	Human Error	Human action or inaction that unintentionally results in the alteration of system functions or data.
	Hardware or Software Error	Error that results in the alteration of system functions or data.
	Natural Disaster	Any natural event (e.g. power surge caused by lightning) that alters system functions or data.[19]

Threats & Attacks



Disruption

Threat Action	Types of Threat Actions	Description
Obstruction <i>A threat action that interrupts delivery of system services by hindering system operations.</i>	Interference	Disruption of system operations by blocking communications or user data or control information.
	Overload	Hindrance of system operation by placing excess burden on the performance capabilities of a system component. (flooding.)





Threats & Attacks

Usurpation

- A circumstance or event that results in **taking control of** system services or functions without having a right to (by an unauthorized entity)
- The following threat actions can cause usurpation:
 - Misappropriation
 - An entity assumes logical or physical control of a system resource.
 - This can include theft of service
 - E.g., distributed denial of service attack
 - When malicious software is installed on a number of hosts to be used as platforms to launch traffic at a target host
 - In this case, the malicious software makes unauthorized use of processor and operating system resources.
 - Misuse
 - Causes a system component to perform a function or service that is detrimental to system security
 - Occurs by means of either malicious logic or a hacker that has gained unauthorized access to a system



Threats & Attacks

Usurpation

Threat Action	Types of Threat Actions	Description
Misappropriation <i>An entity assumes unauthorized logical or physical control of a system resource.</i>	Theft of Service	Unauthorized use of service by an entity.
	Theft of functionality	Unauthorized acquisition of actual hardware, software, or firmware of a system component.
	Theft of data	Unauthorized acquisition and use of data.
Misuse <i>A threat action that causes a system component to perform a function or service that is detrimental to system security.</i>	Tamper	A deliberate alteration of a system's logic, data, or control information to cause the system to perform unauthorized functions or services.
	Malicious Logic	Any hardware, software, or firmware intentionally introduced into a system to perform or control the execution of an unauthorized function or service.
	Violation of permissions	Action by an entity that exceeds the entity's system privileges by executing an unauthorized function.



Threats & Attacks

Summary

Threat Consequence	Threat Action (Attack)
Unauthorized Disclosure <i>A circumstance or event whereby an entity gains access to data for which the entity is not authorized</i>	<p>Exposure: Sensitive data are directly released to an unauthorized entity.</p> <p>Interception: An unauthorized entity directly accesses sensitive data traveling between authorized sources and destinations.</p> <p>Inference: A threat action whereby an unauthorized entity indirectly accesses sensitive data (but not necessarily the data contained in the communication) by reasoning from characteristics or by-products of communications.</p> <p>Intrusion: An unauthorized entity gains access to sensitive data by circumventing a system's security protections.</p>
Deception <i>A circumstance or event that may result in an authorized entity receiving false data and believing it to be true</i>	<p>Masquerade: An unauthorized entity gains access to a system or performs a malicious act by posing as an authorized entity.</p> <p>Falsification: False data deceive an authorized entity.</p> <p>Repudiation: An entity deceives another by falsely denying responsibility for an act.</p>

Threats & Attacks



Summary

Threat Consequence	Threat Action (Attack)
Disruption <i>A circumstance or event that interrupts or prevents the correct operation of system services and functions.</i>	<p>Incapacitation: Prevents or interrupts system operation by disabling a system component.</p> <p>Corruption: Undesirably alters system operation by adversely modifying system functions or data.</p> <p>Obstruction: A threat action that interrupts delivery of system services by hindering system operation.</p>
Usurpation <i>A circumstance or event that results in control of system services or functions by an unauthorized entity.</i>	<p>Misappropriation: An entity assumes unauthorized logical or physical control of a system resource.</p> <p>Misuse: Causes a system component to perform a function or service that is detrimental to system security.</p>



Threats & Assets

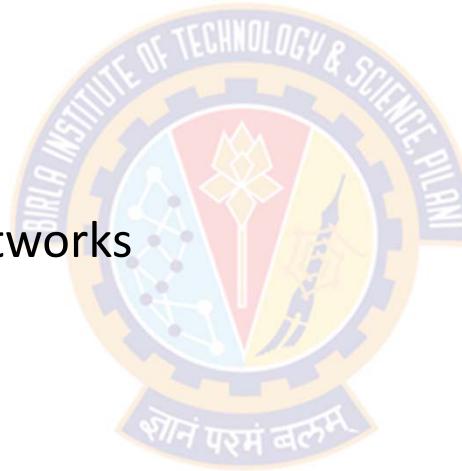


Threats & Assets



Categories

- The assets of a computer system can be categorized as:
 - Hardware
 - Software
 - Data
 - Communication lines and networks



Threats & Assets



Hardware

- Includes personal computers, workstations, networks, and peripherals such as USB Drives, External Hard drives, etc.
 - Availability
 - A major threat to computer system hardware is the threat of availability
 - Hardware is the most vulnerable to attack and automated controls have least effect on them
 - Threats include accidental and deliberate damage to equipment as well as theft
 - The proliferation of personal computers and workstations and the widespread use of LANs increase the potential for losses in this area
 - Confidentiality
 - Theft of USB Drives can lead to loss of confidentiality
 - Physical and administrative security measures are needed to deal with these threats



Threats & Assets

Software

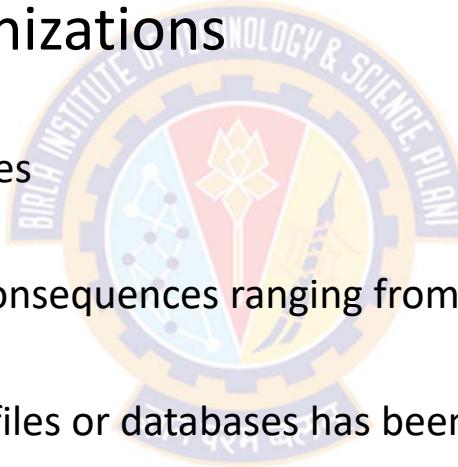
- Includes the operating system, utilities, and application programs
 - Availability
 - Application software, is often easy to delete
 - Software can also be altered or damaged to render it useless
 - Software configuration management, which includes making backups of the most recent version of software, can improve availability
 - Integrity
 - A modified software can still function but that behaves differently than before
 - Computer viruses and related attacks fall into this category
 - Confidentiality
 - Protection against software piracy is a major challenge
 - Although certain countermeasures are available, by and large the problem of unauthorized copying of software has not been solved.

Threats & Assets



Data

- Involves files and other forms of data controlled by individuals, groups, and business organizations
 - Availability
 - Involves destruction of data files
 - Integrity
 - Data modifications can have consequences ranging from minor to disastrous
 - Confidentiality
 - Unauthorized reading of data files or databases has been the most researched topic in the area of computer security

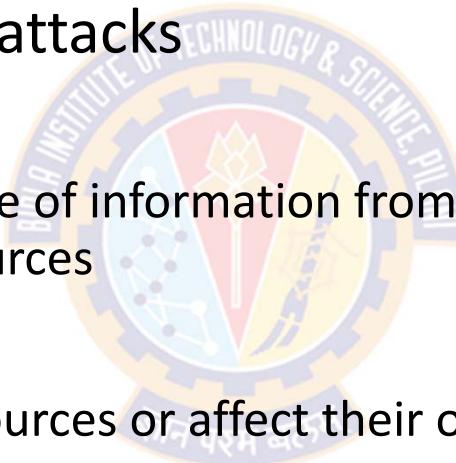


Threats & Assets



Communication Lines and Networks

- Attacks on communication lines and networks can be classified as passive attacks and active attacks
- Passive attack
 - Attempts to learn or make use of information from the system but does not cause any harm to the system resources
- Active attack
 - Attempts to alter system resources or affect their operation





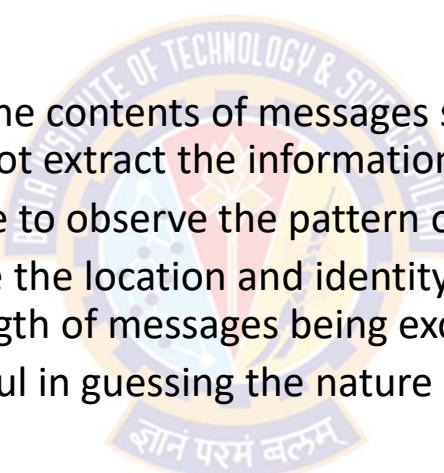
Threats & Assets

Communication Lines and Networks

- Passive attack
 - They are in the nature of eavesdropping on (monitoring of) transmissions
 - The goal of the attacker is to obtain information that is being transmitted
 - Two types of passive attacks:
 - Release of message contents
 - Traffic analysis
 - Release of message contents
 - E.g., a telephone conversation, an electronic mail message, and a transferred file may contain sensitive or confidential information
 - We would like to prevent an opponent from learning the contents of these transmissions.

Communication Lines and Networks

- Passive attack
 - Traffic analysis
 - Suppose that we can encrypt the contents of messages so that opponents, even if they captured the message, could not extract the information from the message
 - An opponent might still be able to observe the pattern of these messages
 - The opponent could determine the location and identity of communicating hosts and could observe the frequency and length of messages being exchanged
 - This information might be useful in guessing the nature of the communication that was taking place



Threats & Assets



Communication Lines and Networks

- Passive attack
 - Passive attacks are very difficult to detect because they do not involve any alteration of the data
 - Typically, the message traffic is sent and received in an apparently normal fashion and neither the sender nor receiver is aware that a third party has read the messages or observed the traffic pattern
 - However, it is feasible to prevent the success of these attacks, usually by means of encryption
 - Thus, the **emphasis** in dealing with passive attacks is on **prevention rather than detection**



Threats & Assets

Communication Lines and Networks

- Active attacks
 - They involve some modification of the data stream or the creation of a false stream and can be subdivided into four categories:
 - Replay
 - Masquerade
 - Modification of messages, and
 - Denial of service.



Threats & Assets



Communication Lines and Networks

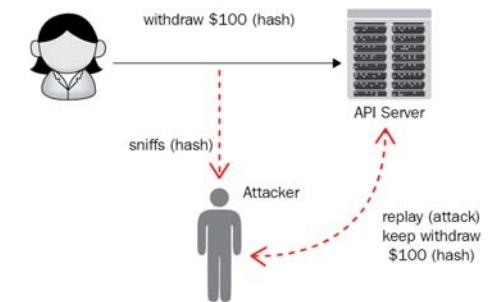
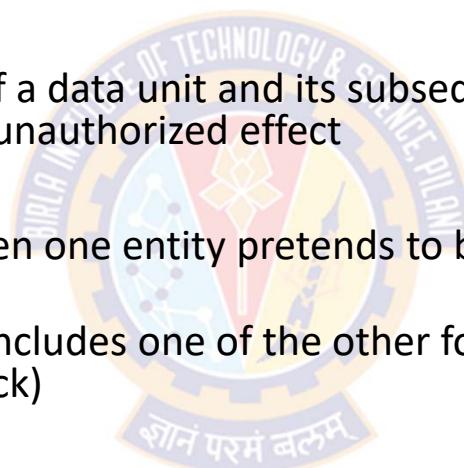
- Active attacks

- Replay

- Involves the passive capture of a data unit and its subsequent retransmission to produce an unauthorized effect

- Masquerade

- A masquerade takes place when one entity pretends to be a different entity
 - A masquerade attack usually includes one of the other forms of active attack (E.g., Replay attack)
 - For example:
 - Authentication sequences are captured
 - After a valid authentication sequence has taken place, it is replayed, thus enabling an authorized entity with few privileges to obtain extra privileges by impersonating an entity that has those privileges





Threats & Assets

Communication Lines and Networks

- Active attacks
 - Modification of messages
 - It means that some portion of a legitimate message is altered, or that messages are delayed or reordered, to produce an unauthorized effect
 - For example, a message stating, "Allow John Smith to read confidential file accounts" is modified to say, "Allow Fred Brown to read confidential file accounts."
 - The denial of service
 - Prevents or inhibits the normal use or management of communication facilities
 - This attack may have a specific target
 - For example, an entity may suppress all messages directed to a particular destination (e.g., the security audit service)
 - Another form of service denial is the disruption of an entire network, either by disabling the network or by overloading it with messages so as to degrade performance

Communication Lines and Networks

- Active attacks
 - Whereas passive attacks are difficult to detect, measures are available to prevent their success
 - On the other hand, it is quite difficult to prevent active attacks 100%
 - Because to do so would require physical protection of all communication facilities and paths at all times
 - Instead, the goal is to detect them and to recover from any disruption or delays caused by them
 - Because the detection has a deterrent effect, it may also contribute to prevention

Threats & Assets



Threats and Assets

	Availability	Confidentiality	Integrity
Hardware	Equipment is stolen or disabled, thus denying service	An unencrypted USB drive is stolen	
Software	Programs are deleted, denying access to users	An unauthorized copy of software is made	A working program is modified, either to cause it to fail during execution or to cause it to do some unintended task
Data	Files are deleted, denying access to users	An unauthorized read of data is performed An analysis of statistical data reveals underlying data	Existing files are modified or new files are fabricated
Communication Lines and Networks	Messages are destroyed or deleted Communication lines or networks are rendered unavailable	Messages are read The traffic pattern of messages is observed	Messages are modified, delayed, reordered, or duplicated False messages are fabricated

Security Design Principles



References

- Computer Security – Principles and Practice
 - William Stallings & Lawrie Brown
 - Chapter-1 – Computer Security Concepts





Thank You!



BITS Pilani
Pilani | Dubai | Goa | Hyderabad

Cyber Security

Introduction

Dr. Ramakrishna Dantu

Associate Professor, BITS Pilani



Disclaimer and Acknowledgement



- The content for these slides has been obtained from books and various other source on the Internet
- I here by acknowledge all the contributors for their material and inputs.
- I have provided source information wherever necessary
- I have added and modified the content to suit the requirements of the course

Cyber Security - Introduction



Agenda

- Computer Security Concepts
- Threats, Attacks, and Assets
- Security Functional Requirements
- Fundamental Security Design Principles
- Attack Surfaces and Attack Trees
- Computer Security Strategy
- Standards



Security Functional Requirements



Security Functional Requirements



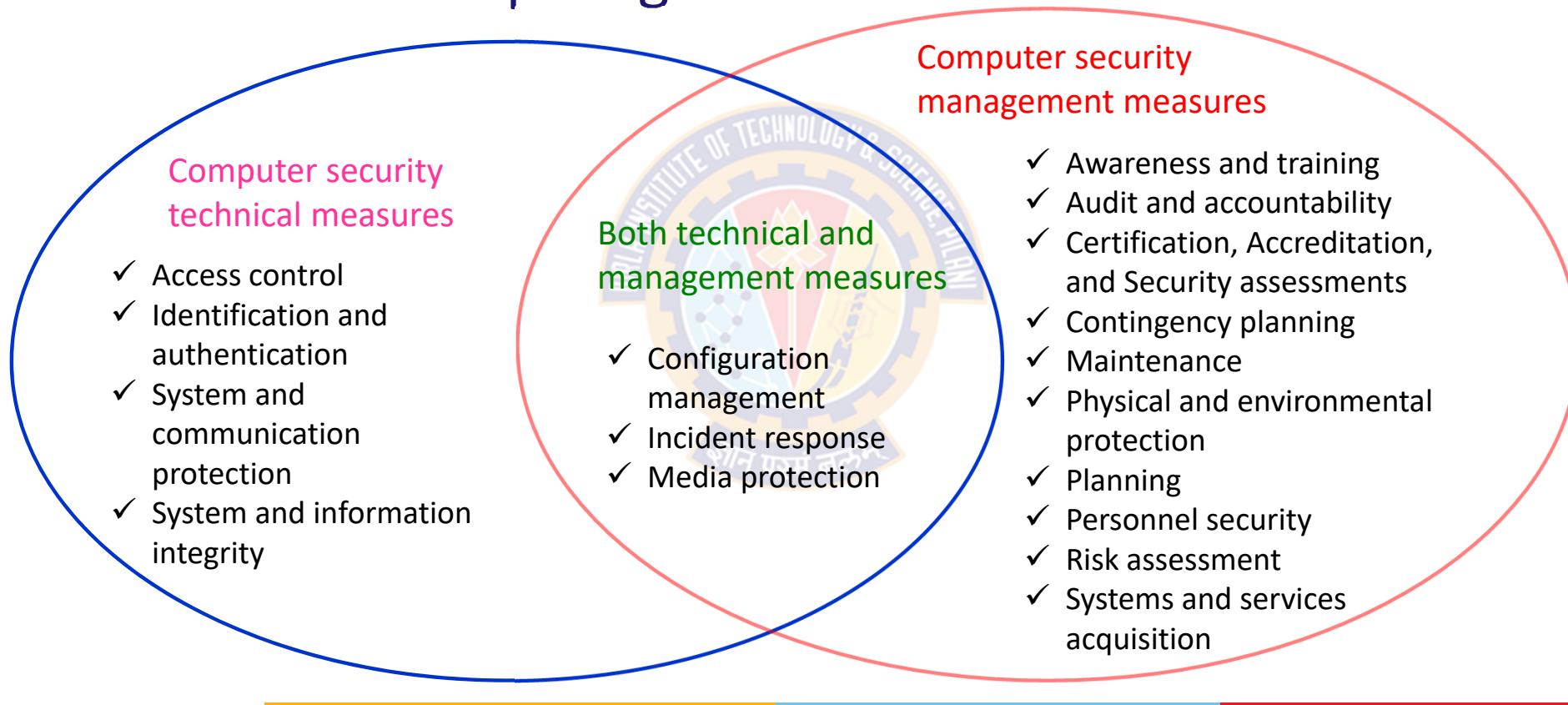
Classifying & Characterizing Countermeasures

- Countermeasures are viewed in terms of functional requirements
- FIPS pub 200 talks about:
 - the Minimum Security Requirements for Federal Information and Information Systems
- FIPS 200 enumerates **17 security areas** with regard to protecting the CIA of
 - the information systems and
 - the information processed, stored, and transmitted by those systems
- The requirements in FIPS 200 can be divided into two categories:
 - Those that require **computer security technical measure**
 - Those that require **management measure**
- <https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.200.pdf>

Security Functional Requirements



Functional Areas Requiring...





Security Functional Requirements

Functional areas involving technical measures

Term	Description
Access Control	<p>Limit IS access to: authorized users, processes acting on behalf of authorized users, other ISs and devices, and to the transactions and functions that authorized users are permitted to exercise</p>
Identification and Authentication	<p>Identify the IS users, processes acting on behalf of users, other ISs and devices, and authenticate (or verify) their identities as a prerequisite to allowing access to OISs</p>
System and Communication Protection	<p>(i) Monitor, control, and protect OCs (i.e., information transmitted or received by OISs) at the external boundaries and key internal boundaries of the ISs; and (ii) Employ architectural designs, software development techniques, and systems engineering principles that promote effective information security within OISs</p>
System and Information Integrity	<p>(i) Identify, report, and correct the flaws in information and ISs in a timely manner; (ii) Provide protection from malicious code at appropriate locations within OISs; and (iii) Monitor IS security alerts and advisories and take appropriate actions in response.</p>



Security Functional Requirements

Functional areas involving managerial measures

Term	Description
Awareness and Training	(i) Ensure that managers and users of OISs are aware of the security risks associated with their activities and of the applicable laws, regulation, and policies related to the security of OISs (ii) Ensure that personnel are adequately trained to carry out their assigned information security-related duties and responsibilities.
Audit and Accountability	(i) Create, protect, and retain IS audit records to the extent needed to enable the monitoring, analysis, investigation, and reporting of unlawful, unauthorized, or inappropriate IS activity (ii) Ensure that the actions of individual IS users can be uniquely traced to those users so they can be held accountable for their actions.
Certification, Accreditation, and Security Assessments	(i) Periodically assess the security controls in OISs to determine if the controls are effective in their application; (ii) Develop and implement plans of action designed to correct deficiencies and reduce or eliminate vulnerabilities in OISs (iii) Authorize the operation of OISs and any associated IS connections (iv) Monitor IS security controls on an ongoing basis to ensure the continued effectiveness of the controls

Security Functional Requirements



Functional areas involving managerial measures

Term	Description
Contingency Planning	Establish, maintain, and implement plans for emergency response, backup operations, and post-disaster recovery for OISs to ensure the availability of critical information resources and continuity of operations
Maintenance	(i) Perform periodic and timely maintenance on OISs (ii) Provide effective controls on the tools, techniques, mechanisms, and personnel used to conduct information system maintenance
Physical and Environmental Protection	(i) Limit physical access to ISs, equipment, and the respective operating environments to authorized individuals (ii) Protect the physical plant and support infrastructure for ISs (iii) Provide supporting utilities for ISs (iv) Protect ISs against environmental hazards (v) Provide appropriate environmental controls in facilities containing ISs
Planning	Develop, document, periodically update, and implement security plans for OISs that describe the security controls in place or planned for the ISs and the rules of behavior for individuals accessing the ISs

OIS = Organizational Information Systems; IS = Information System;



Security Functional Requirements

Functional areas involving managerial measures

Term	Description
Personnel Security	(i) Ensure that organizational personnel (including third-party service providers) are trustworthy and meet established security criteria for their positions (ii) Ensure that organizational information and ISs are protected during and after personnel actions such as terminations and transfers (iii) Employ formal sanctions for personnel failing to comply with organizational security policies and procedures
Risk Assessment	Periodically assess the risk to organizational operations (including mission, functions, image, or reputation), organizational assets, and individuals, resulting from the operation of OISs and the associated processing, storage, or transmission of organizational information.
Systems and Services Acquisition	(i) Allocate sufficient resources to adequately protect OISs (ii) Employ system development life cycle processes that incorporate information security considerations (iii) Employ software usage and installation restrictions (iv) Ensure that third party providers employ adequate security measures to protect information, applications, and/or services outsourced from the organization



Security Functional Requirements

Functional areas that overlap both

Term	Description
Configuration Management	(i) Establish and maintain baseline configurations and inventories of OISs (including hardware, software, firmware, and documentation) throughout the respective system development life cycles (ii) Establish and enforce security configuration settings for IT products employed in OISs
Incident Response	(i) Establish an operational incident-handling capability for OISs that includes adequate preparation, detection, analysis, containment, recovery, and user-response activities (ii) Track, document, and report incidents to appropriate organizational officials and/or authorities.
Media Protection	(i) Protect IS media, both paper and digital (ii) Limit access to information on IS media to authorized users (iii) Sanitize or destroy IS media before disposal or release for reuse.





Fundamental Security Design Principles

साने परमं बलं

Ordering Pizza



Caller	Google
Is this Pizza Delight?	No sir, it's Google Pizza
I must have dialed a wrong number. Sorry	No sir, Google bought Pizza Delight last month
OK. I would like to order a pizza.	Do you want your usual, sir?
My usual? You know me?	According to our caller ID data sheet, the last 12 times you called you ordered an extra-large pizza with three cheeses, sausage, pepperoni, mushrooms and meatballs on a thick crust.
OK! That's what I want ...	May I suggest that this time you order a pizza with ricotta, arugula, sun-dried tomatoes and olives on a whole wheat gluten-free thin crust?
What? I detest vegetable!	Your cholesterol is not good, sir.
How the hell do you know!	Well, we cross-referenced your home phone number with your medical records. We have the result of your blood tests for the last 7 years.
Okay, but I do not want your rotten vegetable pizza! I already take medication for my cholesterol.	Excuse me sir, but you have not taken your medication regularly. According to our database, you purchased only a box of 30 cholesterol tablets once, at Drug RX Network, 4 months ago.

Ordering Pizza



Caller	Google
I bought more from another drugstore.	That doesn't show on your credit card statement.
I paid in cash.	But you did not withdraw enough cash according to your bank statement.
I have other sources of cash.	That doesn't show on your last tax return unless you bought them using an undeclared income source, which is against the law.
WHAT THE HELL!	I'm sorry, sir, we use such information only with the sole intention of helping you.
Enough already! I'm sick to death of Google, Facebook, Twitter, WhatsApp and all the others. I'm going to an island without internet, cable TV, where there is no cell phone service and no one to watch me or spy on me.	I understand sir, but you need to renew your passport first. It expired 6 weeks ago...

Security Design Principles



Design Principles from NCAE in IA/CD

- NCAE in IA/CD lists the following principles
- Security design principles are meant to guide the development of protection mechanisms



Security Design Principles



Economy of Mechanism

- EoM means that the design of software and hardware security measures should be **as simple and small** as possible
- This is the most difficult principle to honor because there is a constant demand for new features in both hardware and software
- The best that can be done is to keep this principle in mind during system design to try to eliminate unnecessary complexity

Simple and Small Design	Complex Design
Simple mechanisms tend to have fewer exploitable flaws and require less maintenance	More likely to possess exploitable flaws
Makes it easier to test and verify thoroughly	Adversaries may discover and exploit subtle weaknesses that are difficult to spot ahead of time
Configuration management issues are simplified	Configuration management issues become more complex
Updating or replacing a simple mechanism becomes a less intensive process	Updating or replacing a complex mechanism becomes more intensive process

Security Design Principles



Fail-safe Default

- Means that access decisions should be based on **permission** rather than **exclusion**
- That is, **by default no access to all**, and the access is permitted based on the requirement
 - For example, most file access systems, and virtually all protected services on client/server systems work on this principle

Default is lack of access

Involves explicitly giving permission

Exhibits better failure mode than the default permit access approach

Implementation mistake (giving explicit permission) only results in refusing permission, which is a safe situation and can be quickly detected

Default is permit access

Involves explicitly excluding access

Exhibits poor failure mode than the default lack of access

Implementation mistake (explicitly excluding access) results in allowing access, which is an unsafe situation and can long go unnoticed

Security Design Principles



Complete Mediation

- It means that every access must be **checked against the access control mechanism** rather than access decisions retrieved from a cache
- For example:
 - File access systems complies with this principle
 - However, typically, once a user has opened a file, no check is made to see if permissions change
- In a system designed to operate continuously, this principle requires that, if access decisions are remembered for future use, careful consideration be given to how changes in authority are propagated into such local memories
- To fully implement complete mediation, every time a user reads a field or record in a file, or a data item in a database, the system must exercise access control
- This resource-intensive approach is **rarely used**

Security Design Principles



Open Design

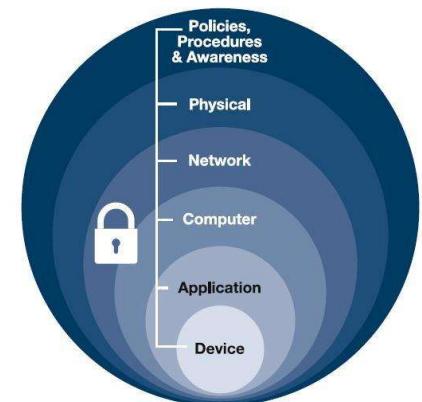
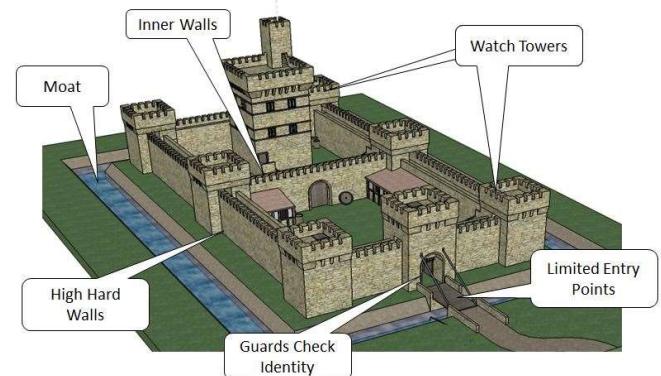
- It means that the design of a security mechanism should be open rather than secret
- For example:
 - although encryption keys must be secret, encryption algorithms should be open to public scrutiny
- The algorithms should be **reviewed by many experts** so that users can have **high confidence** in them
- This is the philosophy behind the NIST program of standardizing encryption and hash algorithms
 - That's why there is a widespread adoption of NIST-approved algorithms

Security Design Principles



Separation of Privilege

- Also known as defense in depth
 - Requires **multiple privilege actions** to achieve access to a restricted resource
- The principle states that a system should not grant permission based on a single condition
- This principle is equivalent to the **separation of duty** principle. For example:
 - Company checks for more than \$100,000 must be signed by two officers of the company
 - If either does not sign, the check is not valid
 - The two conditions are the signatures of both officers

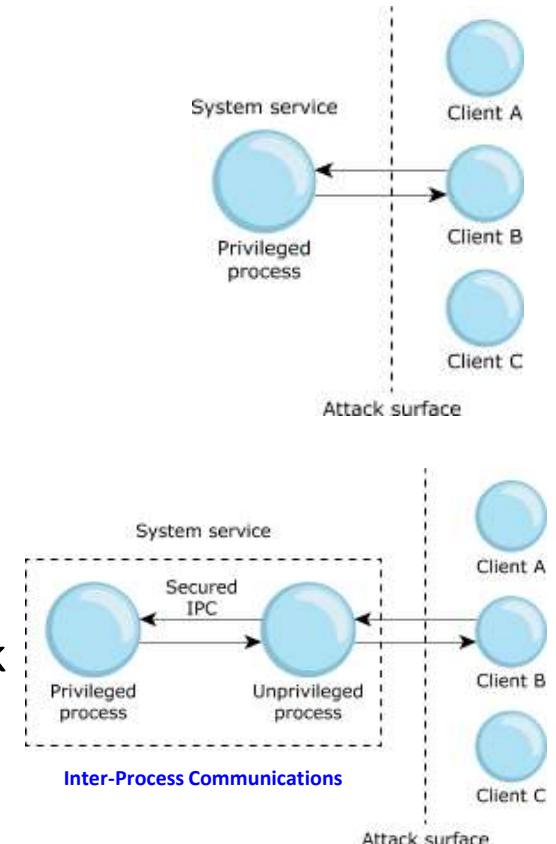


Security Design Principles



Separation of Privilege

- In a software context, a program is divided into multiple parts
- Each part has limited privileges it requires in order to perform a specific task
- For example, the computer program forks into two processes:
 - The main program drops privileges, and the smaller program keeps privileges in order to perform a certain task
 - The two halves then communicate via a socket pair.



Security Design Principles



Least Privilege

- A subject (a user, application, or process) should have only the **minimum necessary privileges** to perform its task, with no additional permissions.
- Example: Role-based privileges
 - The system security policy identifies and define various roles of users or processes
 - Each role is assigned only those permissions needed to perform its functions.
- Each permission specifies certain access to a particular resource:
 - E.g., users may have access to the files on their workstations and a select set of files on a file server, but no access to data that is held within the database
 - E.g., read and write access to a specified file or directory, and connect access to a given host and port
- There is also a temporal aspect to the least privilege principle
 - For example, individuals who have special privileges should have those privileges only for the specific purpose.
 - When they are doing ordinary activities the privileges should be withdrawn.

Security Design Principles



Least Common Mechanism

- This principle states that **mechanisms** used to access resources should not be **shared**
- For example:
 - A program that enables employees to check their payroll information (read) should be separate from a program that modifies the information (write)
- **Covert channels**
 - Covert channel attack creates capability to transfer information between processes that are not supposed to be communicating by the computer security policy.
- Sharing resources **provides a channel** along which information can be transmitted, and so such **sharing should be minimized**
- Solutions using isolation:
 - Virtual machines
 - Sandboxes

Security Design Principles



Least Common Mechanism - Example

- Example
 - A website provides electronic commerce services for a major company.
 - Attackers try to deprive the company of the revenue it obtains from that website
 - They flood the site with messages and tie up the electronic commerce services
 - Legitimate customers are unable to access the website and, as a result, take their business elsewhere.
- Explanation
 - Here, the sharing of the Internet with the attackers' sites caused the attack to succeed
 - The appropriate countermeasure would be to restrict the attackers' access to the segment of the Internet connected to the website
 - Techniques for doing this include proxy servers or traffic throttling
 - Throttling is concerned with limiting traffic coming from legitimate visitors as opposed to dealing with denial-of-service attacks

Security Design Principles



Psychological Acceptability

- Security mechanisms should not add to the difficulty of accessing a resource
 - Simultaneously should meet the needs of those who authorize access
 - E.g., requesting hair samples from the users who have gone completely bald (lost hair) in order to comply with a biometric authentication mechanism
- If security mechanisms hinder the usability or accessibility of resources, users will look for ways to defeat those mechanisms
 - Users write down passwords which are too difficult to remember
 - Authentication for Remote Logins (rlogin): .rhosts mechanism bypasses password security check
 - The .rhosts file contains a list of hosts and user names that determines who can log in to a system remotely without a password.
 - if you set up the /etc/hosts.equiv or .rhosts file, you are not asked for a password, because the network already knows who you are

Security Design Principles



Isolation

- This principle applies in three contexts
- **Restricting public access** to critical resources
 - The system that has critical data, processes, or resources must be isolated such that it restricts public access:
 - Physical isolation:
 - The system with critical information is physically isolated from the system with public access information.
 - Logical isolation:
 - Security services layers are established between the public system and the critical systems.
- Files or data of one user must be kept isolated with the files or data of another user
 - New operating systems have this functionality.
 - Each user operating the system have an isolated memory space, process space, file space along with the mechanism to prevent unwanted access.
- The security mechanisms themselves must be isolated such that they are prevented from unwanted access.
 - E.g., isolating cryptographic software from other parts of the host system so that the software is protected from tampering

Security Design Principles



Encapsulation

- Encapsulation can be viewed as a specific form of isolation based on object-oriented functionality
- Protection is provided by encapsulating a methods and data objects so that the internal structure of a data object is accessible only to the procedures of the protected subsystem
- These procedures can be called only at the designated entry points

Security Design Principles



Modularity

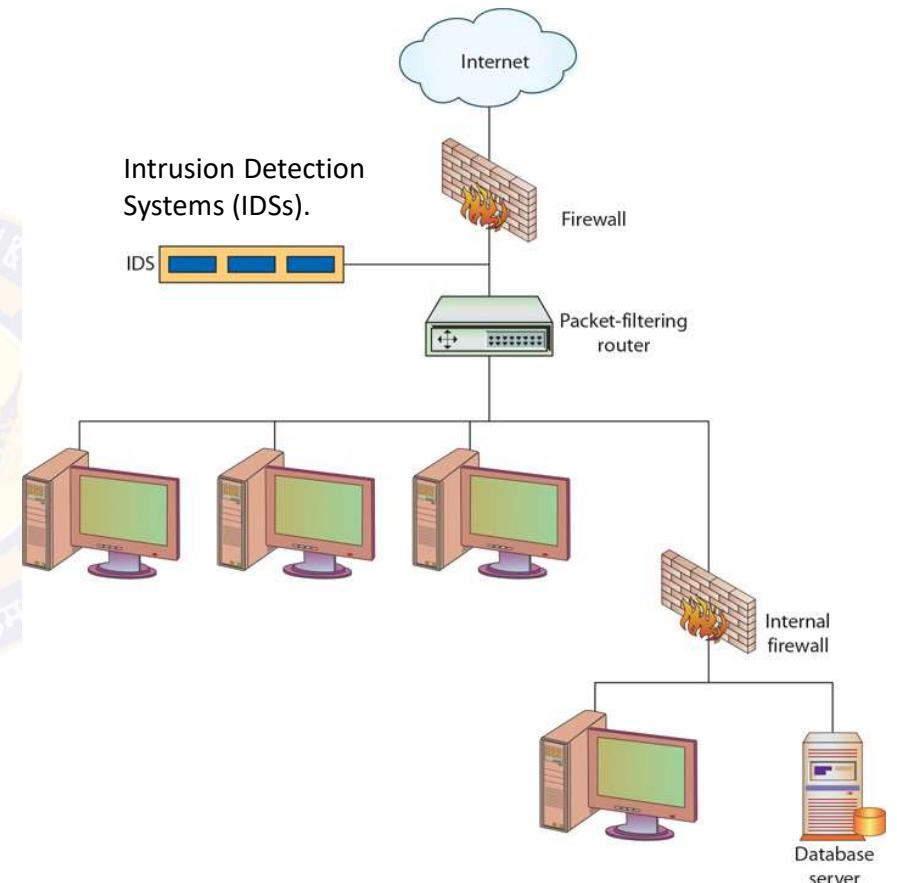
- Modularity principle says that the security mechanism must be developed:
 - as separate and protected modules, and
 - using the modular architecture.
- The design goal here is to provide security functions and services, such as cryptographic functions, as common modules
- For example:
 - numerous protocols and applications make use of cryptographic functions
 - Rather than implementing such functions in each protocol or application, a more secure design is to provide a common cryptographic module that can be invoked by other applications
- This allows us to focus on
 - a) the secure design and implementation of a single cryptographic module, and
 - b) the mechanisms to protect the module from tampering
- The modular structure helps in migrating to new technology or upgrading the features of security mechanism without modifying the entire system

Security Design Principles



Layering

- Similar to defense in depth
- Involves the use of multiple, overlapping protection approaches in a series
- Provides multiple barriers to the adversary if he tries to access the protected system.
- Allows for numerous, different controls to guard against whatever threats come to pass.
- Addresses people, technology, and operational aspects of information systems
- Security breach of any one layer will not leave the system unprotected



Security Design Principles



Least Astonishment

- Security mechanisms should use a model that the users can easily understand
- The security mechanisms should be designed such that using the mechanism is simple
 - Hide complexity introduced by security mechanisms
 - Ease of installation, configuration, and use
- The security mechanism should be such that the user has a good intuitive understanding of how the security goals map to the provided security mechanism
- For example:
 - The program should always respond in the way that is least likely to astonish the user. Such as at the time of login, the system should not ask your SSN
- Configuring and executing a program should be as easy and as intuitive as possible, and any output should be clear, direct, and useful.





Attack Surfaces and Attack Trees

साने परमं बलं

Attack Surfaces and Attack Trees



Attack Surfaces

- An attack surface
 - is the **set of entry points** that attackers can use to compromise a system.
 - consists of **reachable and exploitable** vulnerabilities in a system
- Keeping the attack surface as small as possible is a basic security measure
- For example:
 - Open ports on outward facing Web and other servers, and code listening on those ports
 - Services that are available on the inside of a firewall
 - Code that processes incoming data, email, XML, office documents, and industry-specific custom data exchange formats
 - Interfaces, SQL, and Web forms
 - An employee with access to sensitive information vulnerable to a social engineering attack

Attack Surfaces and Attack Trees



Attack Surfaces

- Categories of Attack surfaces:

- Network attack surface

- Refers to vulnerabilities over an [LANs](#), [WANs](#), or the [Internet](#)
 - Includes [network protocol vulnerabilities](#), such as those used for a denial-of-service attack, disruption of communications links, and various forms of intruder attacks.

- Software attack surface

- Refers to vulnerabilities in [application](#), utility, or operating system code
 - A particular focus in this category is [Web server software](#)

- Human attack surface

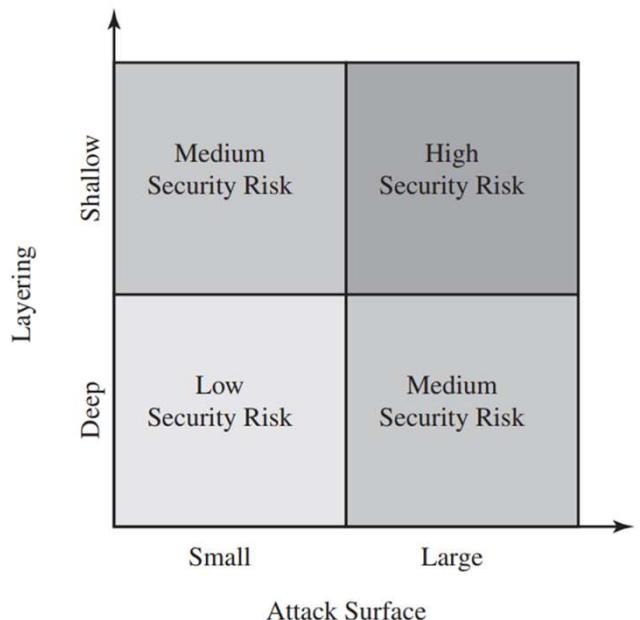
- Refers to vulnerabilities created by [employees](#) or [outsiders](#)
 - Includes, social engineering, human error, and trusted insiders

Attack Surfaces and Attack Trees



Attack Surface Analysis

- Is a useful technique for assessing the **scale and severity** of threats to a system
- A systematic analysis of vulnerable points makes security analysts aware of where security mechanisms are required
- Once an **attack surface is defined**, designers may be able to find ways to make the surface smaller, thus making the task of the adversary more difficult
- It provides guidance on setting priorities for testing, strengthening security measures, or modifying the service or application
- The use of layering (or defense in depth), and attack surface reduction complement each other in mitigating security risk

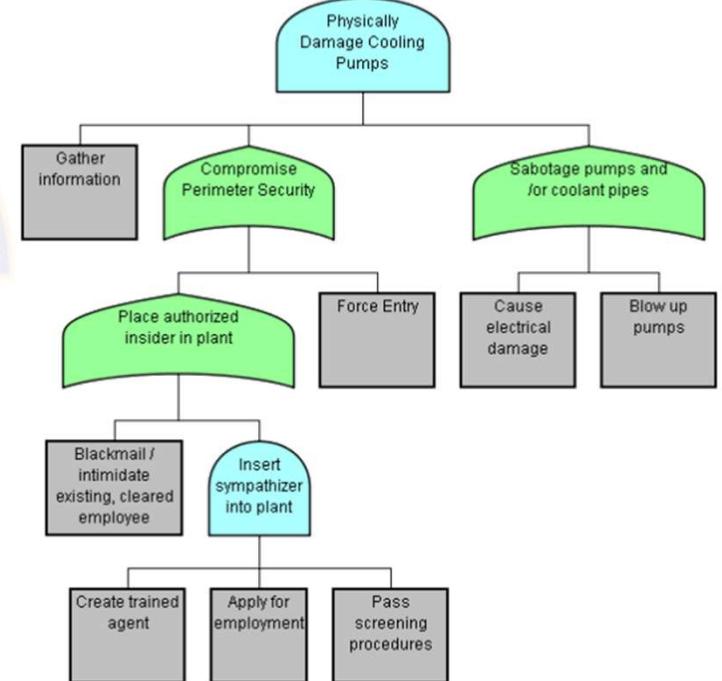
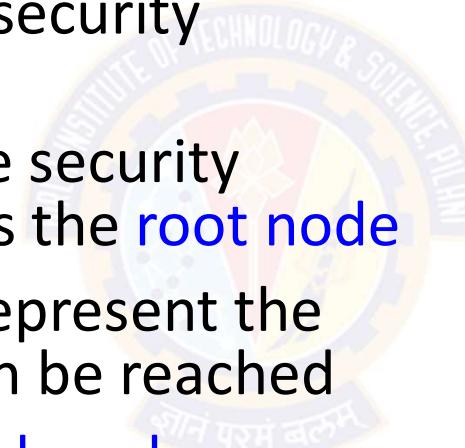


Attack Surfaces and Attack Trees



Attack Trees

- An attack tree shows a set of potential techniques for exploiting security vulnerabilities
- The goal of the attack (the security incident) is represented as the root node
- Branches and subnodes represent the ways in which the goal can be reached
- Each subnode defines a subgoal
 - Each subgoal may have its own set of further subgoals, etc.

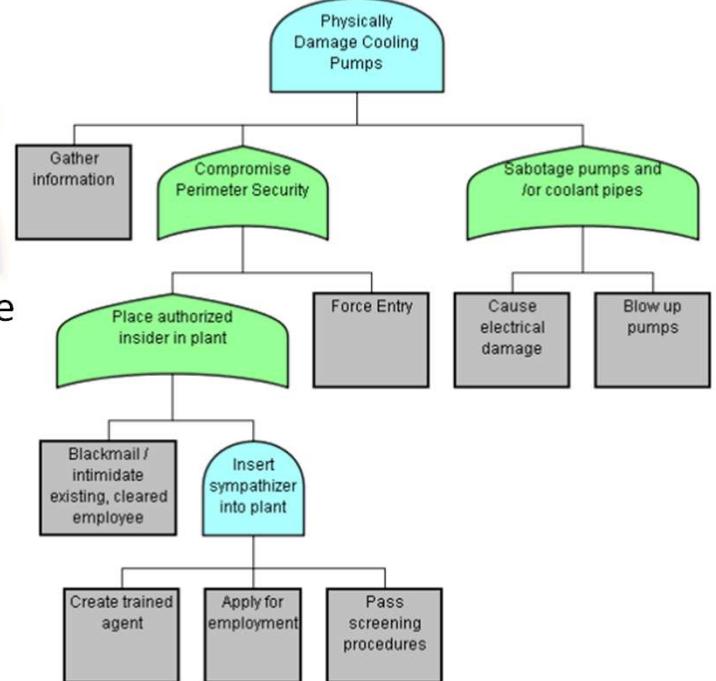
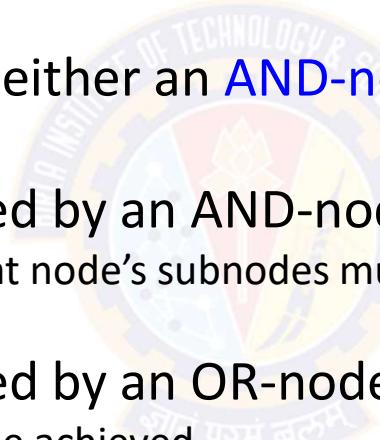


Attack Surfaces and Attack Trees



Attack Trees

- The **leaf nodes** represent different ways to initiate an attack
- Each node other than a leaf is either an **AND-node** or an **OR-node**
- To achieve the goal represented by an AND-node,
 - all the subgoals represented by that node's subnodes must be achieved
- To achieve the goal represented by an OR-node,
 - at least one of the subgoals must be achieved
- Branches can be labeled with values representing difficulty, cost, or other attack attributes, so that alternative attacks can be compared

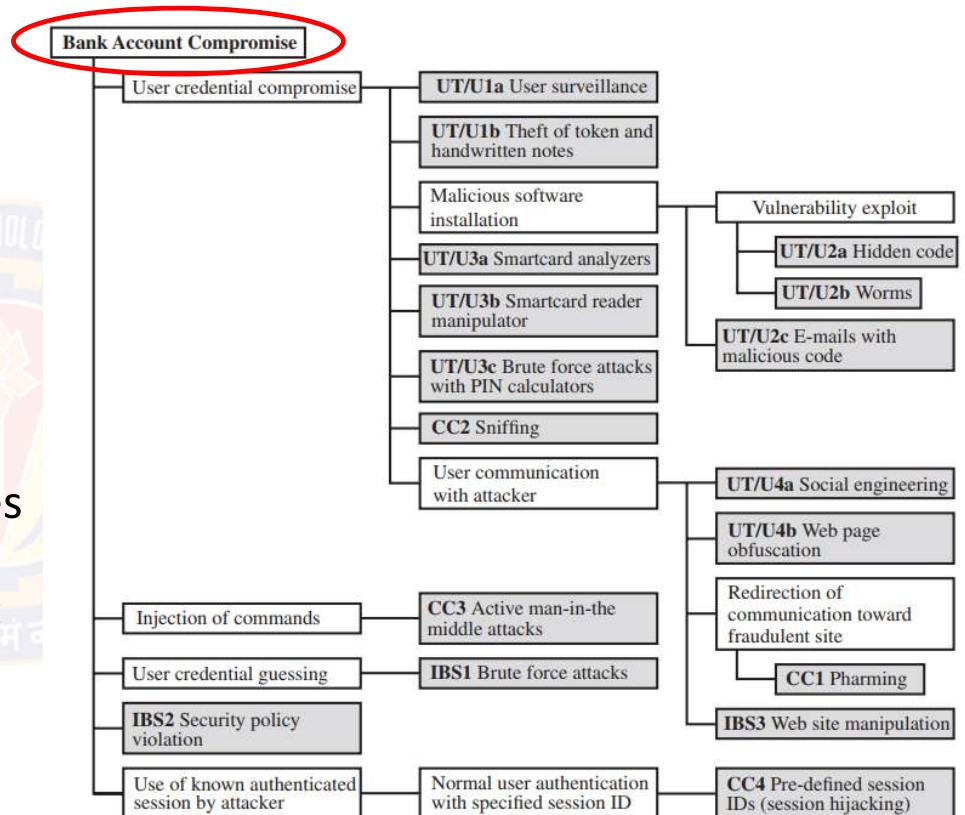


Attack Surfaces and Attack Trees



Attack Trees – Example

- The **goal** of the attacker is to **compromise a user's bank account**
- The shaded boxes (**leaf nodes**) represent the **attack events**
- The **white boxes** are categories which consist of one or more specific attack events (leaf nodes)
- In this tree, all the nodes other than leaf nodes are **OR-nodes**
- Three components involved in authentication:
 - User terminal and user (UT/U)
 - Communications channel (CC)
 - Internet banking server (IBS)



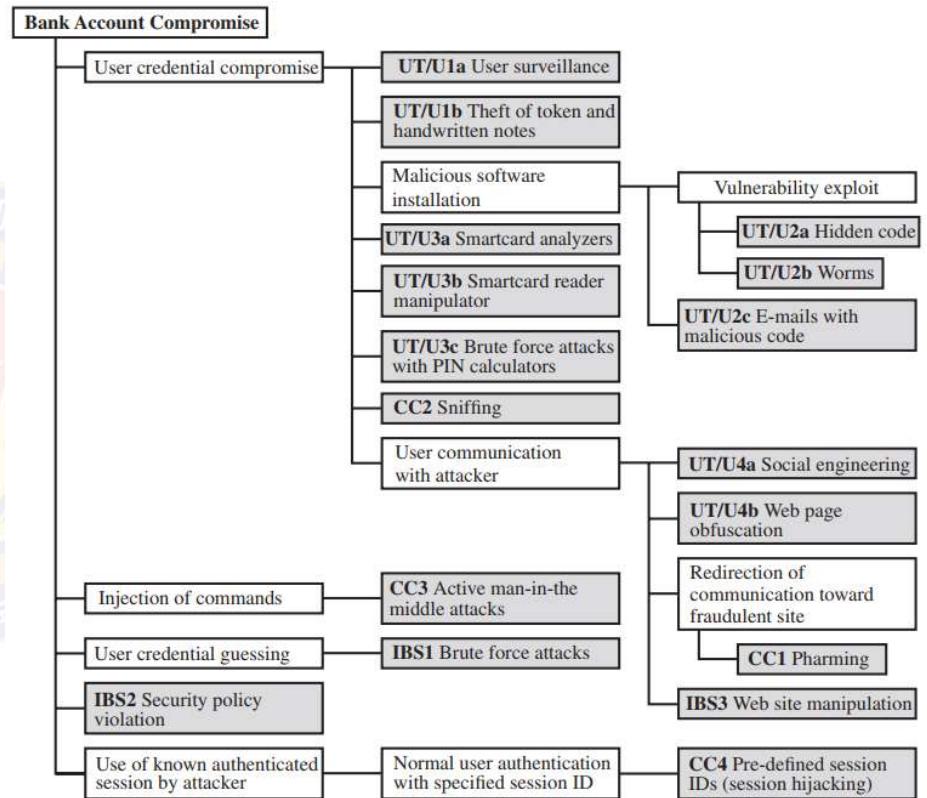
An Attack Tree for Internet Banking Authentication

Attack Surfaces and Attack Trees



Attack Trees – Example

- User terminal and user (UT/U):
 - These attacks target the user equipment, including the tokens such as smartcards or other password generators, as well as the actions of the user
- Communications channel (CC):
 - This type of attack focuses on communication links
- Internet banking server (IBS):
 - These types of attacks target the servers that host the Internet banking application



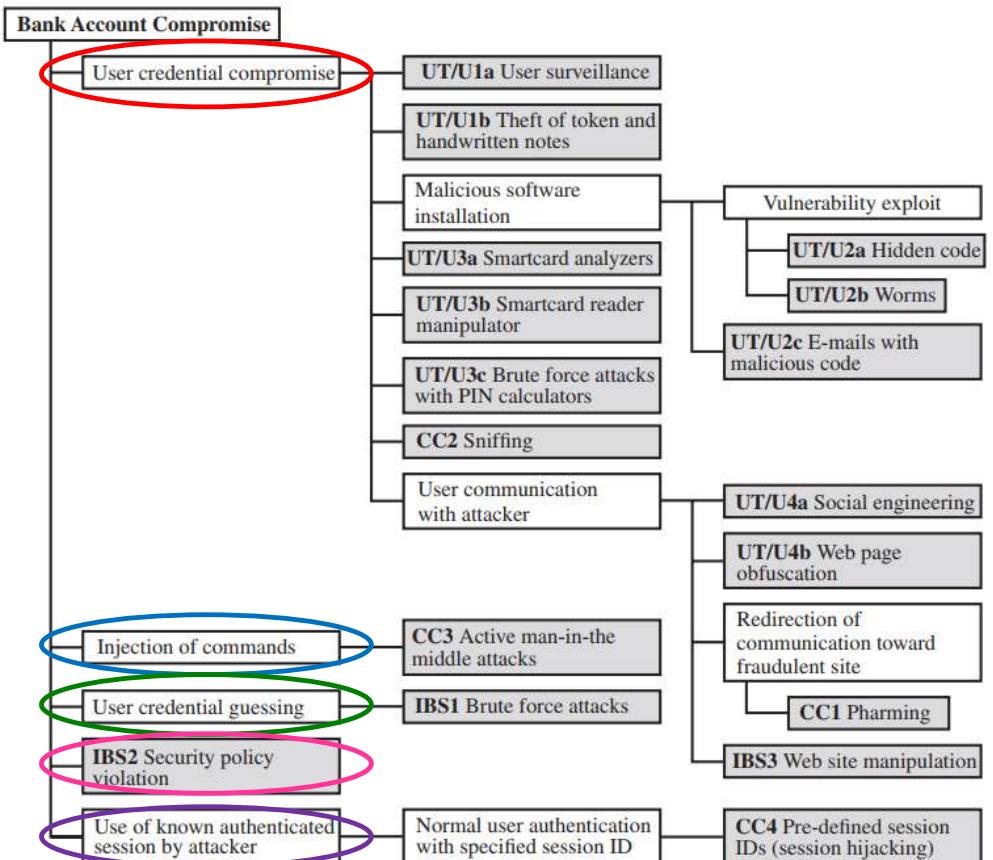
Attack Surfaces and Attack Trees



Attack Trees – Example

• Attack Strategies

- Five attack strategies can be identified, each of which exploits one or more of the three components
 - User credential compromise
 - Injection of commands
 - User credential guessing
 - IBS Security policy violation
 - Use of known authenticated session

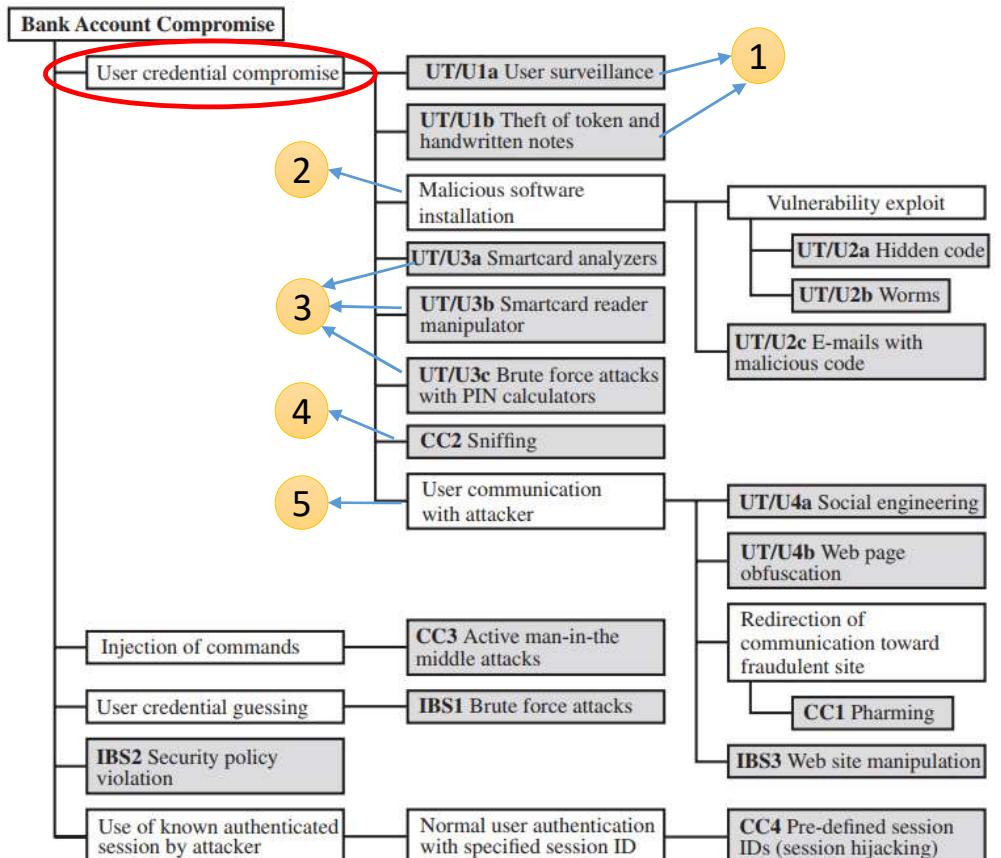


Attack Surfaces and Attack Trees



Attack Trees – Example

- User credential compromise
 - This strategy can be used against many elements of the attack surface
 - One by using procedural attacks
 - Monitoring a user's action to observe a PIN or other credential
 - Theft of the user's token or handwritten notes
 - Two
 - Embedding malicious software to compromise the user's login and password
 - Three by using token attack tools
 - Hacking the smartcard
 - Using a brute force approach to guess the PIN
 - Four
 - Obtaining credential information via the communication channel (sniffing)
 - Five
 - Engaging in communication with the target user

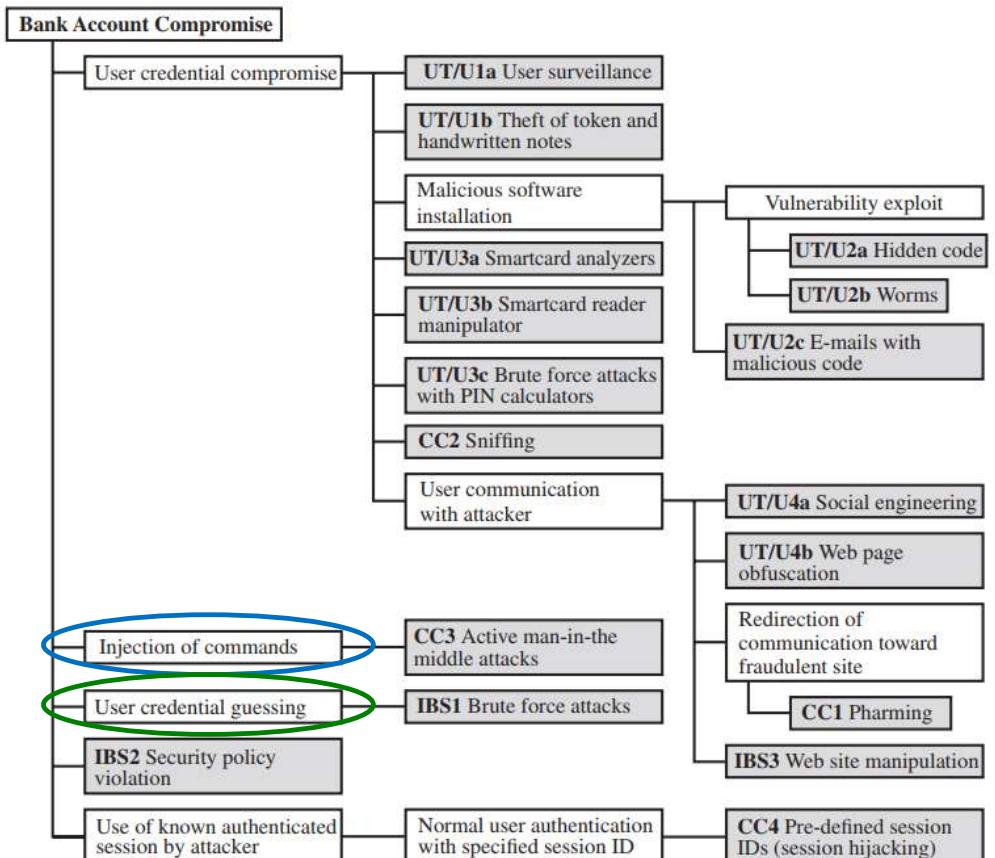


Attack Surfaces and Attack Trees



Attack Trees – Example

- **Injection of commands**
 - Involves **intercepting communication** between the UT and the IBS
 - Involves **impersonating** the valid user to gain access to the banking system.
- **User credential guessing**
 - Involves **brute force attacks** against banking authentication schemes by
 - sending random usernames and passwords
 - The attack mechanism can be by using
 - **distributed zombie personal computers**,
 - **hosting automated programs** for username- or password-based calculation

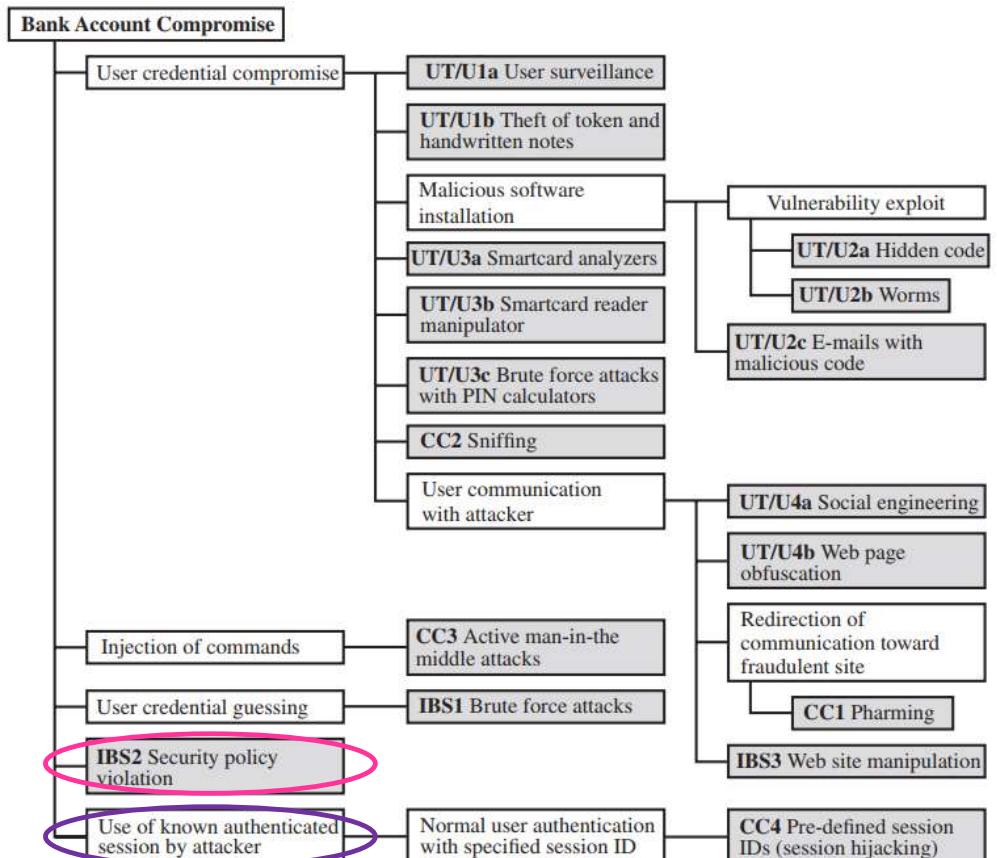


Attack Surfaces and Attack Trees



Attack Trees – Example

- Security policy violation
 - An employee may expose a customer's account by
 - Sharing passwords
 - Using weak access control and logging mechanisms
- Use of known authenticated session
 - Persuading or forcing the user to connect to the IBS with a preset session ID
 - Once the user authenticates to the server, the attacker may utilize the known session ID to send packets to the IBS, spoofing the user's identity



Attack Surfaces and Attack Trees



Attack Trees

- Attack trees are used to effectively exploit the information available on attack patterns
- Organizations such as CERT developed body of knowledge about both general attack strategies and specific attack patterns
- These organizations publish security advisories
- Security analysts can use the attack tree to document security attacks in a structured form that reveals key vulnerabilities
- The attack tree can guide both the design of systems and applications, and the choice and strength of countermeasures.



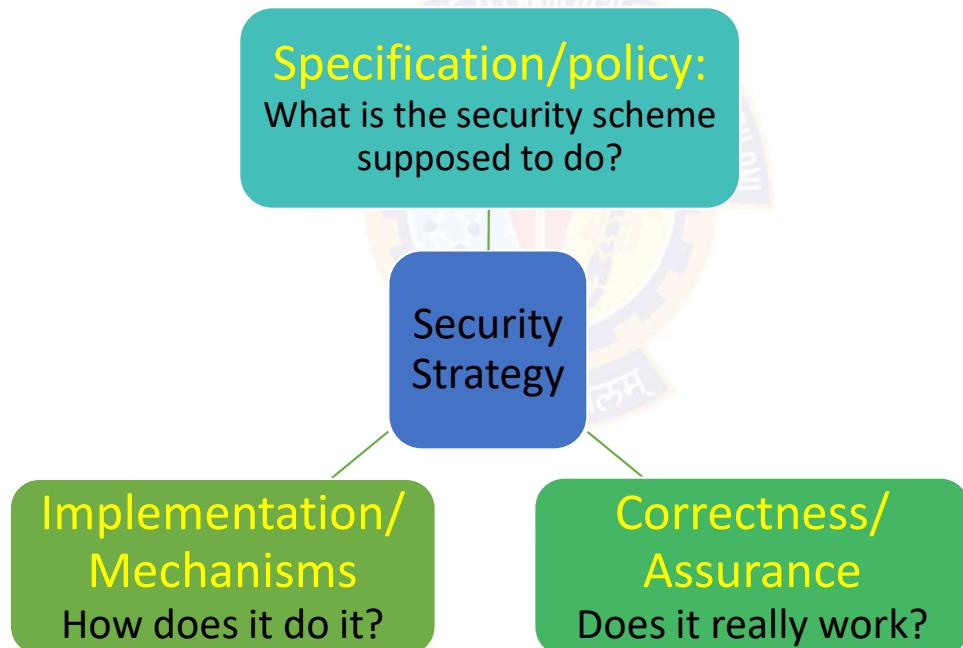
Computer Security Strategy

Computer Security Strategy



Comprehensive Security Strategy

- A comprehensive security strategy involves three aspects:



Computer Security Strategy



Security Policy

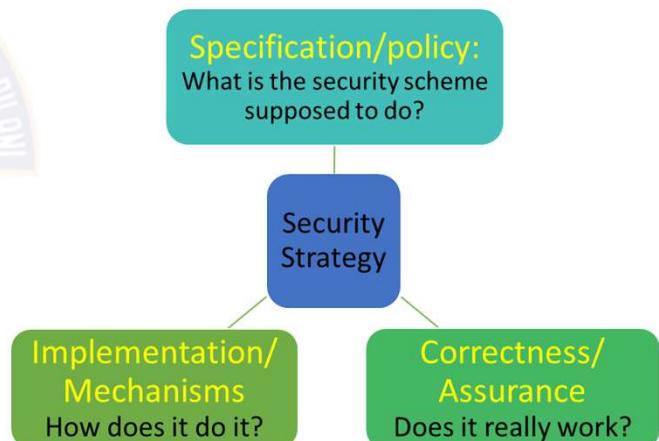
- Developing a security policy is the first step in devising security services and mechanisms
- A security policy
 - Is a **statement of rules and practices** that specify or regulate how a system or organization provides security services to protect sensitive and critical system resources
 - Describes the desired system behavior
 - Includes the requirements for **confidentiality, integrity, and availability**
 - Formal security policies are **enforced** by the system's **technical controls** as well as its **management and operational controls**

Computer Security Strategy



Security Policy

- In developing a security policy, a security manager needs to consider the following factors and tradeoffs:
 - Factors
 - The **value of the assets** being protected
 - The **vulnerabilities** of the system
 - Potential threats and the **likelihood of attacks**
 - Trade-offs
 - Ease of use versus **security**
 - Cost of security versus cost of failure and recovery

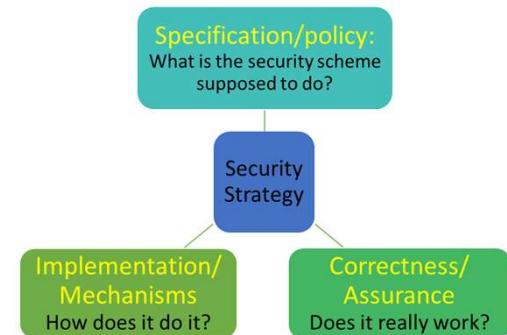
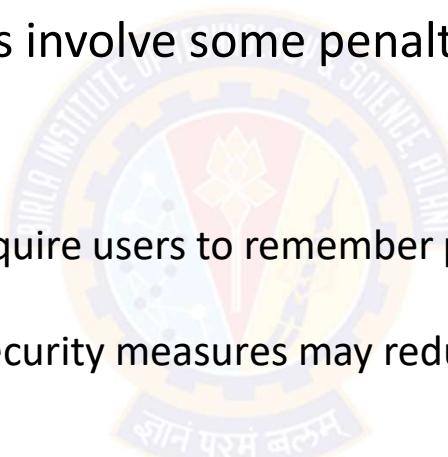


Computer Security Strategy



Security Policy – Trade-offs

- Ease of use versus security
 - Virtually all security measures involve some penalty in the area of ease of use
 - For example:
 - Access control mechanisms require users to remember passwords and perhaps perform other access control actions
 - Firewalls and other network security measures may reduce available transmission capacity or slow response time
 - Virus-checking software
 - reduces available processing power and
 - introduces the possibility of system crashes or malfunctions due to improper interaction between the security software and the operating system

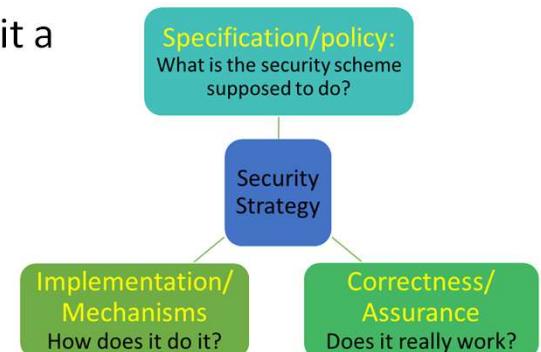
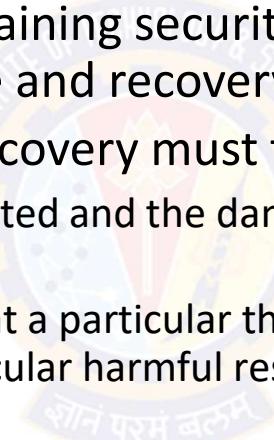


Computer Security Strategy



Security Policy – Trade-offs

- Cost of security versus cost of failure and recovery
 - Costs of implementing and maintaining security measures must be balanced against the cost of security failure and recovery
 - The cost of security failure and recovery must take into account:
 - the value of the assets being protected and the damages resulting from a security violation
 - the risk, which is the probability that a particular threat will exploit a particular vulnerability with a particular harmful result



Computer Security Strategy



Security Implementation

- Security implementation involves four complementary courses of action:
 - Prevention
 - Detection
 - Response
 - Recovery



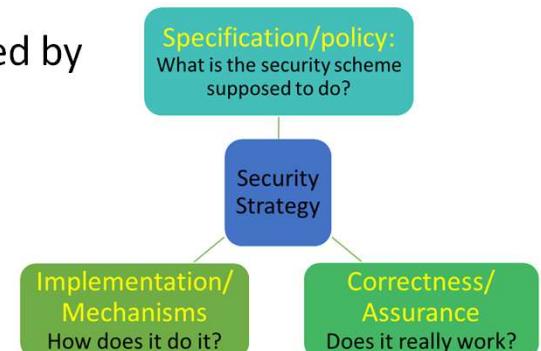
Computer Security Strategy



Security Implementation

- Prevention

- An ideal security scheme is one in which no attack is successful, which is impractical
- There is a wide range of threats in which prevention is a reasonable goal
- Example: Transmission of encrypted data
 - Attacks on confidentiality of the transmitted data can be prevented by
 - using secure encryption algorithm and
 - taking measures to prevent unauthorized access to encryption keys



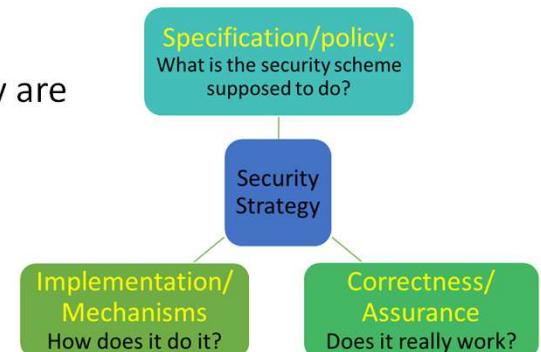
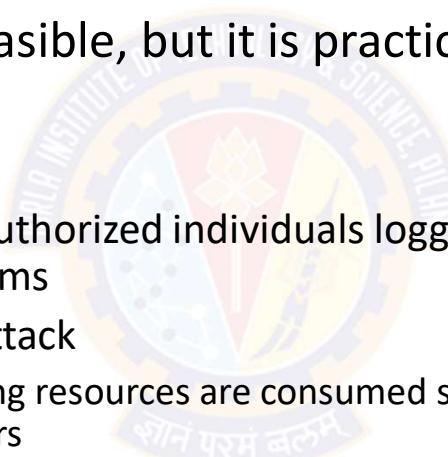
Computer Security Strategy



Security Implementation

- **Detection**

- Absolute prevention is not feasible, but it is practical to detect security attacks
- For example:
 - Detecting the presence of unauthorized individuals logged into a system using intrusion detection systems
 - Detecting a denial of service attack
 - Communications or processing resources are consumed so that they are unavailable to legitimate users



Computer Security Strategy



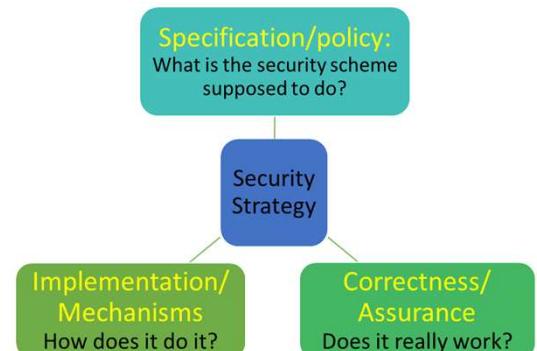
Security Implementation

- Response:

- Once an attack (E.g., denial of service) is detected, the system can respond by halting the attack and preventing further damage

- Recovery:

- Assets (E.g., data) can be recovered using backup systems
- If data integrity is compromised, a prior, correct copy of the data can be reloaded

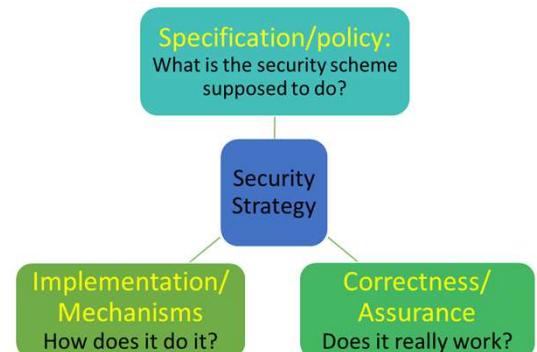
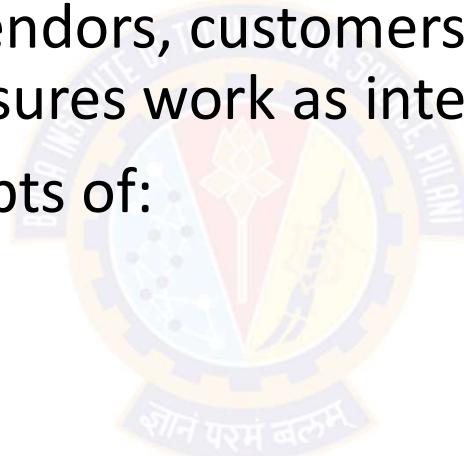


Computer Security Strategy



Assurance and Evaluation

- The "consumers" of computer security services and mechanisms (e.g., system managers, vendors, customers, and end users) want to feel that the security measures work as intended
- This bring us to the concepts of:
 - Assurance and Evaluation.



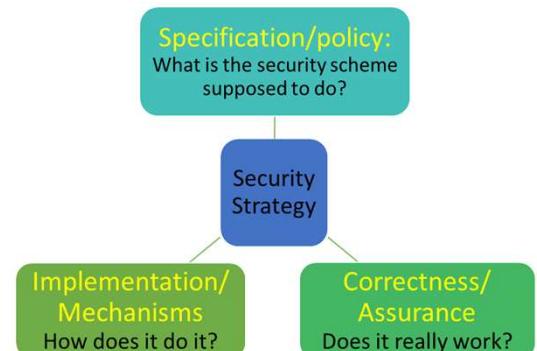
Computer Security Strategy



Assurance and Evaluation

- Assurance

- "The **degree of confidence** one has that the security measures, both technical and operational, work as intended to protect the system and the information it processes."
-- NIST95
- This encompasses both system design and system implementation
- Assurance deals with the questions such as:
 - "Does the security system design meet its requirements?"
 - "Does the security system implementation meet its specifications?"
- Note:
 - Assurance is expressed as a **degree of confidence**, not in terms of a **formal proof** that a design or implementation is correct
 - It is **not possible to provide absolute proof** that designs and implementations are correct



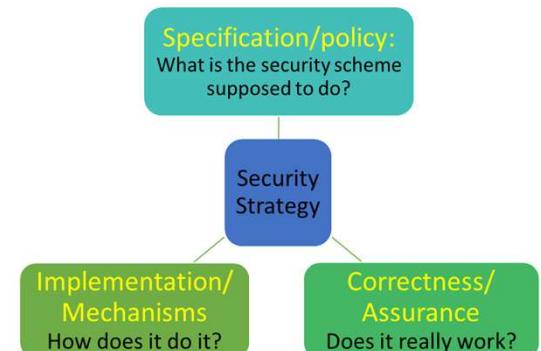
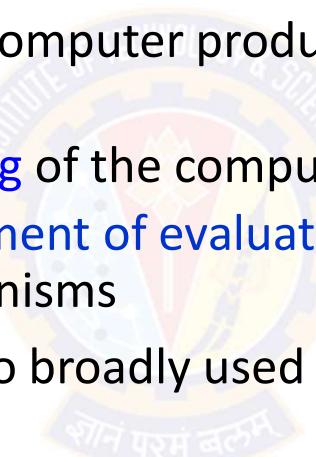
Computer Security Strategy



Assurance and Evaluation

- **Evaluation**

- It is the **process of examining** a computer product or system with respect to certain criteria
- Evaluation involves formal **testing** of the computer product and process
- The core work involves **development of evaluation** criteria that can be applied to any security services and mechanisms
- These evaluation criteria can also broadly used for making product comparisons





Thank You!



BITS Pilani
Pilani | Dubai | Goa | Hyderabad

Cyber Security

Security Architecture: Policies, Models and Mechanisms

Dr. Ramakrishna Dantu

Associate Professor, BITS Pilani



Disclaimer and Acknowledgement



- The content for these slides has been obtained from books and various other source on the Internet
- I here by acknowledge all the contributors for their material and inputs.
- I have provided source information wherever necessary
- I have added and modified the content to suit the requirements of the course

Security Architecture: Policies, Models and Mechanisms



Agenda

- Introduction to security policies, models and mechanisms
- The Nature of Security Policies
- Types of Security Policies
- The Role of Trust
- Types of Access Control
- Policy Languages
- The CIA Classification:
 - Confidentiality Policies:
 - Integrity Policies:
 - Availability Policies:





The Nature of Security Policies

१०८ परमं बलू०

The Nature of Security Policies



Terms

- Security Policy
- Secure System
- Breach of Security
 - Confidentiality, Integrity, and Availability
- Security Mechanism
- Policy Model



The Nature of Security Policies



Overview

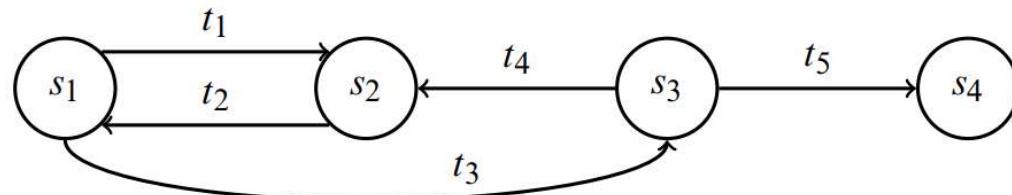
- Consider a computer system to be a **finite-state automaton** with a set of transition functions that change state, then:
- Definition:
 - A *security policy* is a statement that partitions the states of the system into a set of *authorized* (or *secure*), states and a set of *unauthorized* (or *non-secure*), states
- A security policy sets the context to define a *secure system*
- What is secure under one policy may not be secure under a different policy
- Definition:
 - A *secure system* is a system that starts in an authorized state and **cannot** enter an unauthorized state

The Nature of Security Policies



Overview

- Consider the finite-state machine. It consists of four states and five transitions



- The security policy partitions the states into a set of **authorized** states $A = \{s_1, s_2\}$ and a set of **unauthorized** states $UA = \{s_3, s_4\}$
- This system is not secure because regardless of which authorized state it starts in, **it can enter an unauthorized state**
- However, if the edge from s_1 to s_3 were not present, the system would be secure, because it could not enter an unauthorized state from an authorized state
- Definition:**
 - A *breach of security* occurs when a system enters an **unauthorized state**.

The Nature of Security Policies



Confidentiality

- Definition:
 - Let X be a set of entities and let I be some information
 - Then I has the property of *confidentiality* with respect to X if no member of X can obtain information about I
- Confidentiality implies that information:
 - must not be disclosed to some set of entities
 - it may be disclosed to others
- The membership of set X is often implicit (understood)
 - For example, when we speak of a document that is confidential,
 - all entities not authorized to have such access make up the set X

The Nature of Security Policies



Integrity

- Definition:
 - Let X be a set of entities and let I be some information or resource
 - Then I has the property of *integrity* with respect to X if all members of X trust I
- In addition, members of X also trust that the *transmission* and *storage* of I do not change the information or its trustworthiness
 - This aspect is sometimes called *data integrity*
- If I is information about the origin of something, or about an identity, the members of X trust that the information is correct and unchanged
 - This aspect is sometimes called *origin integrity* or, *authentication*
- If I is a resource (E.g., database or application), then integrity means that the resource functions correctly (meeting its specifications)
 - This aspect is called *assurance*

The Nature of Security Policies



Availability

- Definition
 - Let X be a set of entities and let I be a resource
 - Then I has the property of *availability* with respect to X if **all members of X can access I**
- The exact definition of "access" varies depending on:
 - the needs of the members of X ,
 - the nature of the resource, and
 - the use to which the resource is put
- Example:
 - If a book-selling server takes up to 20 minutes to service a book purchase request, that may meet the client's requirements for "availability."
 - If a server of medical information takes up to 10 minutes to provide allergy information of a patient to an anesthetic, that will not meet an emergency room's requirements for "availability."



The Nature of Security Policies

Confidentiality Policy

- With respect to **confidentiality**,
 - a security policy identifies the states in which information leaks to those who are not authorized to receive it
 - This includes the **leakage of rights** and the **illicit transmission** of information without leakage of rights, called **information flow**
- Also, the policy must handle changes of authorization, so it includes a temporal element
- For example:
 - A contractor working for a company may be authorized to access proprietary information during the lifetime of a nondisclosure agreement, but when that nondisclosure agreement expires, the contractor can no longer access that information
- This aspect of the security policy is often called a ***confidentiality policy***

The Nature of Security Policies



Integrity Policy

- With respect to integrity,
 - a security policy identifies **authorized ways** in which information may be **altered** and **entities** authorized to **alter** it
- Authorization may derive from a variety of relationships, and external influences may constrain it
- For example:
 - In many transactions, a principle called **separation of duties** forbids an entity from completing the transaction on its own
- Those parts of the security policy that describe the conditions and manner in which data can be altered are called the **integrity policy**

The Nature of Security Policies



Availability Policy

- With respect to availability,
 - a security policy describes the availability details of various services
- It may present parameters within which the services will be accessible. For example:
 - A browser may download web pages but not Java applets
- It may describe a level of service. For example
 - A server will provide authentication data within 1 minute of the request being made
- Those parts of the security policy that
 - discusses the conditions and manner in which systems and services must be available is called the *Availability policy*



The Nature of Security Policies

Desired Properties of the System

- Typically, the security policy assumes that the reader understands the context in which the policy is issued:
 - in particular, the laws, organizational policies, and other environmental factors
- The security policy then describes conduct, actions, and authorizations defining "authorized users" and "authorized use."
- EXAMPLE
 - A university disallows cheating, which is defined to include copying another student's homework assignment (with or without permission)
 - A computer science class requires the students to do their homework on the department's computer
 - Student A notices that student B has not read-protected the file containing her homework and copies it
 - Has either student (or have both students) breached security?

The Nature of Security Policies



Desired Properties of the System

- Student B
 - The student has not breached security, despite her failure to protect her homework
 - The security policy requires no action to prevent files from being read
 - She may have been too trusting, but the policy does not ban this
 - Thus, student B has not breached security
- Student A
 - The student has breached security
 - The security policy disallows the copying of homework, and the student has done exactly that
- Whether the security policy specifically states that:
 - "files containing homework shall not be copied" or simply says that
 - "users are bound by the rules of the university"is irrelevant
- If the security policy is silent on such matters, the most reasonable interpretation is that the **policy disallows actions that the university disallows**, because
 - the computer science department is part of the university

The Nature of Security Policies



Security Mechanism

- Definition:
 - A *security mechanism* is an entity or procedure that **enforces** some part of the security policy
- Example
 - In the preceding example, the policy is the statement that no student may copy another student's homework
 - One mechanism is the **file access controls**
 - If the student B had set permissions to prevent the student A from reading the file containing her homework, then A could not have copied that file



The Nature of Security Policies

Procedural or Operational Security Mechanisms - Example

- A site's security policy states that **information** relating to a **particular product** is **proprietary** and is **not to leave** the control of the company
- The company stores its backup tapes in a vault in the town's bank
- The company must ensure that only authorized employees have access to the backup tapes even when the tapes are stored off-site
- The bank's controls on access to the vault, and the procedures used to transport the tapes to and from the bank, are considered **security mechanisms**
- These mechanisms are not technical controls built into the computer
- **Procedural**, or **operational**, controls also can be **security mechanisms**



The Nature of Security Policies

Security Mechanism - Example

- The UNIX operating system, initially developed for a small research group, had mechanisms sufficient to prevent users from accidentally damaging one another's files
 - For e.g., the user A could not delete the user B's files (unless B had set the files and the containing directories to allow this)
- The **implied security policy** for this "friendly" environment was
 - "do not delete or corrupt another's files, and any file not protected may be read."
- When the UNIX operating system moved into academic, commercial, and government environments, the previous **security policy became inadequate**
 - For e.g., some files had to be protected from individual users (rather than from groups of users)
- Similarly, the **security mechanisms were inadequate** for those environments



Types of Security Policies

१०८ परमं बलूः



Types of Security Policies

Policy Model

- Each site has its own requirements for the levels of confidentiality, integrity, and availability
 - The site security policy states these needs for that particular site
- Types of Security Policies
 - Military (or governmental) Security Policy
 - Policy primarily protecting confidentiality
 - Commercial Security Policy
 - Policy primarily protecting integrity
 - Transaction-oriented integrity security policy
 - Confidentiality Policy
 - Policy protecting only confidentiality
 - Integrity Policy
 - Policy protecting only integrity

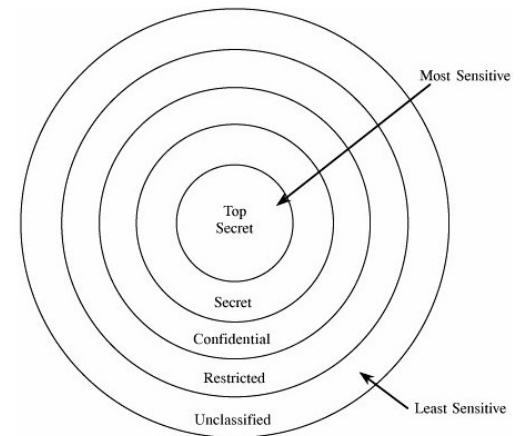




Types of Security Policies

Military Security Policy

- A **military security policy** (or a **governmental security policy**) is concerned with protecting **classified information**
 - It is a security policy developed primarily to provide **confidentiality**
 - Each piece of information is ranked at a particular sensitivity level,
 - such as **unclassified, restricted, confidential, secret, or top secret**.
 - The name comes from the military's need to keep information secret, such as the date that a troop ship will sail
-
- Although integrity and availability are important, organizations using this class of policies can overcome the loss of either
 - For example, they can use orders not sent through a computer network
 - But the **compromise of confidentiality would be catastrophic**, because an opponent would be able to plan countermeasures



Hierarchy of Sensitivities.



Types of Security Policies

Commercial Security Policy

- A *commercial security policy* is a security policy developed primarily to provide integrity
- The name comes from the need of commercial firms to prevent tampering with their data, because they could not survive such compromises
- For example:
 - If the confidentiality of a bank's computer is compromised, a customer's account balance may be revealed
 - This would certainly embarrass the bank and possibly cause the customer to take her business elsewhere
 - But the loss to the bank's "bottom line" would be minor
- However, if the integrity of the computer holding the accounts were compromised, the balances in the customers' accounts could be altered
 - This can lead to financially ruinous effects on the bank

Types of Security Policies



Commercial Security Policy

- Some integrity policies use the notion of a **transaction**
 - E.g., a database transaction must not leave the database in an inconsistent state
- Like database specifications, they require that actions occur in such a way as to leave the database in a **consistent state**
- These policies, called ***transaction-oriented integrity security policies***, are critical to organizations that require consistency of databases.



Types of Security Policies

Commercial Security Policy – Example

- When a customer moves money from one account to another, the bank uses a **well-formed transaction**
- This transaction has two distinct parts:
 - money is first debited to the original account and then credited to the second account
- Unless both parts of the transaction are completed successfully,
 - the customer will lose the money
- With a **well-formed transaction**, if the transaction is interrupted, the state of the database is **still consistent**
 - Either as it was before the transaction began or as it would have been when the transaction ended
- Hence, part of the bank's security policy is that all transactions **must be well-formed**



Types of Security Policies

Confidentiality Policy Vs. Integrity Policy

- The difference in these two policies is based on the role of trust in these policies
- Confidentiality policy
 - Places **no trust in objects**
 - The policy dictates whether the **object can be disclosed**
 - The policy says nothing about whether the **object should be believed**
- Integrity policy
 - Indicate how much the **object can be trusted**
 - The policy dictates what a subject **can do** with that object
 - But the crucial question is how the level of trust is assigned



Types of Security Policies

Confidentiality Policy Vs. Integrity Policy – Example

- Consider a site obtains a new version of a software. Should that software have
 - high integrity (that is, the site trusts the new version of that program) or
 - low integrity (that is, the site does not yet trust the new program) or
 - somewhere in between (because the vendor supplied the program, but it has not been tested at the local site as thoroughly as the old version)?
- This makes **integrity policies** considerably **more vague than confidentiality policies**
- Assigning a **level of confidentiality** is based on what the organization wants others to know
- Assigning a **level of integrity** is based on what the organization **subjectively** believes to be true about the **trustworthiness** of the information



Types of Security Policies

Confidentiality Policy Vs. Integrity Policy

- Definition
 - A confidentiality policy is a security policy dealing **only with confidentiality**
 - An integrity policy is a security policy dealing **only with integrity**
- Both confidentiality policy and military policy deal with confidentiality
- However, a confidentiality policy **does not** deal with integrity at all, whereas a military policy may
- A similar distinction holds for integrity policies and commercial policies



The Role of Trust

१०८ परमं बला

The Role of Trust



Overview

- The role of trust is crucial to understanding the nature of computer security
- All theories and mechanisms for analyzing and enhancing computer security rely on certain assumptions
- If we understand these assumptions on which security policies, mechanisms, and procedures are based, then
 - we will have a very good understanding of the effectiveness of these policies, mechanisms, and procedures
- Let us examine the consequences of this maxim
 - A system administrator receives a security patch for his computer's operating system. He installs it. Has he improved the security of his system?



The Role of Trust

Assumptions – Informal

- The system administrator has succeeded, given the correctness of certain assumptions:
 - that the patch came from the trusted or known vendor
 - that the patch didn't come from an attacker who is trying to trick him into installing a bogus patch that would actually open security holes
 - that the patch was not tampered with in transit
 - that the vendor tested the patch thoroughly
 - that the vendor's test environment corresponds to his environment
 - that there are no possible conflicts between different patches and patches from different vendors of software that the system is using
 - that the patch is installed correctly

The Role of Trust



Assumptions – Some examples

- The vendor tested the patch thoroughly
 - Vendors are often under considerable pressure to issue patches quickly and sometimes test them only against a particular attack
 - The vulnerability may be deeper and other attacks may succeed
 - When someone released an exploit of one vendor's operating system code, the vendor released a correcting patch in 24 hours
 - Unfortunately, the patch opened a second hole, one that was far easier to exploit
 - The next patch (released 48 hours later) fixed both problems correctly

The Role of Trust



Assumptions – Some examples

- The vendor's test environment corresponds to his environment
 - A vendor's patch once **enabled** the host's personal firewall, causing it to block incoming connections by default
 - This prevented many programs from functioning
 - The host had to be reconfigured to allow the programs to continue to function

The Role of Trust



Assumptions – Some examples

- The patch is installed correctly
 - Some patches are simple to install, because they are simply executable files
 - Others are complex, requiring the system administrator to
 - reconfigure network-oriented properties, add a user, modify the contents of a registry, give rights to some set of users, and then reboot the system
 - An error in any of these steps could prevent the patch from correcting the problems
 - Something similar to an inconsistency between the environments in which the patch was developed and in which the patch is applied
 - Furthermore, the patch **may claim to require specific privileges**, when in reality the privileges are unnecessary and in fact dangerous

The Role of Trust



Trust in Formal Verification

- Gives formal mathematical proof that given input i , program P produces output o as specified in the requirements
- Suppose a security-related program S has been formally verified for the operating system O
- What assumptions are made when it was installed?

The Role of Trust



Trust in Formal Methods

- The formal verification of S is correct—that is, the proof has no errors
 - Because formal verification relies on automated theorem provers and these theorem provers must be programmed correctly
- The preconditions hold in the environment in which the program is to be executed
- The version of O in the environment in which the program is to be executed is the same as the version of O used to verify S
- Note:
 - Automated Theorem Proving (ATP) deals with the development of computer programs that show that some statement (the conjecture) is a logical consequence of a set of statements (the axioms)
 - Example:
 - A = { All men are mortal, Socrates is a man }
 - C = Socrates is mortal

The Role of Trust



Trust in Formal Methods

- The program will be transformed into an executable whose actions correspond to those indicated by the source code
 - In other words, the compiler, linker, loader, and any libraries are correct
- Example
 - An experiment with one version of the UNIX operating system demonstrated how devastating a rigged compiler could be
 - Some attack tools replace libraries with others that perform additional functions, thereby increasing security risks



The Role of Trust

Trust in Formal Methods

- The hardware will execute the program as intended
- Example
 - A program that relies on floating-point calculations would yield incorrect results on some computer CPU chips
 - regardless of any formal verification of the program, owing to a flaw in these chips
 - The Pentium F00F bug
 - The name is shorthand for F0 0F C7 C8, the hexadecimal encoding of one offending instruction
 - A design flaw in the majority of Intel Pentium, Pentium MMX, and Pentium OverDrive processors (all in the P5 microarchitecture). Discovered in 1997, it can result in the processor ceasing to function until the computer is physically rebooted. The bug has been circumvented through operating system updates.



Access Control

१०८ परमं बलू०

Access Control



Overview

- Access Control is all about protecting objects
 - such as, files, tables, hardware devices, or network connections, and other resources
- Need to have different ways of access control. For example:
 - Certain users can have read only access
 - Others can have modification access
 - Some others have no access at all
- Techniques used for this must be robust, easy to use, and efficient.
- Basic access control means
 - "A subject is permitted to access an object in a particular mode, and only such authorized accesses are allowed."
 - --Scott Graham and Peter Denning

Access Control



Definition of Access Control

- NISTIR 7298 – Glossary of Key IS Terms
 - Access Control is 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 – Internet Security Glossary
 - Access Control is 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



Terms

- Subjects:
 - Are human users, often represented by surrogate programs running on behalf of users
- Objects (System Resources)
 - Are things on which an action can be performed. For example,
 - Files, tables, programs, memory objects, hardware devices, strings, data fields, network connections, and processors
 - Users, or programs or processes representing users
 - E.g., an operating system (a program representing the system administrator) can allow a user to execute a program, halt a user, or assign privileges to a user
- Access modes or rights
 - Describe the way in which a subject may access an object
 - Are any controllable actions of subjects on objects. For example
 - Read, write, modify, delete, execute, create, destroy, copy, export, import, and so forth

Access Control



Overview

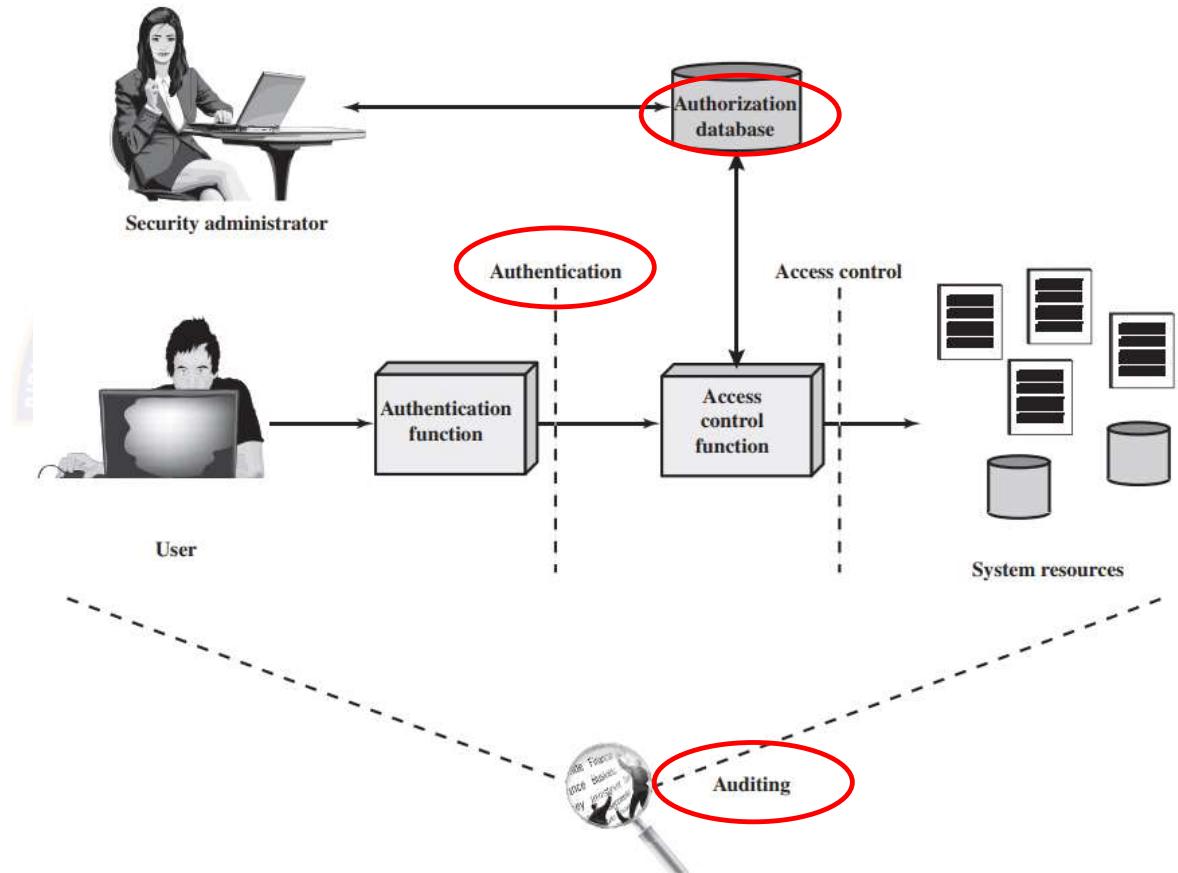
- Access control is the central element of computer security
- The primary objectives of computer security are:
 - to prevent unauthorized users from gaining access to resources
 - to prevent legitimate users from accessing resources in an unauthorized manner, and
 - to enable legitimate users to access resources in an authorized manner.
- Access control implements a security policy
- A security policy specifies
 - **who or what** (e.g., a process or program) may have access to **each specific system resource** and the **type of access** that is **permitted** or **denied** in each instance

Access Control



Context

- In addition to access control, the context involves the following entities and functions:
 - Authentication
 - Authorization
 - Audit



Access Control



Context

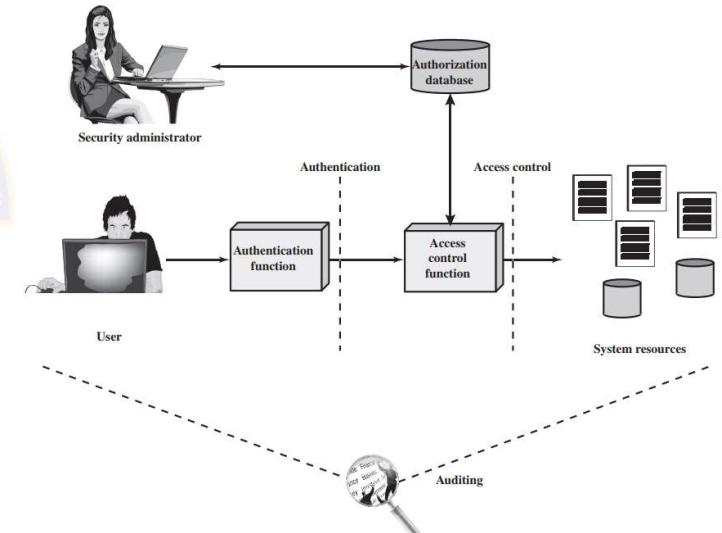
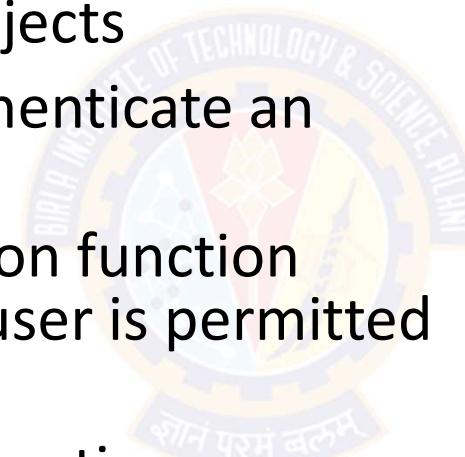
- Authentication:
 - Verification that the credentials of a user or other system entity are valid.
- Authorization:
 - The granting of a right or permission to a system entity to access a system resource
 - This function determines who is trusted for a given purpose.
- Audit:
 - An independent review and examination of system records and activities in order
 - to test for adequacy of system controls
 - to ensure compliance with established policy and operational procedures
 - to detect breaches in security, and to recommend any indicated changes in control, policy and procedures.

Access Control



Context

- An access control mechanism mediates between a subject and objects
- The system must first authenticate an entity seeking access
- Typically, the authentication function determines whether the user is permitted to access the system at all
- Then the access control function determines if the specific requested access by this user is permitted

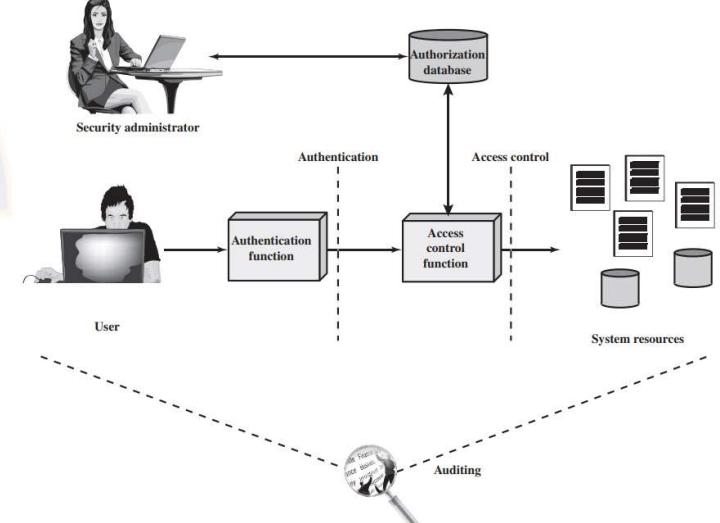
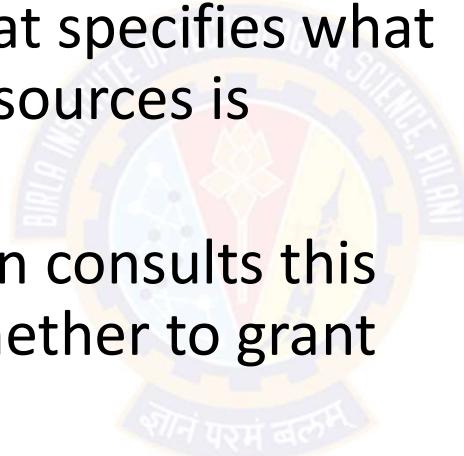


Access Control



Context

- A security administrator maintains an authorization database that specifies what type of access to which resources is allowed for this user
- The access control function consults this database to determine whether to grant access
- An auditing function monitors and keeps a record of user accesses to system resources





Types of Access Control

१०८ परमं बलूः



Types of Access Control

Overview

- There are three main types of access control
 - Discretionary Access Control (DAC) or Identity-based Access Control (IBAC)
 - Individual user sets access control mechanism to allow or deny access to an object
 - Nondiscretionary Access Controls
 - Mandatory Access Control (MAC), occasionally called a Rule-based Access Control
 - System mechanism controls access to object, and individual cannot alter that access
 - Role-based access control (RBAC)
 - Attribute-based access control (ABAC)
 - Originator-controlled Access Control (ORCON or ORGCON)
 - Originator (creator) of information controls who can access information



Types of Access Control

Discretionary Access Control (DAC)/IBAC

- An individual user can set an access control mechanism to allow or deny access to an object
 - Also called an *identity-based access control* (IBAC).
- Most widely known access control
- DACs base access rights on the identities of the subject and the object involved
 - Identity is the key here
- The owner of the object decides who can access it by allowing only particular subjects to have access
- **Identity-based access control** is a subset of DAC because systems identify users based on their identity and assign resource ownership to identities



Types of Access Control

DAC/IBAC - Example

- If you create a file, you are the owner and can grant permissions to any other user to access the file
- The New Technology File System (NTFS), used on Microsoft Windows operating systems, uses the DAC model
- For example
 - If a user creates a new spreadsheet file, that user is both the creator of the file and the owner of the file
 - As the owner, the user can modify the permissions of the file to grant or deny access to other users
 - Data owners can also delegate day-to-day tasks for handling data to data custodians, giving data custodians the ability to modify permissions



Types of Access Control

DAC/IBAC Model – Access Control Lists

- A DAC model is implemented using access control lists (ACLs) on objects
- Each ACL defines the types of access granted or denied to subjects
- It does not offer a centrally controlled management system because owners can alter the ACLs on their objects at will
- Microsoft Windows systems use the DAC model to manage files
- Each file and folder has an ACL identifying the permissions granted to any user or group and the owner can modify permissions
- Within a DAC environment, administrators can easily suspend user privileges while they are away, such as on vacation
- Similarly, it's easy to disable accounts when users leave the organization

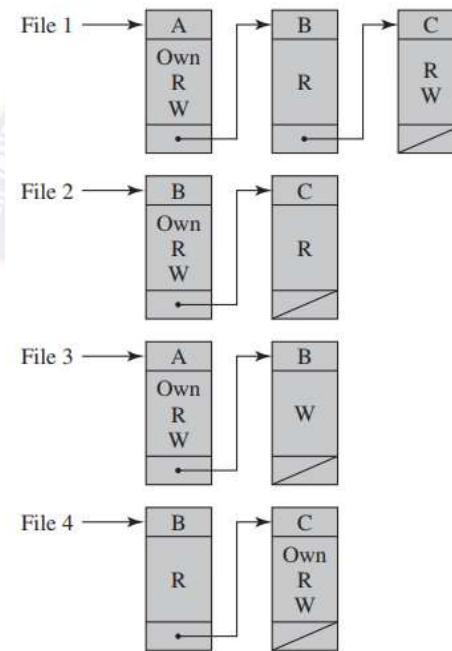
Types of Access Control



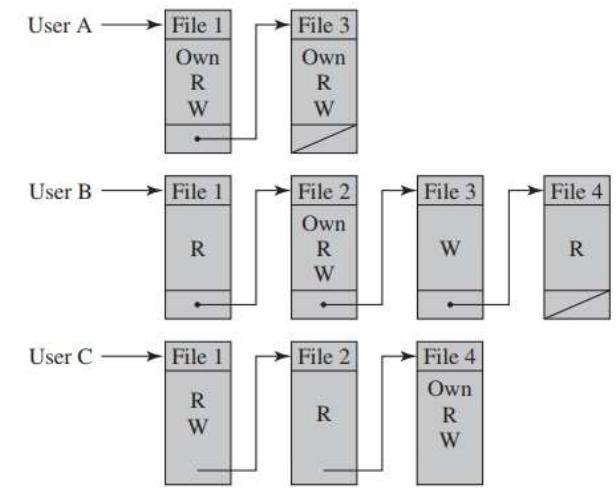
DAC/IBAC Model – Access Control Lists

		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)



(c) Capability lists for files of part (a)



Types of Access Control

DAC/IBAC Model – Access Control Lists

Authorization Table for Files

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



Types of Access Control

Nondiscretionary Access Controls

- The major difference between discretionary and nondiscretionary access controls is in how they are controlled and managed
- Nondiscretionary access controls are **centrally administered** and administrators can make changes that affect the entire environment
- In contrast, DAC models allow owners to make their own changes, and their changes don't affect other parts of the environment.
- In a non-DAC model, access does not focus on user identity
 - Instead, a static set of rules governing the whole environment manages access
- Non-DAC systems are easier to manage, but are less flexible
- These include:
 - Mandatory Access Control (MAC)
 - Role-based access control (RBAC)
 - Attribute-based access control (ABAC)



Types of Access Control

Mandatory Access Control (MAC)

- When a mechanism controls access to an object and an individual user cannot alter that access, the control is a *Mandatory Access Control* (MAC) or *Rule-based Access Control (RAC)*.
- MAC is based on fiat (official sanction), and identity is irrelevant:
- The *operating system* enforces mandatory access controls
- Neither the subject nor the owner of the object can determine whether access is granted
- Typically, the system mechanism checks attributes associated with both the subject and the object to determine whether the subject should be allowed to access the object
- Rules describe the conditions under which access is allowed.



Types of Access Control

MAC/RAC – Example

- The law allows a court to access driving records without an owner's permission
- This is a mandatory control, because the owner of the record has no control over the court's access to the information



Types of Access Control

MAC/RAC

- A MAC model relies on the use of classification labels
- Each classification label represents a security domain, or a realm of security
- A security domain is a collection of subjects and objects that share a common security policy
- For example
 - If a security domain has the label "Secret," the MAC model would protect all objects with the "Secret" label in the same manner
- Subjects are only able to access objects with the "Secret" label when they have a matching "Secret" label



Types of Access Control

MAC/RAC

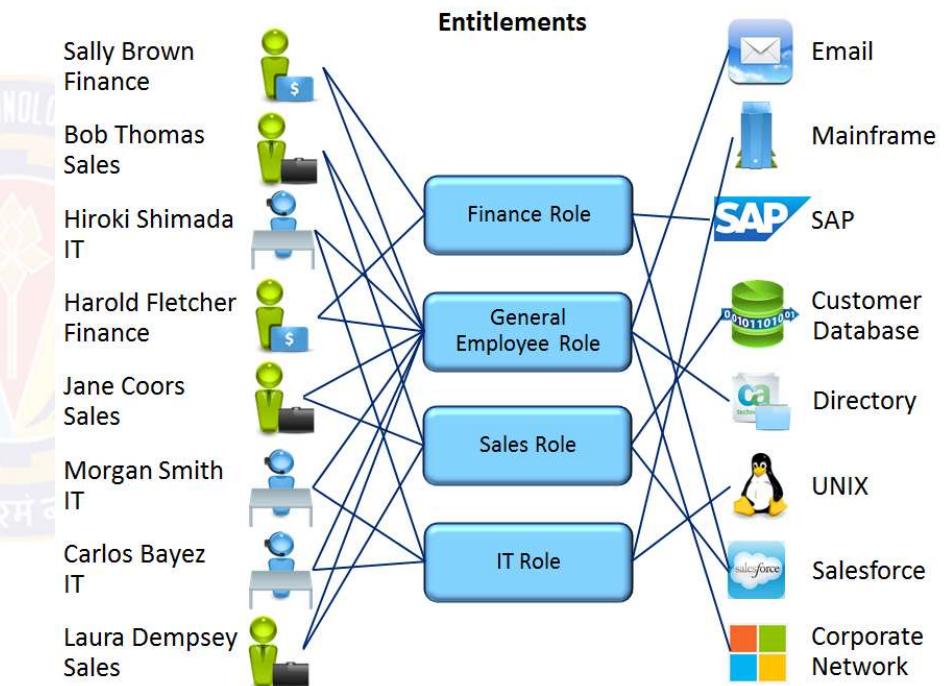
- **Users** have labels assigned to them based on their **clearance level**, which is a form of privilege
- **Objects** have labels, which indicate their **level of classification** or sensitivity
- For example
 - The U.S. military uses the labels of Top Secret, Secret, and Confidential to classify data
 - Administrators can grant access to Top Secret data to users with Top Secret clearances
 - However, administrators cannot grant access to Top Secret data to users with lower-level clearances such as Secret and Confidential
- Governments use labels mandated by law, organizations in private sector are free to choose their labels, such as
 - confidential (or proprietary), private, sensitive, and public

Types of Access Control



Role Based Access Control (RBAC)

- Role-based or task-based access controls define a subject's ability to access an object based on the subject's role or assigned tasks
- Role Based Access Control (RBAC) is often implemented using groups

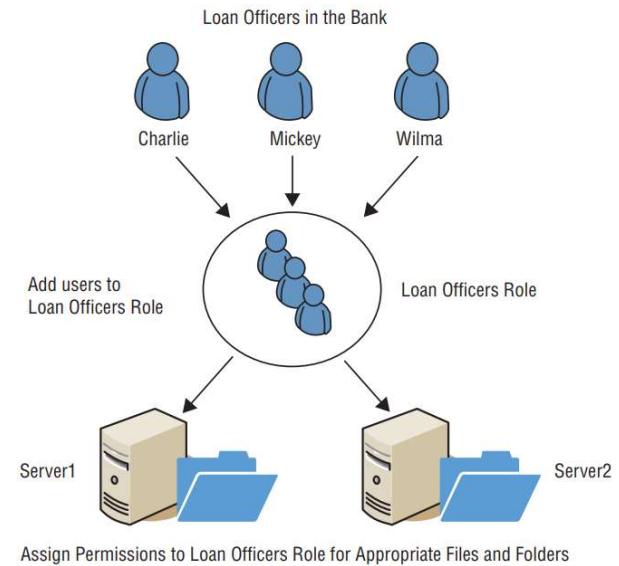


Types of Access Control



Role Based Access Control (RBAC)

- For example:
 - A bank may have loan officers, tellers, and managers
 - Administrators can create a group named Loan Officers, place the user accounts of each loan officer into this group, and then assign appropriate privileges to the group
 - If a new loan officer joins the organization, administrators simply add the new loan officer's account into the Loan Officers group
 - Administrators would take similar steps for tellers and managers.





Types of Access Control

Role Based Access Control (RBAC)

- This helps enforce the principle of least privilege
- Prevents privilege creep, where users accrue privileges over time as their roles and access needs change
- Ideally, administrators revoke user privileges when users change jobs within an organization
- However, when privileges are assigned to users directly, it is challenging to identify and revoke all of a user's unneeded privileges



Types of Access Control

Rule-based Access Controls

- A rule-based access control model uses a set of rules, restrictions, or filters to determine what can and cannot occur on a system
- Distinctive characteristic:
 - Rule(s) apply to all regardless of who the user is
 - They have **global rules** that **apply to all subjects**
- Examples
 - Firewall rules: examines all the traffic going through it and only allows traffic that meets one of the rules
 - Disk or mail quotas
 - Data Loss Prevention (DLP): for making sure that end users do not send sensitive or critical information outside the corporate network



Types of Access Control

Attribute Based Access Controls (ABAC)

- Rule-based access control models include **global rules** that apply to **all subjects** equally
- An advanced implementation of a rule-based access control is an **Attribute Based Access Control** (ABAC) model
- ABAC models use policies that include multiple attributes for rules
 - Attributes are characteristics of users, the network, and devices on the network
- Many software-defined networking applications use ABAC models

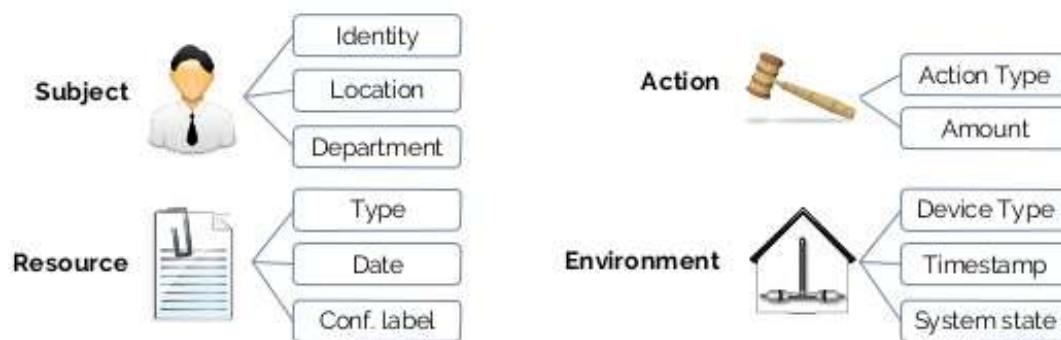
User	Object	Environment
Title	Type	Geo-Location
Group	Date	Network
Department	Sensitivity	Time of Day
Devices		Network



Types of Access Control

Attribute Based Access Controls (ABAC)

Attribute-based Access Control (ABAC)



Managers of the auditing department in Brussels can inspect the financial reports from the current financial year within office hours

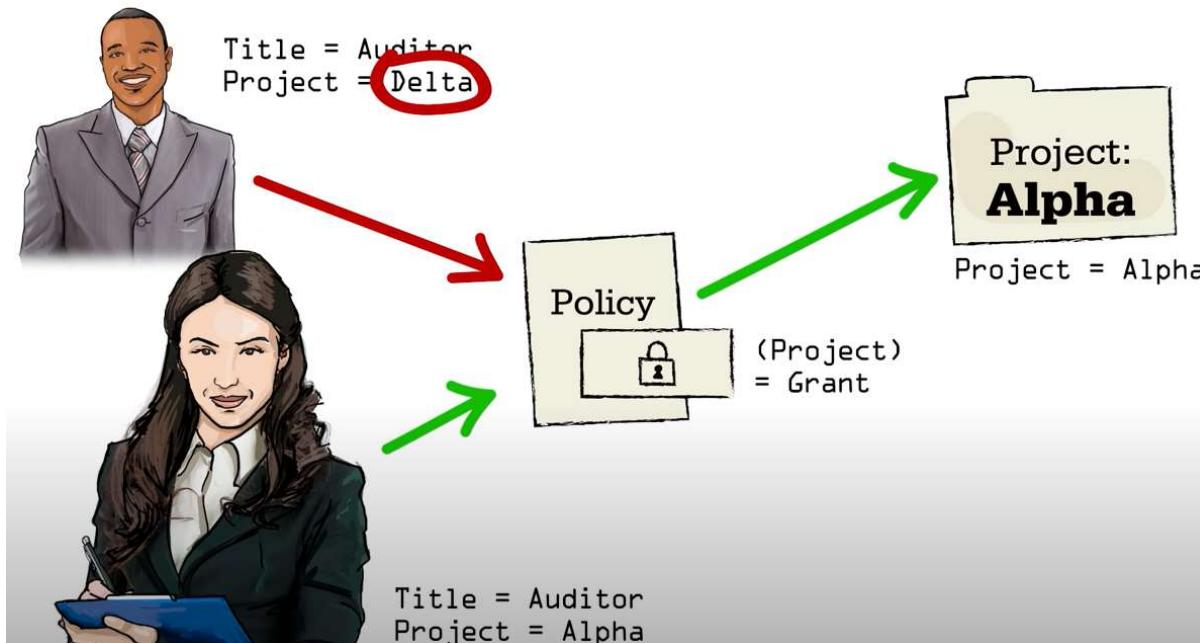
- **Subject**
 - Managers
 - Auditing Department
 - Brussels
- **Action**
 - Inspect
- **Resource**
 - Financial reports
 - Financial year
- **Environment**
 - Current
 - Office Hours

13



Types of Access Control

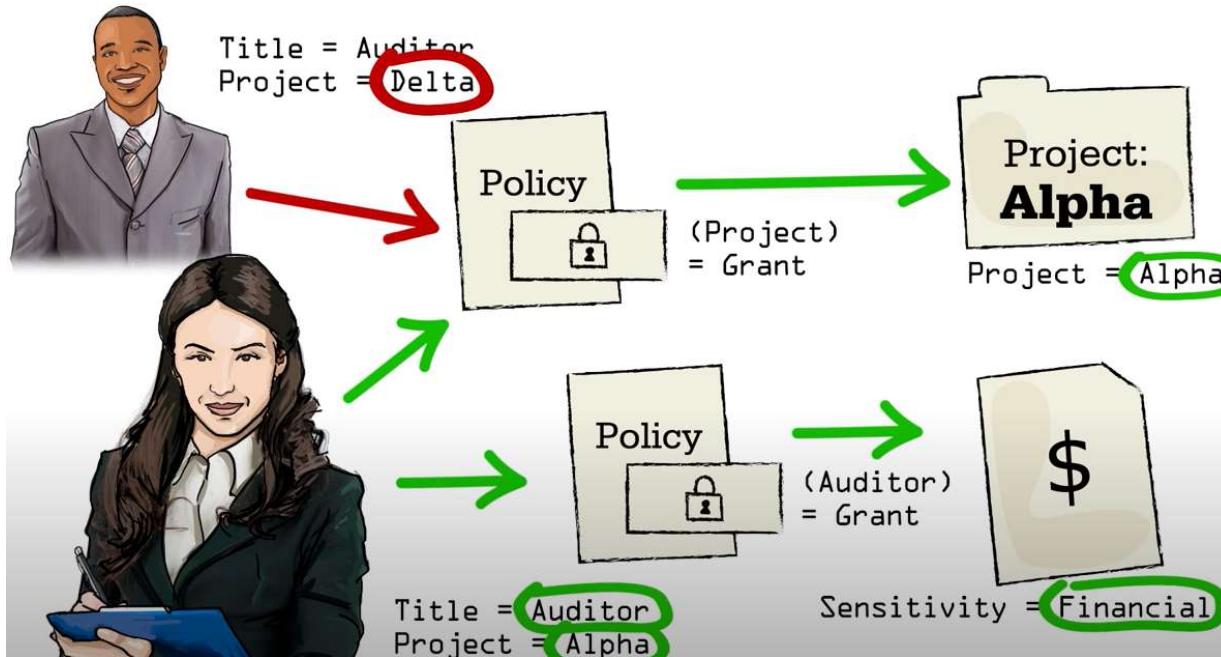
Attribute Based Access Controls (ABAC)



- Subject
 - Auditor
- Action
 - Read, Write
- Resource
 - Financial reports
 - Project = Alpha
 - Project = Delta
- Environment
 - Project Duration

Types of Access Control

Attribute Based Access Controls (ABAC)



- Subject
 - Auditor
- Action
 - Read, Write
- Resource
 - Financial reports
 - Project = Alpha
 - Project = Delta
- Environment
 - Project Duration



Types of Access Control

Attribute Based Access Controls (ABAC) - Example

- CloudGenix has created a software-defined wide area network (SD-WAN) solution that implements policies to allow or block traffic
- Administrators create ABAC policies using plain language statements such as
 - "Allow Managers to access the WAN using tablets or smartphones."
- This allows users in the Managers role to access the WAN using tablet devices or smartphones
- This improves the rule-based access control model, where the control applies to all users, but the ABAC can be much more specific



Types of Access Control

ORCON or ORGCON

- Definition
 - An Originator Controlled Access Control (ORCON or ORGCON) bases access on the creator of an object (or the information it contains)
- The goal of this control is to allow the **originator** of the file (or of the information it contains) **to control** the dissemination of the information
- The owner of the file has no control over who may access the file



Types of Access Control

ORCON or ORGCON – Example

- Bit Twiddlers, Inc., an embedded systems company contracts with Microhackers Ltd., a company equally famous for its microcoding abilities
- The contract requires Microhackers to develop a new microcode language for a particular processor
 - which is designed to be used in high-performance embedded systems
- Bit Twiddlers gives Microhackers a copy of its specifications for the processor
- The terms of the contract require Microhackers to obtain permission before it gives any information about the processor to its subcontractors
- This is an originator controlled access mechanism because, even though Microhackers owns the file containing the specifications, it may not allow anyone to access that information unless the creator of that information, Bit Twiddlers, gives permission



Thank You!



BITS Pilani
Pilani | Dubai | Goa | Hyderabad

Cyber Security

Formal Models of Computer Security

Dr. Ramakrishna Dantu

Associate Professor, BITS Pilani

Disclaimer and Acknowledgement



- The content for these slides has been obtained from books and various other source on the Internet
- I here by acknowledge all the contributors for their material and inputs.
- I have provided source information wherever necessary
- I have added and modified the content to suit the requirements of the course

Formal Models of Computer Security



Agenda

- The CIA Classification:
 - Confidentiality Policies:
 - Bell-LaPadula Model
 - Integrity Policies:
 - The Biba Model
 - Lipner's Integrity Matrix Model
 - Clark-Wilson Integrity Model
 - Trust Models
 - Availability Policies:
 - Deadlock
 - Denial of Service Models



Confidentiality Policies



Overview

- A **confidentiality policy**, also called an **information flow policy**
- Goal: prevent the unauthorized disclosure of information
 - Deals with the flow of information
 - Unauthorized alteration (integrity) of information is secondary
- Multi-level security models are best-known examples
 - Bell-LaPadula Model basis for many, or most, of these
- Example
 - In the United States, the Privacy Act requires that certain personal data be kept confidential
 - Income tax returns are legally confidential and are available only to the Internal Revenue Service or to legal authorities with a court order
 - Governmental models represent the policies that satisfy these requirements



Bell LaPadula Model



Bell LaPadula Model



Overview

- David Bell and Leonard LaPadula first described the DoD multilevel military security policy in 1973 in abstract, formal mathematical terms
- Each **subject** and each **object** is assigned a **security class**
- Security classes form a **strict hierarchy** and are referred to as **security levels**
- Example: The U.S. military classification scheme:
 - top secret > secret > confidential > restricted > unclassified
- Example: Commercial classification scheme
 - strategic > sensitive > confidential > public

Bell LaPadula Model



Overview

- Levels consist are:
 - *Security clearance L(s)* for subjects
 - A subject is said to have a security clearance of a given level
 - *Security classification L(o)* for objects
 - An object is said to have a security classification of a given level
- A subject's (usually a user's) access to an object (usually a file) is allowed or disallowed by
 - comparing the object's security classification with the subject's security clearance
- BPL model uses mathematical notation and set theory to define the concepts of:
 - a secure state, the modes of access, and the rules for granting access



Bell LaPadula Model

Example

<i>security level</i>	<i>subject</i>	<i>object</i>
Top Secret	Tamara	Personnel Files
Secret	Samuel	E-Mail Files
Confidential	Claire	Activity Logs
Unclassified	Ulaley	Telephone Lists

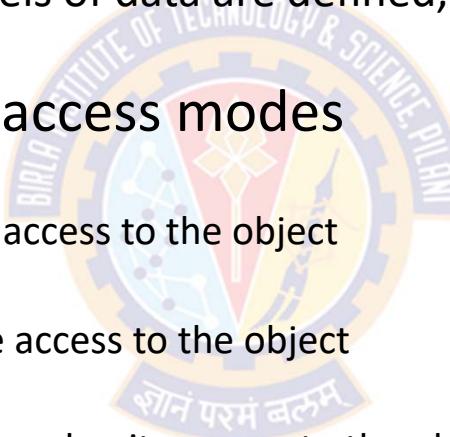
- Tamara can read all files
- Claire cannot read Personnel or E-Mail Files
- Ulaley can only read Telephone Lists

Bell LaPadula Model



Access Modes

- Multilevel Security (MLS)
 - When multiple categories or levels of data are defined, the requirement is referred to as multilevel security (MLS)
- The BLP model defined four access modes
 - read:
 - The subject is allowed only read access to the object
 - append:
 - The subject is allowed only write access to the object
 - write:
 - The subject is allowed both read and write access to the object.
 - execute:
 - The subject is allowed neither read nor write access to the object but may invoke the object for execution.

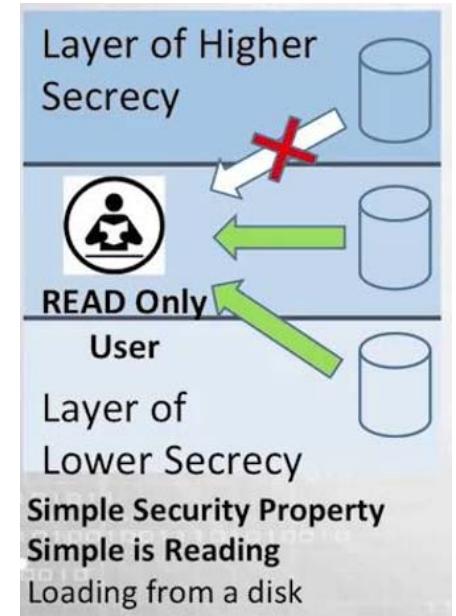


Bell LaPadula Model



Reading Information

- Information flows up, not down
 - "Reads up" disallowed, "reads down" allowed
- Simple Security Condition (Step 1)
 - A subject may only read an object if she has a **clearance level equal to or greater than** the **security level** of the file
 - Subject s can read object o iff, $L(o) \leq L(s)$ and s has permission to read o
 - Note: combines mandatory control (relationship of security levels) and discretionary control (the required permission)
 - Sometimes called "**no reads up**" rule

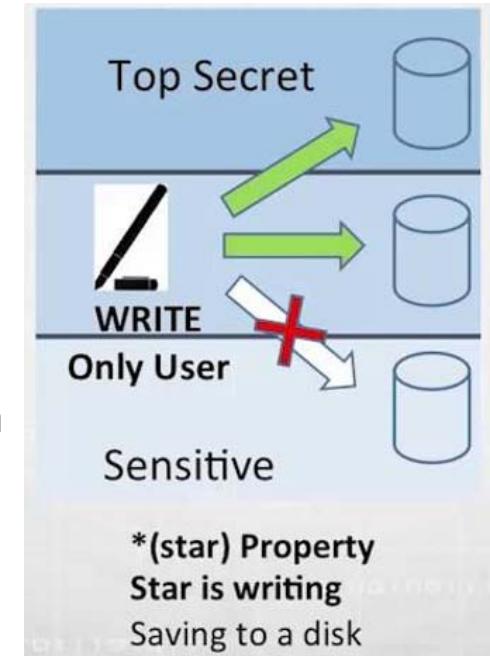
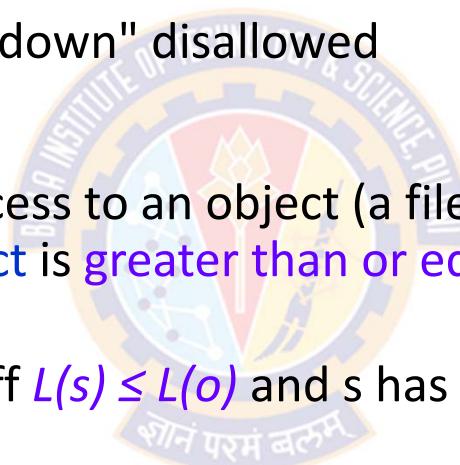


Bell LaPadula Model



Writing Information

- Information flows up, not down
 - "Writes up" allowed, "writes down" disallowed
- *-Property (Step 1)
 - A subject is allowed write access to an object (a file) only if the **security level of the object is greater than or equal to the clearance level of the subject**
 - Subject s can write object o iff $L(s) \leq L(o)$ and s has permission to write o
 - Note: combines mandatory control (relationship of security levels) and discretionary control (the required permission)
 - Sometimes called "**no writes down**" rule



Bell LaPadula Model



Three Basic Rules

- The *-property (star property) (Step-1)
 - This makes it impossible for data from a highly cleared subject to become available to users with a lower security clearance in an object (file/directory) with a low security level
 - Without this rule, a user with a high security clearance could copy sensitive data into a low security clearance document—thus allowing "confidential" data to be written down, or to flow from a "top secret" to an "unclassified" level

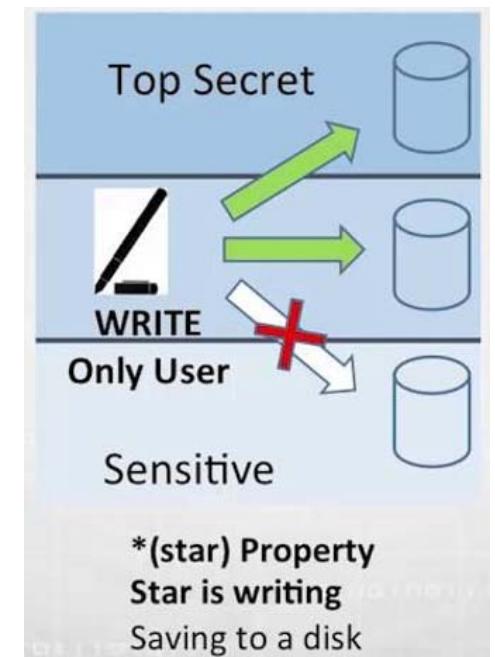


Image Source: Skillset.com

Bell LaPadula Model



Three Basic Rules

- The simple security condition (Step-1)
 - someone with a "secret" security level cannot read a file with a "top secret" security level, but can read a file with a "secret" or "confidential" security level
- The tranquility property
 - It states that the security level of an object cannot be changed while it is being processed by a computer system
 - This keeps a program or attack from modifying the sensitivity of a file while it is open and vulnerable

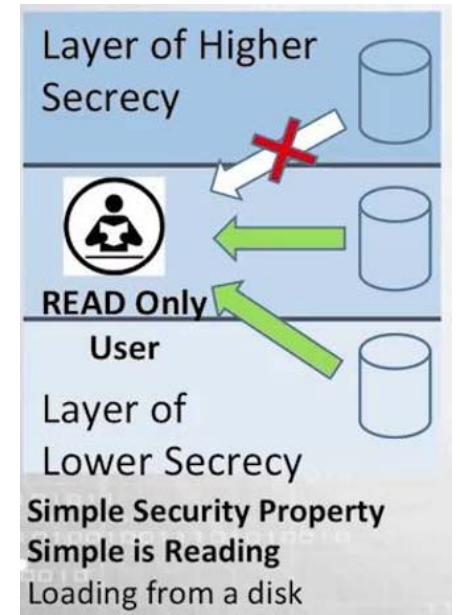
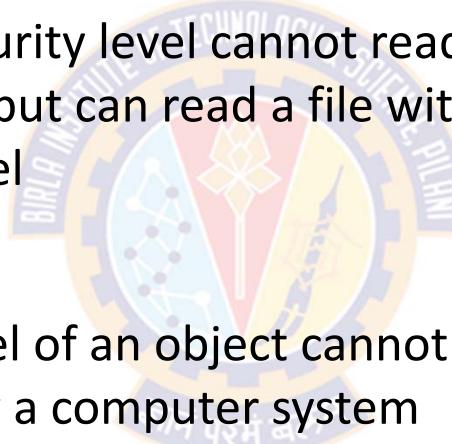


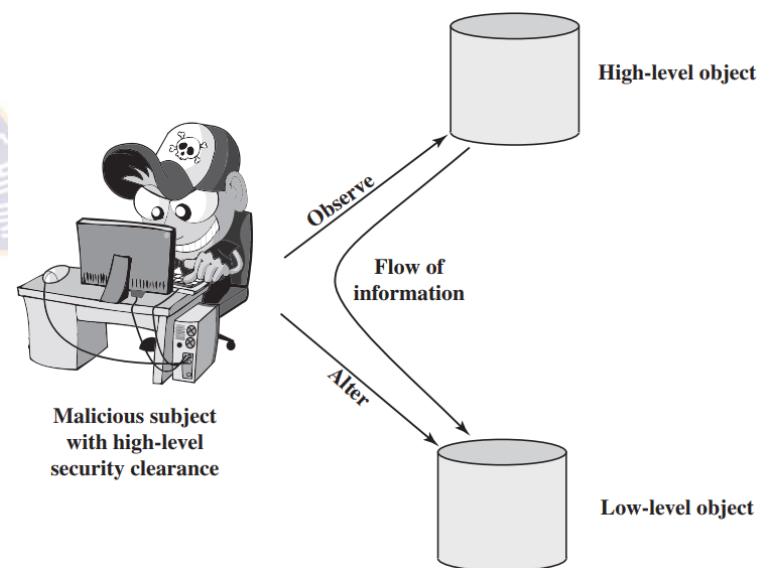
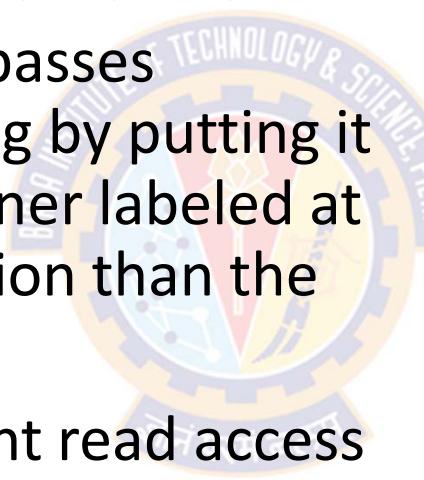
Image Source: Skillset.com

Bell LaPadula Model



Three Basic Rules

- What is the need for the *-property?
- Here, a malicious subject passes classified information along by putting it into an information container labeled at a lower security classification than the information itself
- This will allow a subsequent read access to this information by a subject at the lower clearance level



Bell LaPadula Model



Approach

- Use state-transition systems to describe computer systems
- Define a system as secure iff. every reachable state satisfies 3 properties
 - simple-security property
 - *-property
 - discretionary security property

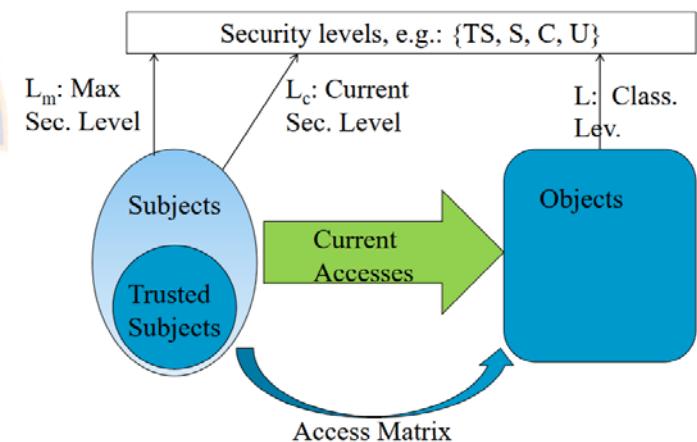
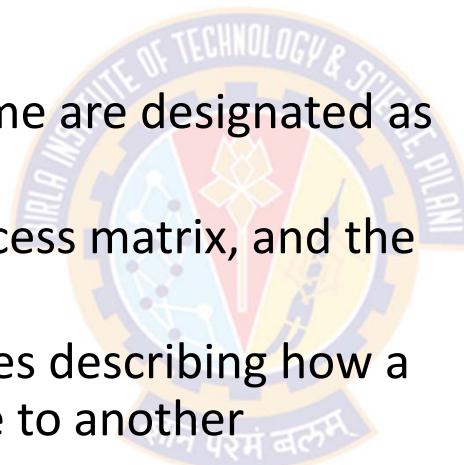


Bell LaPadula Model



Approach

- A computer system is modeled as a state transition system
 - There is a set of subjects; some are designated as trusted
 - Each state has objects, an access matrix, and the current access information
 - There are state transition rules describing how a system can go from one state to another
 - Each subject s has a maximal sec level $L_m(s)$ and a current sec level $L_c(s)$
 - Each object has a classification level

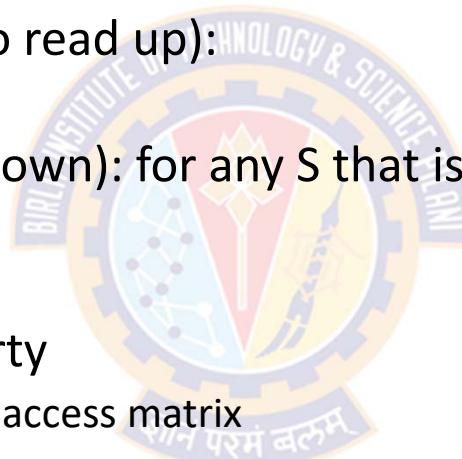


Bell LaPadula Model



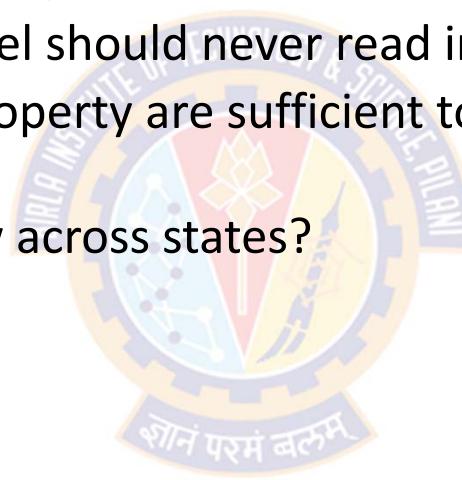
Approach

- A state is considered secure if it satisfies
 - Simple Security Condition (no read up):
 - S can read O iff $L_m(S) \geq L(O)$
 - The Star Property (no write down): for any S that is not trusted
 - S can read O iff $L_c(S) \geq L(O)$
 - S can write O iff $L_c(S) \leq L(O)$
 - Discretionary-security property
 - every access is allowed by the access matrix
- A system is secure if and only if every reachable state is secure



Is BLP Notion of Security Good?

- The objective of BLP security is to ensure
 - a subject cleared at a low level should never read information classified high
 - The ss-property and the *-property are sufficient to stop such information flow at any given state
 - What about information flow across states?

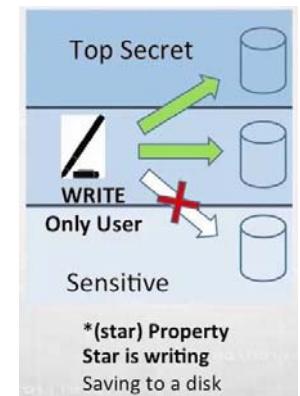
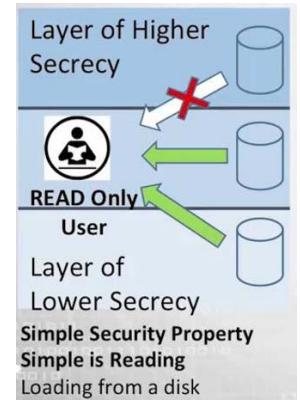
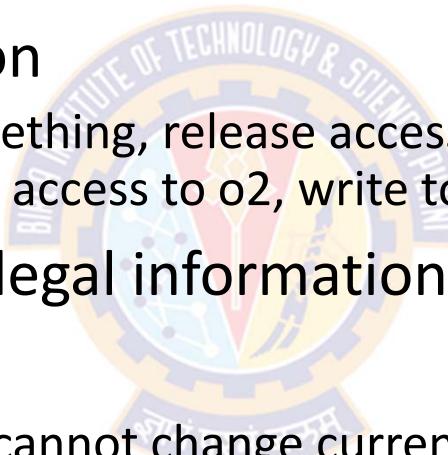


Bell LaPadula Model



Is BLP Notion of Security Good?

- Consider a system with s_1, s_2, o_1, o_2
- And the following execution
 - s_1 gets access to o_1 , read something, release access, then change current level to low, get write access to o_2 , write to o_2
- Every state is secure, yet illegal information exists
- Solution:
 - Tranquility principle: subject cannot change current levels



Bell LaPadula Model



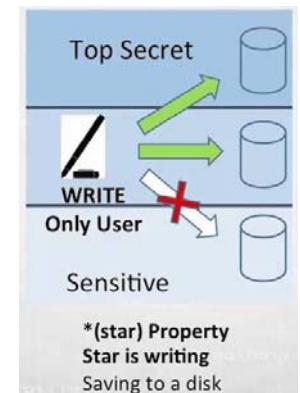
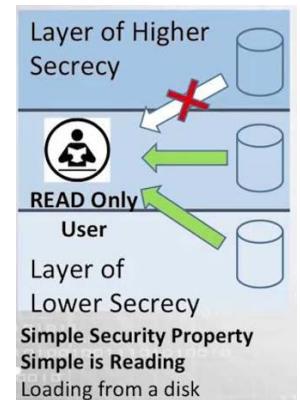
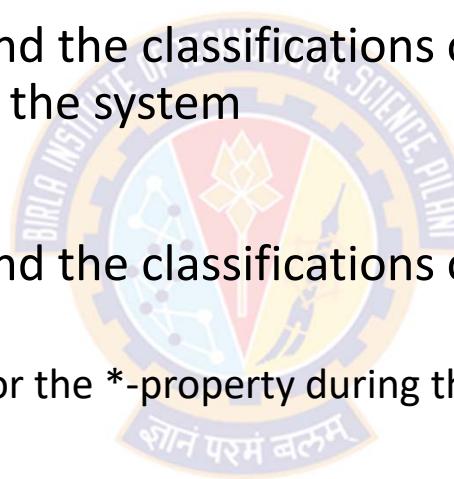
Principle of Tranquility

- **Strong Tranquility**

- The clearances of subjects, and the classifications of objects, do not change during the lifetime of the system

- **Weak Tranquility**

- The clearances of subjects, and the classifications of objects, do not change in a way that violates
 - the simple security condition or the *-property during the lifetime of the system



Bell LaPadula Model



Extension

- Why Extension is needed?
 - Since all information is not meant for all people, we need to classify the information too into categories
 - Categories also known as compartments
- Typical military security categories
 - Nuclear Defense (abbreviated: NUC)
 - European Politics (EUR)
 - US Governmental issues (US)
 - army, navy, air force
 - nato, nasa, noforn
- Typical commercial security categories
 - Sales, , R&D, HR
 - Dept A, Dept B, Dept C
- But how these categories can go with security classification levels:
 - Top Secret (TS), Secret (S), Confidential (c) and Unclassified (UC)



Bell LaPadula Model



Attaching Category with i) User and ii) Info. Security Levels

- Example:
 - William may be cleared into the level:
 - (SECRET, {EUR}) and
 - George may be cleared into the level
 - (TOP SECRET,{NUC,US})
- A document may be classified as
 - (CONFIDENTIAL, {EUR})
- How can we compare the security levels of user with that of documents?
- This is needed to satisfy the Bell-LaPadula model

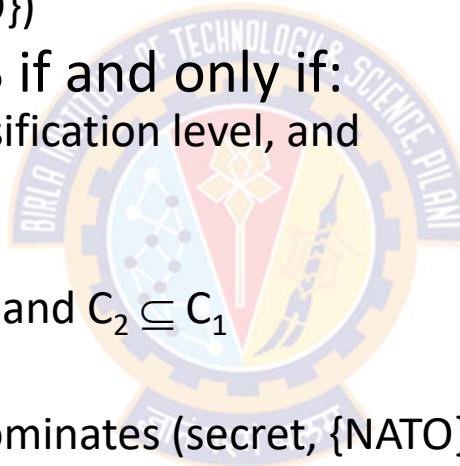


Bell LaPadula Model



Security Categories and Dominance

- Security Level = {Security Classification, {Set of Categories} }
 - E.g., (top-secret, {Nuclear, NATO})
- Security level A dominates B if and only if:
 - A's classification level > B's classification level, and
 - A's category set contains B's
- That is,
 - $(SC_1, C_1) \geq (SC_2, C_2)$ iff. $SC_1 \geq SC_2$ and $C_2 \subseteq C_1$
- For instance
 - (top-secret, {Nuclear, NATO}) dominates (secret, {NATO})
- because
 - top-secret > secret, and
 - the set {Nuclear, NATO} contains {NATO}

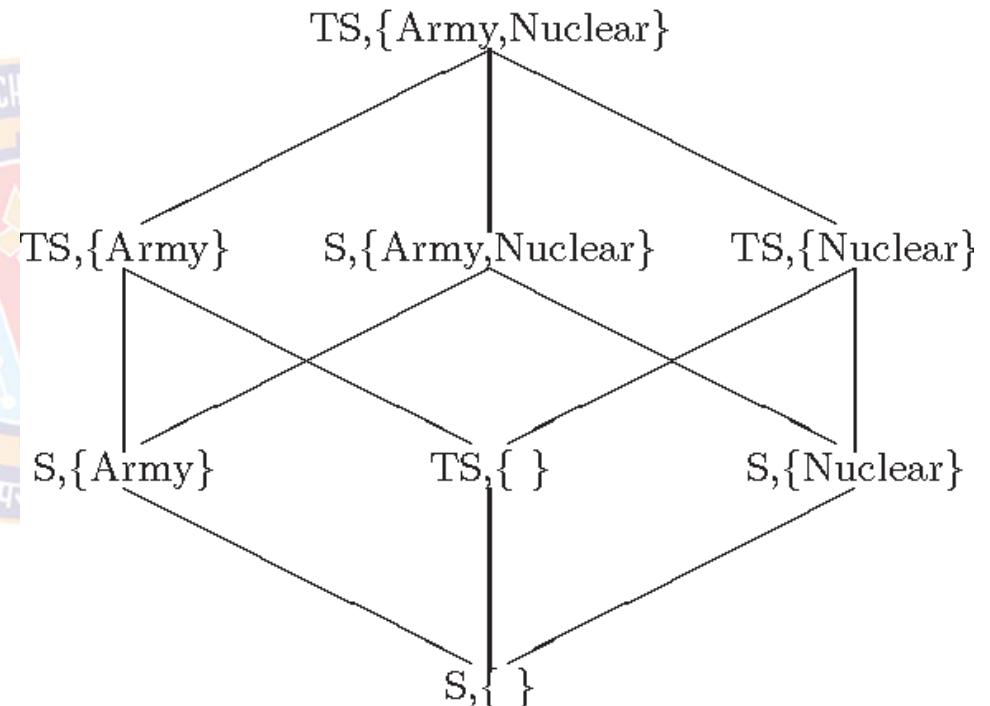


Bell LaPadula Model



Lattice of Categories

- $(TS, \{\text{Army}, \text{Nuclear}\})$ dominates $(S, \{\text{Army}\})$
- $(TS, \{\text{Army}, \text{Nuclear}\})$ dominates $(TS, \{\text{Nuclear}\})$
- $(S, \{\text{Army}, \text{Nuclear}\})$ dominates $(S, \{\text{Nuclear}\})$
- $(S, \{\text{Army}\})$ dominates $(S, \{\})$





Integrity Policies

A circular university crest featuring a blue border with the text "JAYTECHNOLGY" and a central emblem with the motto "शोनं परमं बलम्".

Integrity Policies



Overview

- Requirements
 - Very different than confidentiality policies
- Biba's models
 - Strict Integrity policy
- Lipner's model
 - Combines Bell-LaPadula, Biba
- Clark-Wilson model
- Trust models
 - Policy-based
 - Reputation-based



Integrity Policies



Requirements

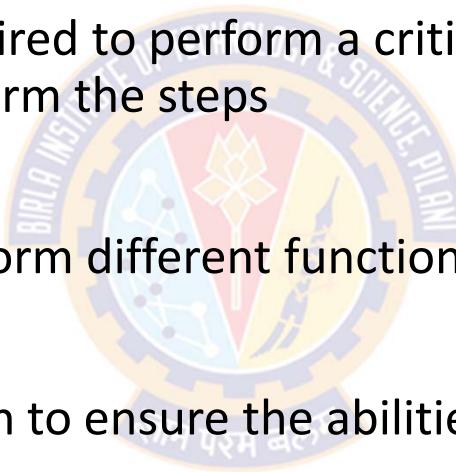
- Users will not write their own programs, but will use existing production programs and databases.
- Programmers will develop and test programs on a non-production system; if they need access to actual data, they will be given production data via a special process, but will use it on their development system.
- A special process must be followed to install a program from the development system onto the production system.
- The special process in requirement 3 must be controlled and audited.
- The managers and auditors must have access to both the system state and the system logs that are generated.

Integrity Policies



Principles of Operation

- *Separation of duty*:
 - if two or more steps are required to perform a critical function, at least two different people should perform the steps
- *Separation of function*:
 - different entities should perform different functions
- *Auditing*:
 - recording enough information to ensure the abilities to both recover and determine accountability





The Biba Model

Overview

- The Biba Model or Biba Integrity Model developed by Kenneth J. Biba in 1975
- The model is based on information flow, and the objects and subjects are grouped into ordered levels of integrity
- The Biba model was designed after the BLP model
 - sometimes called the Bell-LaPadula upside down model
- Where the BLP model addresses **confidentiality**, the Biba model addresses **integrity**
- The model is designed so that subjects may not corrupt data in a level ranked higher than the subject, or be corrupted by data from a lower level than the subject.
- The model is also built on state transition system of computer security policy that describes a set of access control rules designed to ensure data integrity



The Biba Model

Overview

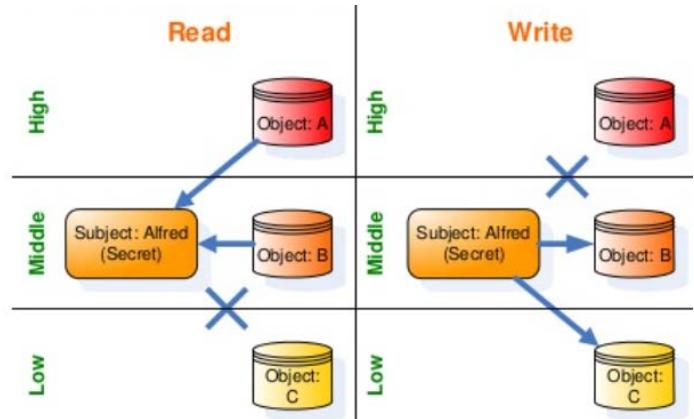
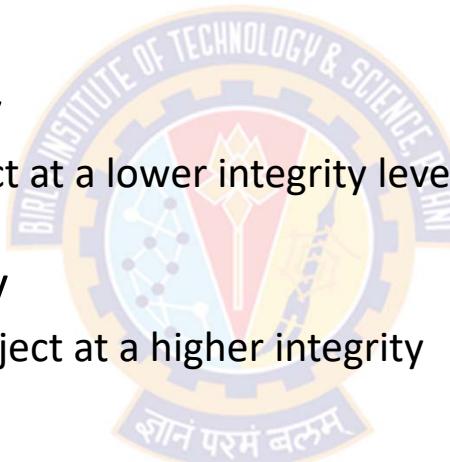
- Like other models, the Biba model supports the access control of both subjects and objects.
 - Subjects:
 - are the active elements in the system that can access information (processes acting on behalf of the users).
 - Objects:
 - are the passive system elements for which access can be requested (files, programs, etc.).
- Each subject and object will have a integrity level associated with it
 - denoted as $I(S)$ and $I(O)$ for subject S and object O, respectively
- A simple hierarchical classification uses a strict ordering of levels from lowest to highest
- Biba was designed to address three integrity issues:
 - Prevent modification of objects by unauthorized subjects.
 - Prevent unauthorized modification of objects by authorized subjects.
 - Protect internal and external object consistency

The Biba Model



Properties

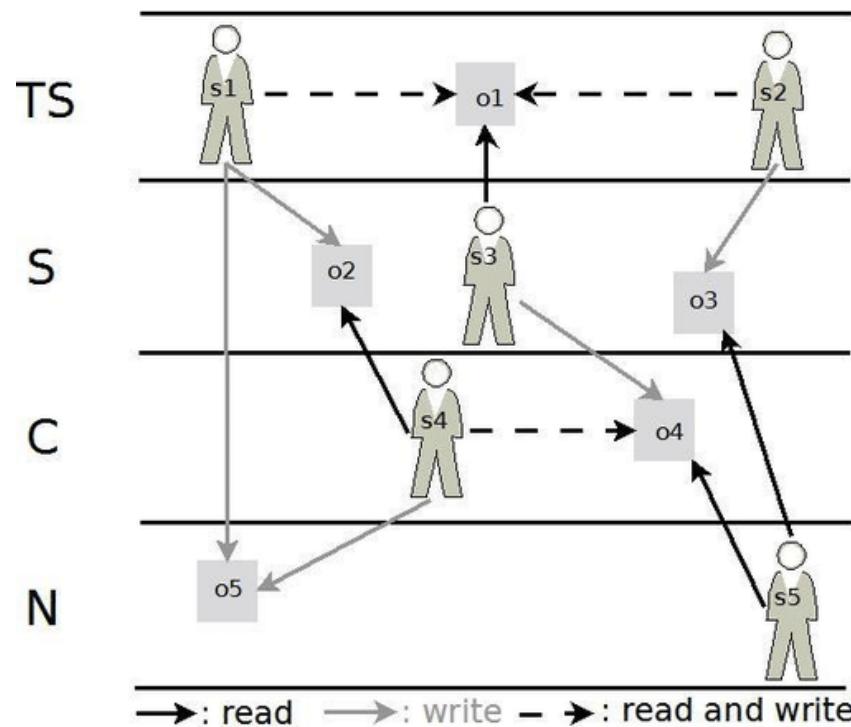
- Basic properties or axioms of the Biba model state machine:
 - The Simple Integrity Property
 - A subject cannot read an object at a lower integrity level (no read-down).
 - The * (star) Integrity Property
 - A subject cannot modify an object at a higher integrity level (no write-up)
 - Invocation Property
 - A subject cannot send messages (logical request for service) to object of higher integrity



Biba Model



Properties



	o_1	o_2	o_3	o_4	o_5
s_1	read write	write			write
s_2	read write			write	
s_3	read			write	
s_4		read		read write	write
s_5			read	read	

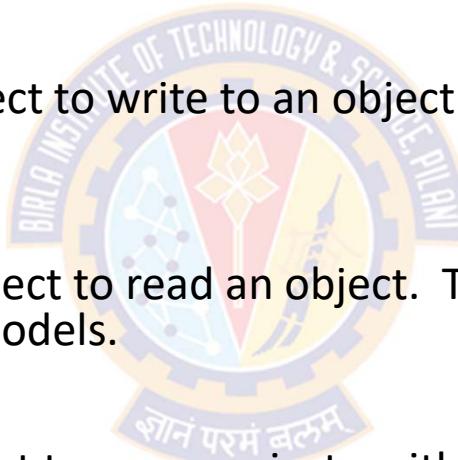
Image Source: https://www.researchgate.net/figure/Two-Laws-of-Biba-Model-The-satisfaction-of-both-Biba-laws-prevents-the-information-flow_fig3_273706233



The Biba Model

Access Modes

- The Biba model consists of the following access modes:
 - **Modify:**
 - The modify mode allows a subject to write to an object. This mode is similar to the write mode in other models.
 - **Observe:**
 - The observe mode allows a subject to read an object. This command is synonymous with the read command of most other models.
 - **Invoke:**
 - The invoke mode allows a subject to communicate with another subject.
 - **Execute:**
 - The execute mode allows a subject to execute an object. The command essentially allows a subject to execute a program which is the object.

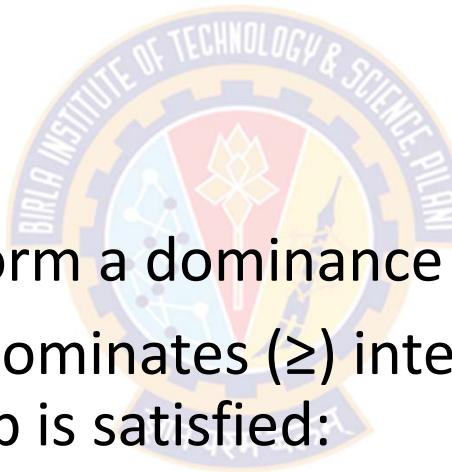


The Biba Model



Integrity Levels

- Each integrity level will be represented as $L = (C, S)$ where:
 - L is the integrity level
 - C is the classification
 - S is the set of categories.
- The integrity levels then form a dominance relationship.
- Integrity level $L_1 = (C_1, S_1)$ dominates (\geq) integrity level $L_2 = (C_2, S_2)$ if and only if this relationship is satisfied:
 - $C_1 \geq C_2$ and $S_2 \subseteq S_1$



The Biba Model



Biba Policies

- The Biba model is actually a family of different policies that can be used.
- The goal of the model is to prevent the contamination of "clean" high level entities from "dirty" low level entities
- The model supports both mandatory and discretionary policies.
- **The Mandatory Policies:**
 - Strict Integrity Policy
 - Low-Watermark Policy for Subjects
 - Low-Watermark Policy for Objects
 - Low-Watermark Integrity Audit Policy
 - Ring Policy
- **The Discretionary Policies:**
 - Access Control Lists
 - Object Hierarchy
 - Ring

The Biba Model



Strict Integrity Policy

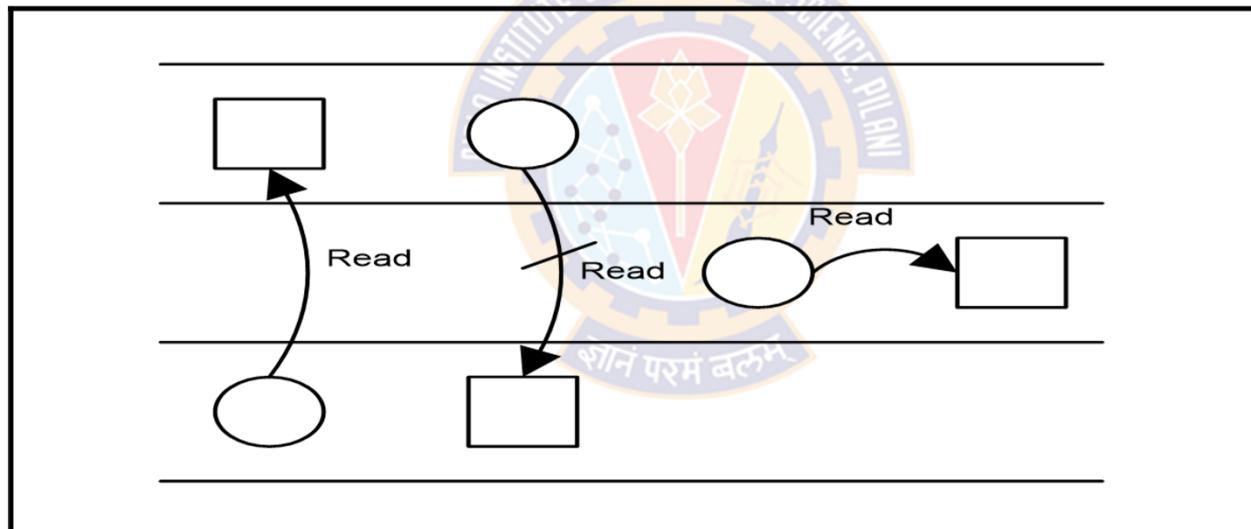
- Simple Integrity Condition (“no read-down”):
 - A subject can read an object only if : $I(S) \leq I(O)$.
 - $s \in S$ can observe $o \in O$ if and only if $i(s) \leq i(o)$
- Integrity Star Property (“no write-up”):
 - A subject can modify an object only if : $I(S) \geq I(O)$.
 - $s \in S$ can modify $o \in O$ if and only if $i(o) \leq i(s)$
- Invocation Property:
 - A subject can invoke/comm with another subject only if : $I(S1) \geq I(S2)$.
 - $s_1 \in S$ can invoke $s_2 \in S$ if and only if $i(s_2) \leq i(s_1)$

The Biba Model



Strict Integrity Policy

- Simple Integrity Condition (“no read-down”):



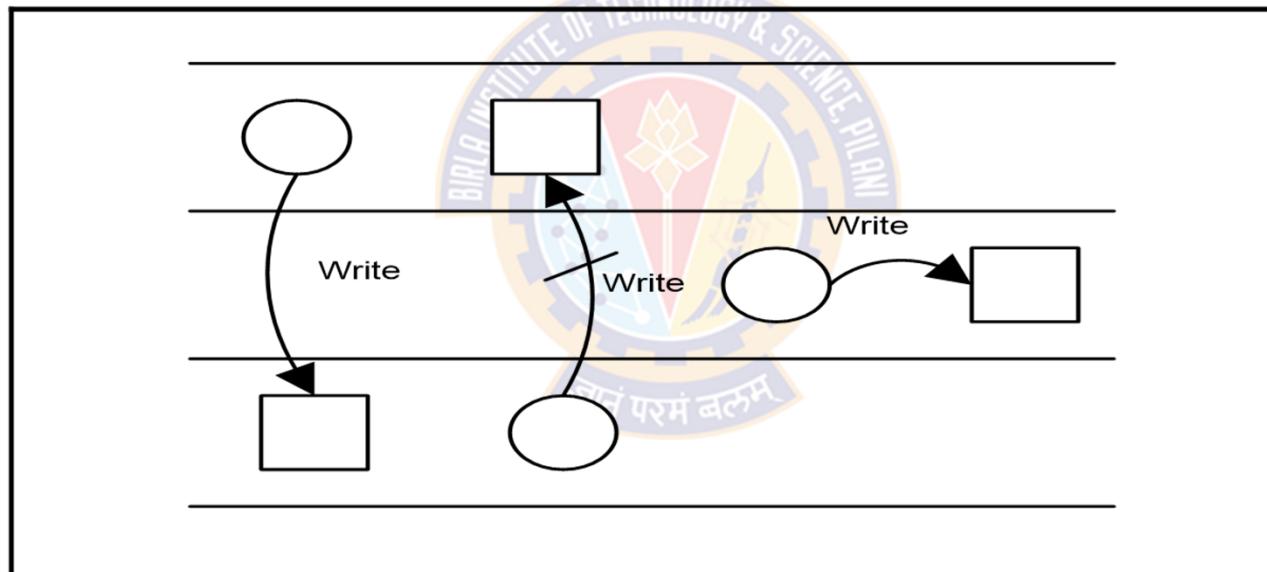
circle = subject, square = object

The Biba Model



Strict Integrity Policy

- Integrity Star Property (“no write-up”):



circle = subject, square = object

The Biba Model



Strict Integrity Policy

- The "no write-up" is essential because it limits the damage that can be done by malicious objects in the system
- For instance:
 - "no write-up" limits the amount of damage that can be done by a trojan horse in the system
 - The trojan horse would only be able to write to objects at its integrity level or lower
 - This is important because it limits the damage that can be done to the operating system.
- The "no read-down" prevents a trust subject from being contaminated by a less trusted object

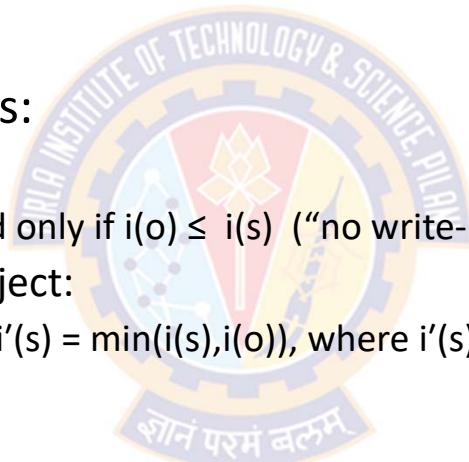


The Biba Model

Low-Water-Mark Policy

- The low-watermark policy for subjects

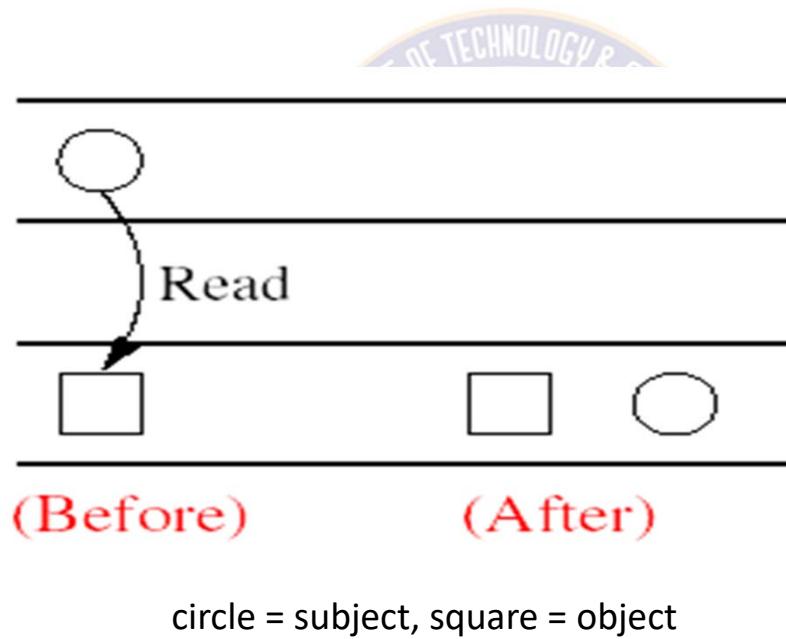
- Is a relaxed "no read-down"
- Contains these following rules:
 - Integrity Star Property:
 - $s \in S$ can modify $o \in O$ if and only if $i(o) \leq i(s)$ ("no write-up").
 - A subject may examine any object:
 - If $s \in S$ examines $o \in O$ then $i'(s) = \min(i(s), i(o))$, where $i'(s)$ is the subjects integrity level after the read.
 - Invocation Property:
 - $s_1 \in S$ can invoke $s_2 \in S$ if and only if $i(s_2) \leq i(s_1)$.



The Biba Model

Low-Water-Mark Policy

- The low-watermark policy for subjects





The Biba Model

Low-Water-Mark Policy

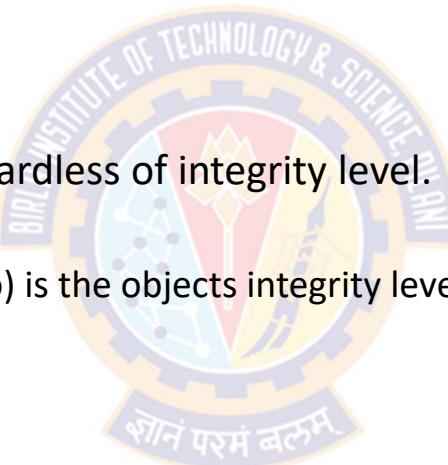
- The low-watermark policy for subjects
 - Does nothing to restrict a subject from reading objects.
 - Is a dynamic policy, because it lowers the integrity level of a subject based on what objects are observed.
 - Drawback
 - One problem with this policy is that if a subject observes a less trusted object, it will drop the subjects integrity level to that of the object
 - Then later, if the subject needs to legitimately observe other objects, it may not be able to do so because the subjects integrity level has been lowered
 - The effect of this would be denial of service depending on the timing of the submissions.



The Biba Model

Low-Water-Mark Policy

- The low-watermark policy for objects
 - Is a relaxed "no write-down"
 - Contains the following rules:
 - $s \in S$ can modify any $o \in O$ regardless of integrity level.
 - If $s \in S$ modifies $o \in O$ then
 - $i'(o) = \min(i(s), i(o))$, where $i'(o)$ is the objects integrity level after it is modified.

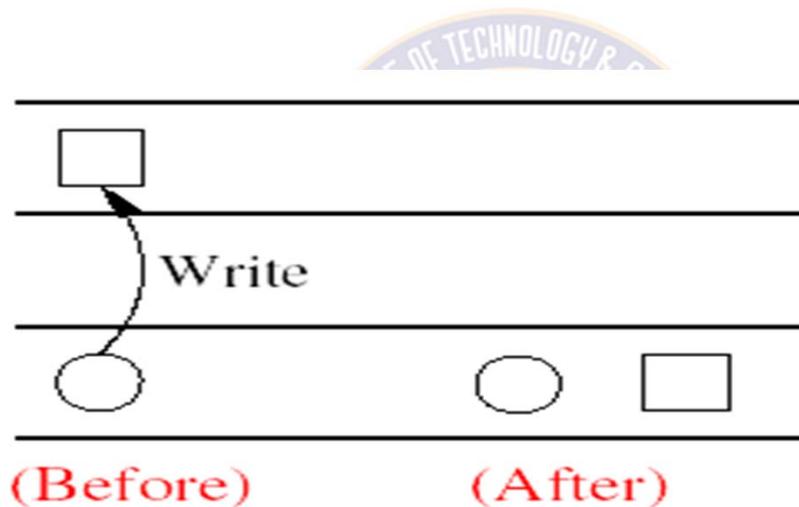


The Biba Model



Low-Water-Mark Policy

- The low-watermark policy for objects



circle = subject, square = object



The Biba Model

Low-Water-Mark Policy

- The low-watermark policy for objects
 - Is also a dynamic policy, similar to the low-watermark policy for subjects.
 - It does nothing to prevent an un-trusted subject from modifying a trusted object
 - In reality policy is not very practical.
 - The policy provides no real protection in a system
 - The policy simply lowers in the trust placed in the objects
 - If a malicious program was inserted into the computer system it could modify any object in the system
 - This model would just lower the integrity level of objects that have become contaminated



The Biba Model

Low-Water-Mark Policy

- The low-watermark Integrity Audit Policy
 - The policy consists of the following rules:
 - Any subject may modify any object, regardless of integrity levels.
 - If a subject modifies an object at higher integrity level (a more trusted object), it results in the transaction being recorded in an audit log.
 - The drawback to this policy is it does nothing to prevent an improper modifications of an object
 - This policy is similar to the low-watermark for objects policy, except in this case the objects integrity level is not lowered, it is recorded.
 - This policy simply records that an improper modification took place.

The Biba Model



Drawbacks

- Advantages:
 - The Biba model is simple and easy to implement.
 - The Biba model provides a number of different policies that can be selected based on need.
- Disadvantages:
 - The model does nothing to enforce confidentiality.
 - The Biba model doesn't support the granting and revocation of authorization.
 - To use this model all computers in the system must support the labeling of integrity for both subjects and objects
 - To date, there is no network protocol that supports this labeling. So there are problems with using the Biba model in a network environment.



Thank You!



BITS Pilani
Pilani | Dubai | Goa | Hyderabad

Cyber Security

Formal Models of Computer Security

Dr. Ramakrishna Dantu

Associate Professor, BITS Pilani

Disclaimer and Acknowledgement



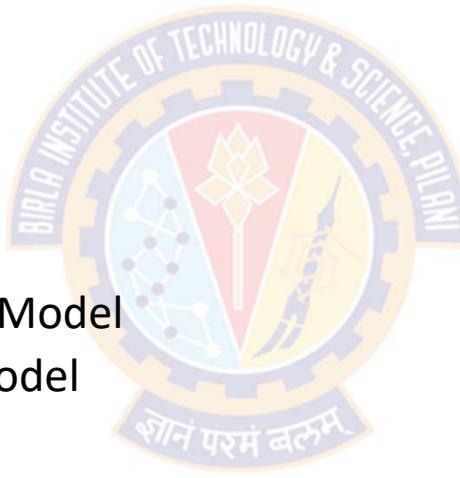
- The content for these slides has been obtained from books and various other source on the Internet
- I here by acknowledge all the contributors for their material and inputs.
- I have provided source information wherever necessary
- I have added and modified the content to suit the requirements of the course

Formal Models of Computer Security



Agenda

- The CIA Classification:
 - Confidentiality Policies:
 - Bell-LaPadula Model
 - Integrity Policies:
 - The Biba Model
 - Lipner's Integrity Matrix Model
 - Clark-Wilson Integrity Model
 - Trust Models
 - Availability Policies:
 - Deadlock
 - Denial of Service Models





Lipner's Integrity Matrix



Integrity Policies - Recap



Commercial Integrity Constraints

- Users will not write their own programs, but use existing production software and databases
- Programmers will develop and test programs on a non-production system; if they need access to actual data, they will be given production data via a special process, but will use it on their development system.
- A special process must be followed to install a program from the development system onto the production system.
- The special process in requirement 3 must be controlled and audited.
- The managers and auditors must have access to both the system state and the system logs that are generated.

Lipner's Integrity Matrix



Overview

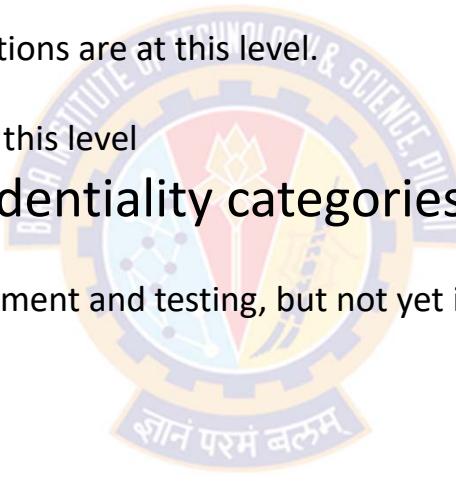
- Lipner devised his Integrity Matrix Model specifically to handle those concerns/constraints in a commercial environment
- Lipner's model accomplishes this by combining the elements of Bell La-Padula and Biba models to provide confidentiality and integrity
- Does it in two steps
 - Bell-LaPadula component first (Confidentiality)
 - Add in Biba component (Integrity)



Lipner's Integrity Matrix

Lipner's Use of Bell-LaPaluda Model

- There are two confidentiality levels (higher to lower):
 - Audit Manager (AM):
 - system audit and management functions are at this level!
 - System Low (SL):
 - any process can read information at this level
- In addition there are five confidentiality categories:
 - Development (D):
 - production programs under development and testing, but not yet in production use
 - Production Code (PC):
 - production processes and programs
 - Production Data (PD):
 - data covered by the integrity policy
 - System Development (SD):
 - system programs under development, but not yet in production use
 - Software Tools (T):
 - programs provided on the production system not related to the sensitive or protected data



Lipner's Integrity Matrix



User/Subject Properties

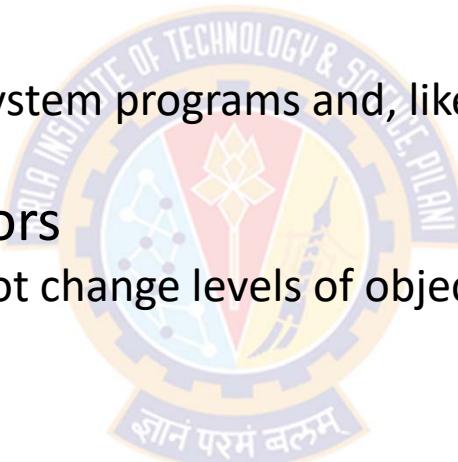
- Lipner then assigned users to security levels based on their jobs.
- Ordinary users
 - can execute (read) production code but cannot alter it
 - can alter and read production data
 - cannot execute category T (Software Tools), so they cannot write their own programs
- Application Developers
 - need access to tools for developing their programs
 - do not have read/write access to PD (Production Data), so cannot access production data
 - If they need production data, the data must first be downgraded to D (this requires sys admins)

Lipner's Integrity Matrix



User/Subject Properties

- Lipner then assigned users to security levels based on their jobs.
- System Programmers
 - System programmers develop system programs and, like application programmers, use tools to do so
- System managers and Auditors
 - need access to all logs but cannot change levels of objects
- System controllers
 - need to install code
 - must have the ability to downgrade code once it is certified for production, so other entities cannot write to it;
- Etc.



Lipner's Integrity Matrix



Users and Security Levels

- Lipner then assigned users to security levels based on their jobs

Subjects	Description	Security Level
Ordinary users	Will use production code to modify production data	(SL, { PC, PD })
Application developers	Develop programs and need access to tools for developing their programs	(SL, { D, T })
System programmers	Develop system programs and, use tools to do so	(SL, { SD, T })
System managers and auditors	Need high clearance to be able to access all logs	(AM, { D, PC, PD, SD, T })
System controllers	Must have the ability to downgrade code once it is certified for production, so other entities cannot write to it	(SL, {D, PC, PD, SD, T}) and downgrade privilege

- E.g.: Ordinary users have security level of System Low (SL) under the categories of Production Code and Production Data
- E.g.: System Programmers have security level of System Low (SL) under the categories of System Development and Software Tools

Lipner's Integrity Matrix



Users and Security Levels

- Lipner then assigned users to security levels based on their jobs

Security Level → Categories↓	Audit Manager (AM)	System Low (SL)
Development (D)	System managers and auditors	Application Developers; System Controller
Production Code (PC)	System managers and auditors	Ordinary Users; System Controller
Production Data (PD)	System managers and auditors	Ordinary Users; System Controller
System Development (SD)	System managers and auditors	System Programmers; System Controller
Software Tools (T)	System managers and auditors	Application Developers; System Programmers; System Controller

Lipner's Integrity Matrix



Objects and Classifications

- Objects are assigned to security levels/categories based on who should access them
- Objects that might be altered have two categories:
 - that of the data itself and that of the program that may alter it
- For example:
 - Ordinary user needs to execute (read) production code,
 - so this is labeled (SL, {PC})
 - This is based on simple security policy of the Bell-LaPadula Model
 - Ordinary users should be able to write production data,
 - so this is labeled (SL, {PC, PD})
 - This is based on *-property of the Bell-LaPadula Model



Objects	Security Level
Development code/test data	(SL, { D, T })
Production code	(SL, { PC })
Production data	(SL, { PC, PD })
Software tools	(SL, { T })
System programs	(SL, \emptyset)
System programs in modification	(SL, { SD, T })
System and application logs	(AM, { appropriate })



Bell LaPadula Model

Access Modes

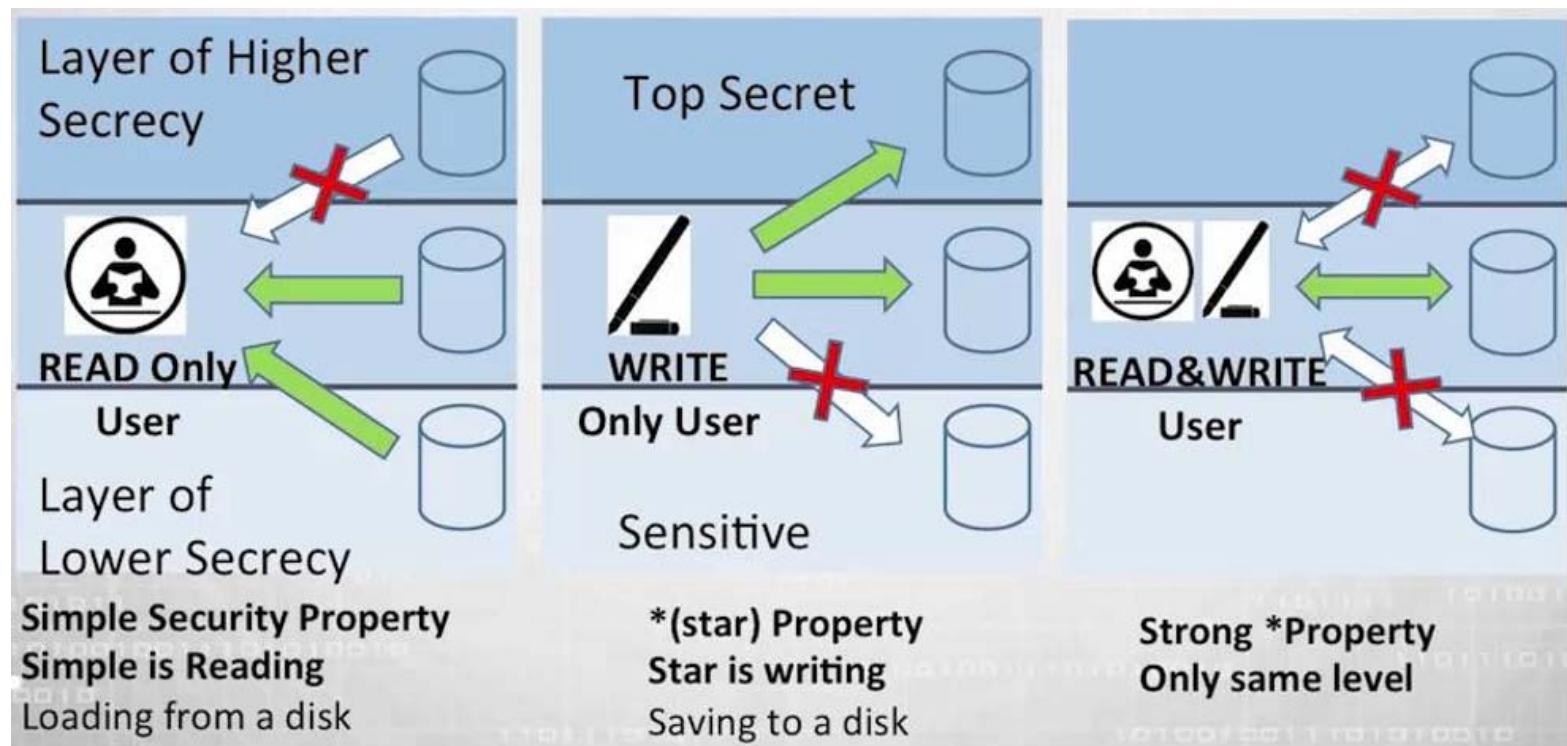


Image Source: Skillset.com



Lipner's Integrity Matrix

Subjects/Objects and Clearance/Classifications

Subjects	Clearance	Objects	Classification
Ordinary users	(SL, { PC, PD })	Development code/test data	(SL, { D, T })
Application developers	(SL, { D, T })	Production code	(SL, { PC })
System programmers	(SL, { SD, T })	Production data	(SL, { PC, PD })
System managers and auditors	(AM, { D, OC, OD, SD, T })	Software tools	(SL, { T })
System controllers	(SL, {D, PC, PD, SD, T}) and downgrade privilege	System programs	(SL, Ø)
		System programs in modification	(SL, { SD, T })
		System and application logs	(AM, { appropriate })

Here downgrade means the ability to move software (objects) from development to production

Lipner's Integrity Matrix



Check Requirements

Requirements	Check
Users will not write their own programs, but will use existing production programs and databases.	Users have no access to T, so cannot write their own programs
Programmers will develop and test programs on a non-production system; if they need access to actual data, they will be given production data via a special process, but will use it on their development system.	Applications programmers have no access to PD, so cannot access production data; if needed, it must be put into D, requiring the system controller to intervene
A special process must be followed to install a program from the development system onto the production system.	Installing a program requires downgrade procedure (from D to PC), so only system controllers can do it
The special process in requirement 3 must be controlled and audited.	Control: only system controllers can downgrade Audit: any such downgrading must be logged
The managers and auditors must have access to both the system state and the system logs that are generated.	System management and audit users are in AM and so have access to system state and logs

Lipner's Integrity Matrix



Problem

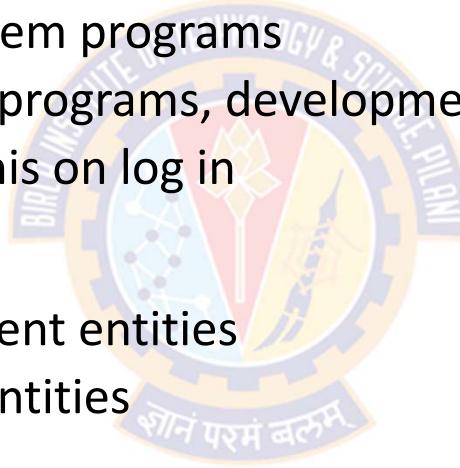
- The model is too inflexible in special-purpose software
 - For example, a program for repairing an inconsistent or erroneous production database cannot be application-level software
 - System managers cannot run programs for repairing inconsistent or erroneous production database
 - System managers at AM, production data at SL
- So to remedy these problems, Lipner integrates his model with Biba's model

Lipner's Integrity Matrix



Adding Biba

- Three integrity classifications (highest to lowest)
 - ISP (System Program): for system programs
 - IO (Operational): production programs, development software
 - ISL (System Low): users get this on log in
- Two integrity categories
 - ID (Development): development entities
 - IP (Production): production entities



ISP > IO > ISL

Lipner's Integrity Matrix



Simplify Bell-LaPadula (Confidentiality)

- In the original model, the security category T (tools) allowed:
 - application developers and system programmers to use the same programs without being able to alter those programs
- The revised model now distinguishes two integrity categories:
 - Development and Production
 - They serve the purpose of the security tools (T) category, which is eliminated from the model
- Production code and production data is collapsed into a single category (called SP)

Lipner's Integrity Matrix



Simplify Bell-LaPadula (Confidentiality)

- This gives rise to the following three confidentiality categories:

- Production (**SP**):
 - Production code (**PC**) and data (**PD**)
- Development (**SD**):
 - Same as previous category Development (**D**)
- System Development (**SSD**):
 - Same as previous category System Development (**SD**)



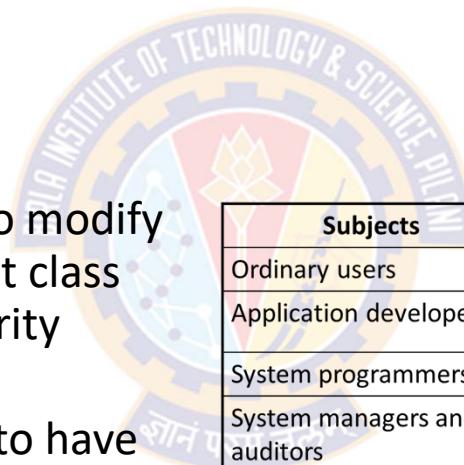
	Original	New
Subjects		
Development	D	SD
Production Code (PC):	PC	SP
Production Data (PD):	PD	SP
System Development (SD):	SD	SSD
Software Tools (T):	T	Eliminated

Lipner's Integrity Matrix



Users and Levels

- The integrity classes are chosen to allow modification of data and programs as appropriate
- For Example:
 - Ordinary users should be able to modify production data, so users of that class must have write access to integrity category IP
 - App developers should be able to have write access to integrity category ID
- Table shows the integrity levels and security categories of users.



Subjects	Security Level	Integrity Level
Ordinary users	(SL, { SP })	(ISL, { IP })
Application developers	(SL, { SD })	(ISL, { ID })
System programmers	(SL, { SSD })	(ISL, { ID })
System managers and auditors	(AM, { SP, SD, SSD })	(ISL, { IP, ID })
System controllers	(SL, { SP, SD }) and downgrade privilege	(ISP, { IP, ID })
Repair	(SL, { SP })	(ISL, { IP })

ISP > IO > ISL



Lipner's Integrity Matrix

Comparison of Old and New Security Levels

	Original	New	New
Subjects	Confidentiality Level	Confidentiality Level	Integrity Level
Ordinary users	(SL, { PC, PD })	(SL, { SP })	(ISL, { IP })
Application developers	(SL, { D, T })	(SL, { SD })	(ISL, { ID })
System programmers	(SL, { SD, T })	(SL, { SSD })	(ISL, { ID })
System managers and auditors	(AM, { D, OC, OD, SD, T })	(AM, { SP, SD, SSD })	(ISL, { IP, ID })
System controllers	(SL, {D, PC, PD, SD, T}) and downgrade privilege	(SL, { SP, SD }) and downgrade privilege	(ISP, { IP, ID })
Repair	Not available	(SL, { SP })	(ISL, { IP })

Here downgrade means the ability to move software (objects) from development to production

ISP > IO > ISL

Lipner's Integrity Matrix



Objects and Classifications

- The final step is to select integrity classes for objects
- Consider the objects Production Code and Production Data
- Ordinary users must be able to:
 - write production data, but not production code
- By placing:
 - Production Data in integrity class (ISL, {IP}) and
 - Production Code in integrity class (IO, {IP})
an ordinary user cannot alter production code but can alter production data ($\text{IO} > \text{ISL}$)
- Similar analysis leads to the levels shown in the next table

Lipner's Integrity Matrix



Objects and Classifications

Objects	Security Level	Integrity Level
Development code/test data	(SL, { SD })	(ISL, { IP })
Production code	(SL, { SP })	(IO, { IP })
Production data	(SL, { SP })	(ISL, { IP })
Software tools	(SL, Ø)	(IO, { ID })
System programs	(SL, Ø)	(ISP, { IP, ID })
System programs in modification	(SL, { SSD })	(ISL, { ID })
System and application logs	(AM, { appropriate })	(ISL, Ø)
Repair	(SL, {SP})	(ISL, { IP })

ISP > IO > ISL



Lipner's Integrity Matrix

Subjects/Objects and Clearance/Classifications - Revised

Subjects	Clearance	Integrity Level	Objects	Classification	Integrity Level
Ordinary users	(SL, { SP })	(ISL, { IP })	Development code/test data	(SL, { D, T })	(ISL, { IP })
Application developers	(SL, { SD })	(ISL, { ID })	Production code	(SL, { SP })	(IO, { IP })
System programmers	(SL, { SSD })	(ISL, { ID })	Production data	(SL, { SP })	(ISL, { IP })
System managers and auditors	(AM, { SP, SD, SSD })	(ISL, { IP, ID })	Software tools	(SL, { Ø })	(IO, { ID })
System controllers	(SL, { SP, SD }) and downgrade privilege	(ISP, { IP, ID })	System programs	(SL, { Ø })	(ISP, { IP, ID })
Repair	(SL, { SP })	(ISL, { IP })	System programs in modification	(SL, { SSD })	(ISL, { ID })
			System and application logs	(AM, { appropriate })	(ISL, { Ø })
			Repair	(SL, { SP })	(ISL, { IP })

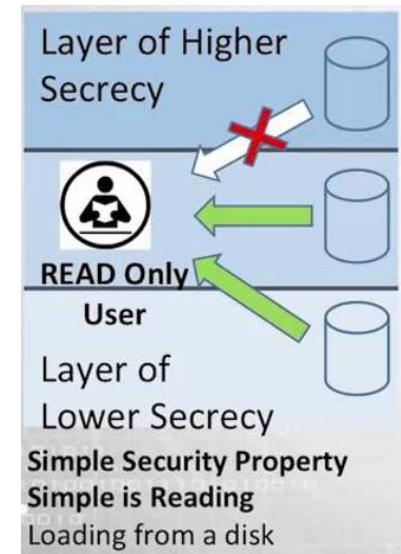
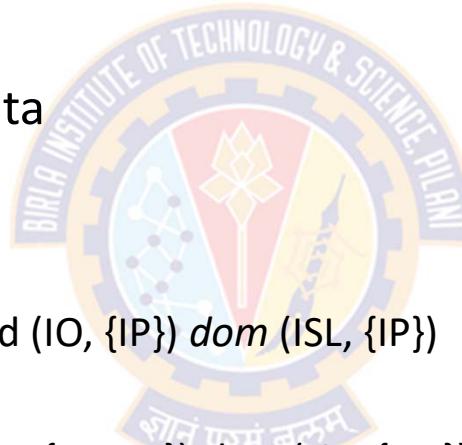
ISP > IO > ISL

Lipner's Integrity Matrix



Repair Class of Users

- Has the same integrity and security clearance as that of production data
 - so can read and write that data
- It can also
 - read production code
 - same security classification and $(IO, \{IP\}) \text{ dom } (ISL, \{IP\})$
 - read system programs
 - $(SL, \{SP\}) \text{ dom } (SL, \{ \phi \})$ and $(ISP, \{ IP, ID \}) \text{ dom } (ISL, \{ IP \})$
 - repair objects
 - same security classes and same integrity classes



Lipner's Integrity Matrix

Repair Class of Users (Contd...)

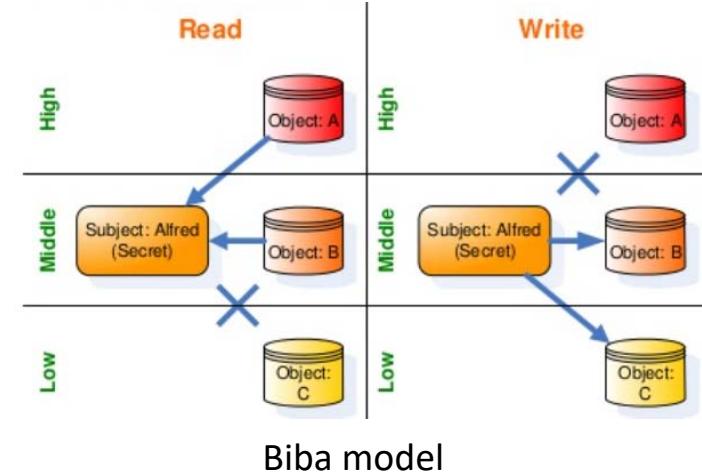
- It can write, but not read
 - the system and application logs, because
 - $(AM, \{ SP \}) \text{ dom } (SL, \{ SP \})$ and $(ISL, \{ IP \}) \text{ dom } (ISL, \{ \phi \})$
- It cannot access
 - development code/test data
 - since the security categories are disjoint
 - system programs in modification
 - since the integrity categories are disjoint
 - software tools
 - since the integrity categories are disjoint
- Thus, the repair function works as needed

Lipner's Integrity Matrix



What can an ordinary user do?

- Ordinary users can : $(SL, \{ SP \})$ $(ISL, \{ IP \})$
 - Read and write production data (same security integrity levels)
 - Read production code
 - same classification – Can Read
 - $(IO, IP) \text{ dom } (ISL, \{ IP \})$ – Cannot write
 - System program
 - $(SL, \{ SP \}) \text{ dom } (SL, \emptyset)$ &
 - $(ISP, \{ IP, ID \}) \text{ dom } \{ ISL, \{ IP \} \}$
 - Repair objects (same levels)
 - Write (not read) the system and application log
 - $(AM, \{ SP \}) \text{ dom } (SL, \{ SP \})$ &
 - $(ISL, \{ IP \}) \text{ dom } \{ ISL, \emptyset \}$





Clark-Wilson Integrity Model



Clark-Wilson Integrity Model



Overview

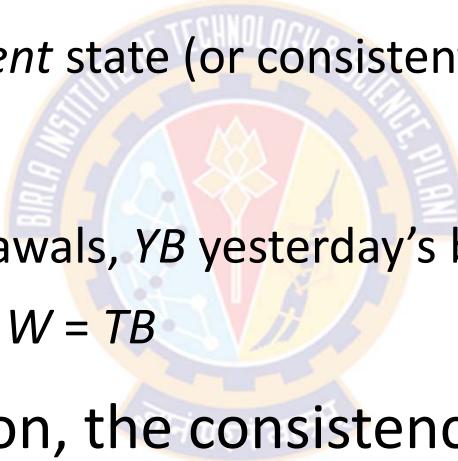
- Clark and Wilson proposed a more elaborate and practical integrity model in 1987
- The Clark-Wilson integrity model (CWM) is specifically designed for commercial operations
- The CWM defines each data item and allows modifications through only a small set of programs
- The CWM does not use of a lattice structure used to define the levels of security that an object may have and that a subject may have access to
- Instead, it uses a three part relationship of subject/program (transaction)/object known as a triple or an access control triple

Clark-Wilson Integrity Model



Overview

- Integrity defined by a set of constraints
 - Data is said to be in a *consistent* state (or consistent) if it satisfies given properties
- Example: Bank
 - D today's deposits, W withdrawals, YB yesterday's balance, TB today's balance
 - Integrity constraint: $YB + D - W = TB$
- Before and after each action, the consistency conditions must hold.

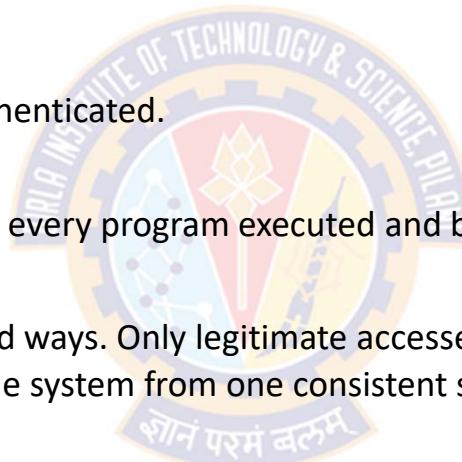


Clark-Wilson Integrity Model



Four Basic Constraints

- Clark and Wilson claimed that the following are four fundamental constraints of any reasonable commercial integrity model:
- **Authentication:**
 - identity of all users must be properly authenticated.
- **Audit:**
 - modifications should be logged to record every program executed and by whom, in a way that cannot be subverted.
- **Well-formed transactions:**
 - Users manipulate data only in constrained ways. Only legitimate accesses are allowed.
 - Is a series of operations that transition the system from one consistent state to another consistent state
- **Separation of duty:**
 - Who examines and certifies that the transactions are performed correctly?
 - The system associates with each user a valid set of programs they can run and prevents unauthorized modifications, thus preserving integrity and consistency with the real world.

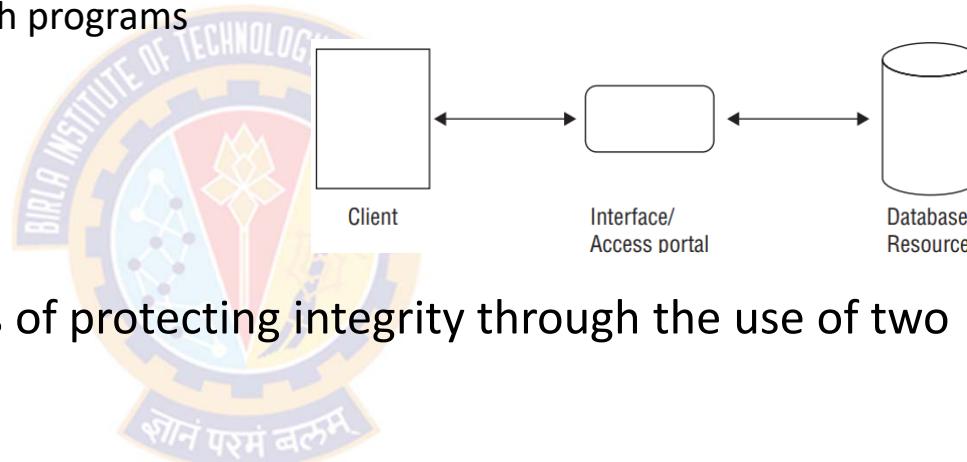


Clark-Wilson Integrity Model



Two Principles

- In CWM, subjects do not have direct access to objects
 - Objects can be accessed only through programs



- CWM provides an effective means of protecting integrity through the use of two principles:
 - Well-formed transactions:
 - A user should not manipulate data arbitrarily, but only in constrained ways that preserve or ensure the integrity of the data.
 - Separation of duty among users:
 - Any person permitted to create or certify a well-formed transaction may not be permitted to execute it (at least against production data)

Clark-Wilson Integrity Model



Entities

- The CWM defines data as:
 - Constrained Data Items (CDIs)
 - Is any data item whose integrity is protected by the security model
 - Unconstrained Data Items (UDIs)
 - Any data item whose integrity is not protected by the security model
 - Any data that is to be input and hasn't been validated, or any output. E.g., a simple text file
- The CWM also defines two sets of procedures:
 - Integrity verification procedures (IVPs)
 - Procedures that ensure CDIs conform to the integrity constraints at the time the IVPs are run
 - Transformation procedures (TPs)
 - Are the only procedures that are allowed to modify a CDI
 - Procedures that change the state of the data in the system from one valid state to another
 - TPs implement well-formed transactions

Clark-Wilson Integrity Model



Certification and Enforcement Rules

- The CWM enforces integrity by means of **certification rules** and **enforcement rules** on TPs
- Certification rules
 - are security policy restrictions on the behavior of Integrity verification procedure (IVPs) and Transformation procedures (TPs)
- Enforcement rules
 - are built-in system security mechanisms that achieve the objectives of the certification rules

Clark-Wilson Integrity Model



Certification and Enforcement Rules

- CR1: All IVPs must ensure that CDIs are in a valid state when the IVP is run
- CR2: All TPs must be certified as integrity-preserving
- CR3: Assignment of TPs to users must satisfy separation of duty
- CR4: The operation of TPs must be logged
- CR5: TPs executing on UDIs must result in valid CDIs
- ER1: Only certified TPs can manipulate CDIs
- ER2: Users must only access CDIs by means of TPs for which they are authorized
- ER3: The identity of each user attempting to execute a TP must be authenticated

Clark-Wilson Integrity Model



Certification and Enforcement Rules

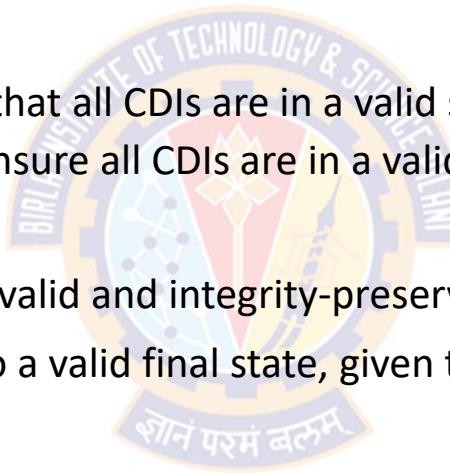
- Certification Rules 1 & 2

- CR1:

- All IVPs must properly ensure that all CDIs are in a valid state at the time the IVP is run.
 - When any IVP is run, it must ensure all CDIs are in a valid state

- CR2:

- All TPs must be certified to be valid and integrity-preserving
 - That is, they must take a CDI to a valid final state, given that it is in a valid state to begin with



Transformation Procedures (TPs)

Integrity verification procedures (IVPs)

Constrained Data Items (CDIs) = Data subject to integrity controls

Clark-Wilson Integrity Model



Certification and Enforcement Rules

- Enforcement Rules 1 & 2

- ER1

- The system must maintain the list of certified relations specified in CR2 and must ensure that only TPs certified to run on a CDI manipulate that CDI.

- ER2

- Users must only access CDIs by means of TPs for which they are authorized
 - The system must associate a user with each TP and set of CDIs
 - The system must maintain a list of relations of the form (`UserID, TPi, (CDIa, CDIb, CDIc, ...)`), which relates a user, a TP, and the data objects that TP may reference on behalf of that user
 - The TP may access those CDIs on behalf of the associated user
 - The TP cannot access that CDI on behalf of a user not associated with that TP and CDI

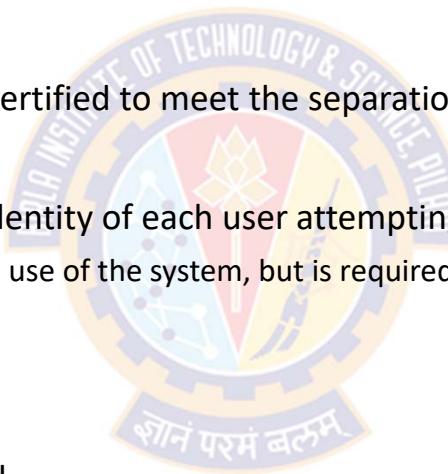
Clark-Wilson Integrity Model



Certification and Enforcement Rules

- Users and Rules

- CR3
 - The list of relations in ER2 must be certified to meet the separation of duty requirement.
- ER3
 - The system must authenticate the identity of each user attempting to execute a TP.
 - Authentication not required before use of the system, but is required before manipulation of CDIs (requires using TPs)



- Logging

- CR4
 - The operation of TPs must be logged
 - All TPs must be certified to write to an append-only CDI (the log)
 - All TPs must append enough information necessary to reconstruct the operation
 - Auditor needs to be able to determine what happened during reviews of transactions

Clark-Wilson Integrity Model

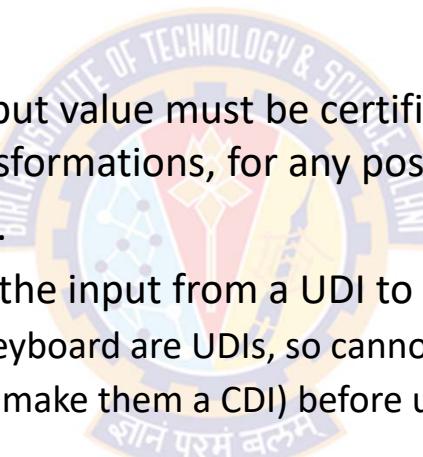


Certification and Enforcement Rules

- Handling Untrusted Input

- CR5

- Any TP that takes a UDI as an input value must be certified to perform only valid transformations, or else no transformations, for any possible value of the UDI
 - Typically, this is an edit program.
 - The transformation should take the input from a UDI to a CDI, or the UDI is rejected.
 - In bank, numbers entered at keyboard are UDIs, so cannot be input to TPs
 - TPs must validate numbers (to make them a CDI) before using them; if validation fails, TP rejects UDI



Clark-Wilson Integrity Model

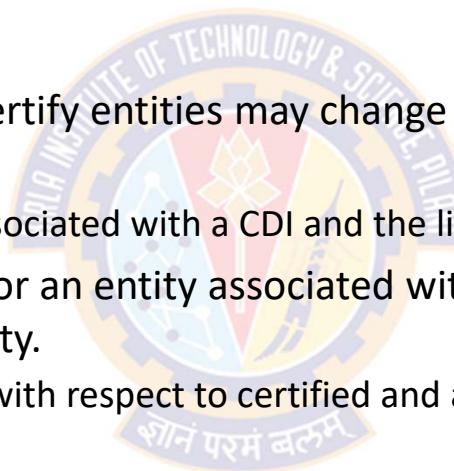


Certification and Enforcement Rules

- Separation of Duty Model

- ER4

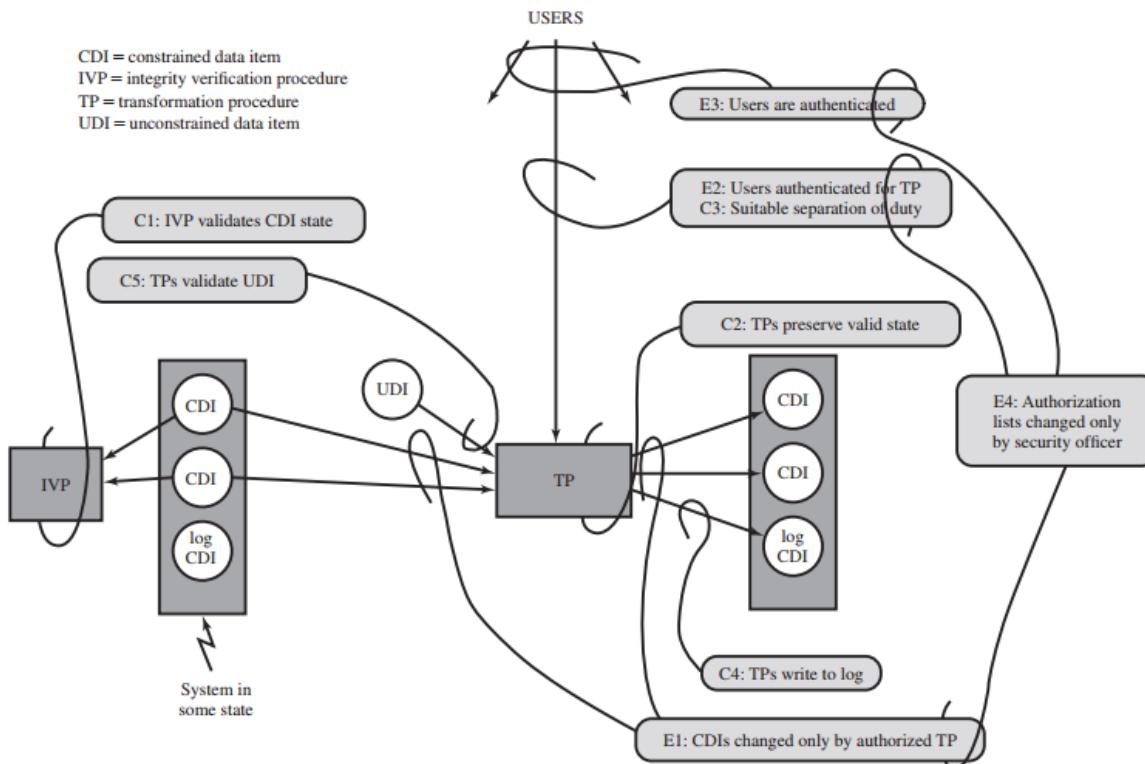
- Only the agent permitted to certify entities may change the list of such entities associated with other entities:
 - Specifically, the list of TPs associated with a CDI and the list of users associated with a TP
 - An agent that can certify a TP or an entity associated with that TP may not have any execute rights with respect to that entity.
 - Enforces separation of duty with respect to certified and allowed relations



Clark-Wilson Integrity Model



Certification and Enforcement Rules





Availability Policies

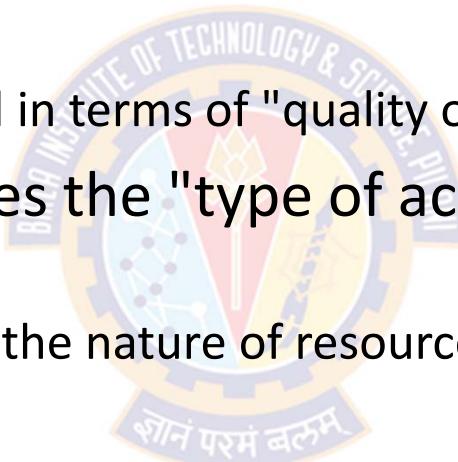


Availability Policies



Overview

- An availability policy ensures that a resource can be accessed in some way in a timely fashion
 - Availability is often expressed in terms of "quality of service."
- An availability policy defines the "type of access" and what a "timely fashion" means
 - "Timely fashion" depends on the nature of resource, the goals of subject using it



Availability Policies



Overview

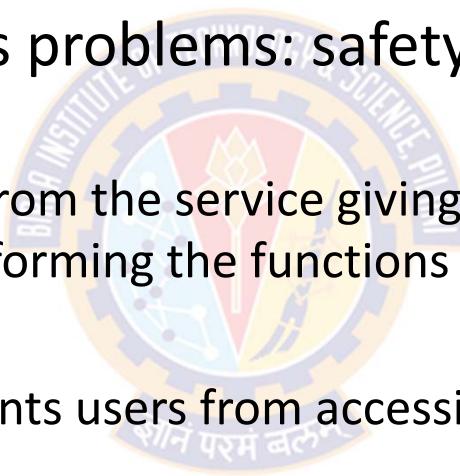
- Example_1:
 - A commercial website selling merchandise will need to display details of items for customer requests in a matter of seconds or, at worst, a minute
 - The goal of the customer is to see what the website is selling, and the goal of the site is to make information available to the customer
 - However, the site does not want customers to alter prices displayed on the website, so there is no availability for altering information
- Example_2:
 - A website enabling students to upload homework must allow some alterations (students must be able to upload their homework, possibly multiple times per assignment) quickly and no access for the students to read other students' assignments.

Availability Policies



Safety and Liveness

- When a resource or service is not available, a denial of service occurs
- This is related to two types problems: safety and liveness
- Safety problem
 - A denial of service resulting from the service giving incorrect responses
 - That is, the service is not performing the functions that the client is expecting
- Liveness problem
 - A denial of service that prevents users from accessing the service is a liveness problem
- But other problems can cause a denial of service, such as assignment of inadequate resources to a process



Availability Policies



Mechanisms to support availability

- Two requirements under which mechanisms are used to support availability:
 - a) in general
 - b) as a security requirement
- The difference between the two lies in the assumptions underlying the failures
 - That is, under what circumstances failures can occur





Availability Policies

Mechanisms to support availability

- Mechanisms to support availability in general
 - The failures occur naturally over time due to usage
 - Lack of accessibility is modeled using an average case, following a statistical model
 - For example:
 - The failure rates of disk drives depends upon many factors such as the age, the manufacturer, and environment and can be statistically modeled, although the precise model to be used is unclear
- Mechanisms used to support availability as a security requirement
 - Lack of availability assumes worst-case
 - Here, an adversary deliberately tries to make the resource or information unavailable
 - Because attackers induce this condition, models used in computer security describe failures that are nonrandom, and indeed may well be non-statistical

Deadlock



Overview

- A *deadlock* is a state in which some set of processes block each waiting for another process in the set to take some action.
- Deadlock can occur if four conditions hold simultaneously:
 - *Mutual exclusion*: At least one resource must be held in a non-sharable mode; If any other process requests this resource, then that process must wait for the resource to be released
 - *Hold and wait*: A process must be simultaneously holding at least one resource and waiting for at least one resource that is currently being held by some other process
 - *No preemption*: Once a process is holding a resource (i.e. once its request has been granted), then that resource cannot be taken away from that process until the process voluntarily releases it
 - *Circular wait*: A set of entities must be holding resources such that each entity is waiting for a resource held by another entity in the set
- Usually not due to an attack

Deadlock



Methods of Handling Deadlocks

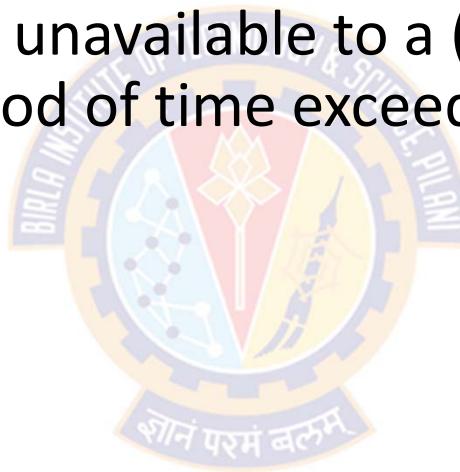
- *Prevention:* prevent 1 of the 4 conditions from holding
 - Do not allow the system to get into a deadlocked state.
 - Do not acquire resources until all needed ones are available
 - When needing a new resource, release all held
- *Avoidance:* ensure process stays in state where deadlock cannot occur
 - *Safe state:* deadlock can not occur
 - *Unsafe state:* may lead to state in which deadlock can occur
 - Abort a process or preempt some resources when deadlocks are detected
- *Detection:* allow deadlocks to occur, but detect and recover
 - If deadlocks only occur once a year or so, it may be better to simply let them happen and reboot as necessary than to incur the constant overhead and system performance penalties associated with deadlock prevention or detection
 - This is the approach that both Windows and UNIX take

Denial of Service



Overview

- A denial of service occurs when a group of authorized users of a service makes that service unavailable to a (disjoint) group of authorized users for a period of time exceeding a defined maximum waiting time



Denial of Service



Overview

- What do we mean by "authorized user"?
- If a user is not authorized, then in theory access control mechanisms that protect the server will block the unauthorized users from accessing the server
- But in practice, the access control mechanisms may be ineffective
 - E.g., An intruder may compromise a user's account to gain access to a server
- The policy controlling access to a network server may be unworkable
- For example:
 - A policy states that only customers interested in the products sold may access the server—but the access control mechanisms could not tell whether a remote user accessing the server was interested in the products, or trying to block access by others
- Hence the first "group of authorized users" is simply the group of users with access to the service, whether the security policy grants them access or not.

Availability and Network Flooding



Example: SYN Flood Attack

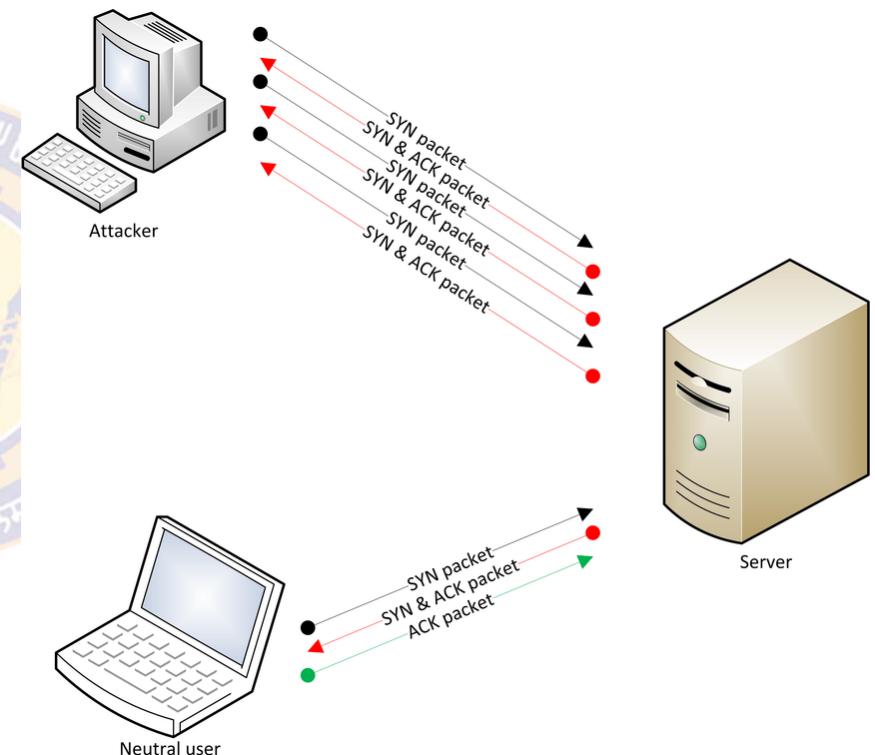
- Access over Internet must be unimpeded
- In flooding attacks attackers try to overwhelm system resources
- If many sources flood a target, it's called *distributed denial of service attack* (DDoS)
- The SYN flood is a type of most common type of flooding attack
 - SYN is short for "synchronize"
- It is based on the initiation of a connection using the TCP protocol
- A SYN flood sends a series of "SYN" messages to a computer (E.g., web server)

Availability and Network Flooding



Example: SYN Flood Attack

- In a normal case, the user sends the SYN packet to the target
- When a server receives a SYN request, it responds with a SYN-ACK (synchronize acknowledge) message
- The source then responds with an ACK (acknowledge) message that establishes a connection between the two systems

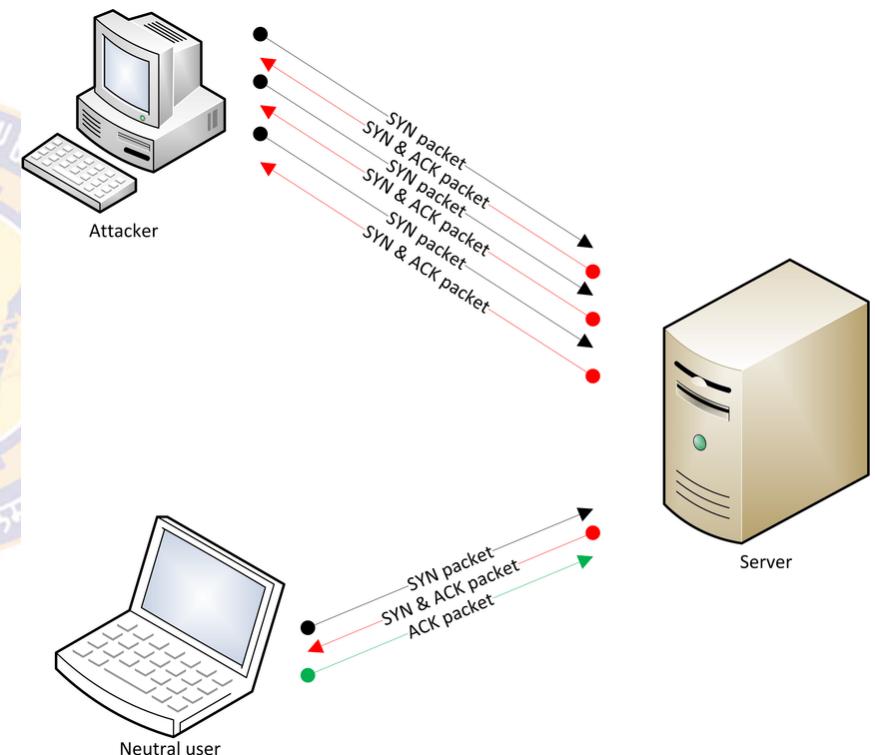
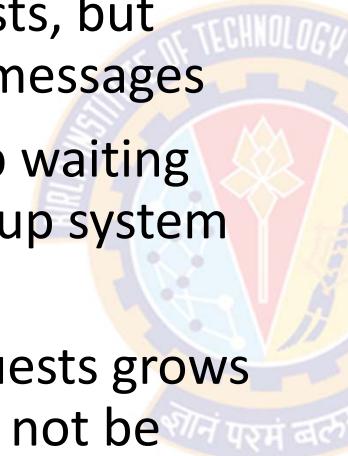


Availability and Network Flooding



Example: SYN Flood Attack

- In a SYN flood attack, a computer sends a large number of SYN requests, but does not send back any ACK messages
- Therefore, the server ends up waiting for multiple responses, tying up system resources
- If the queue of response requests grows large enough, the server may not be able respond to legitimate requests
- This results in a slow or unresponsive server

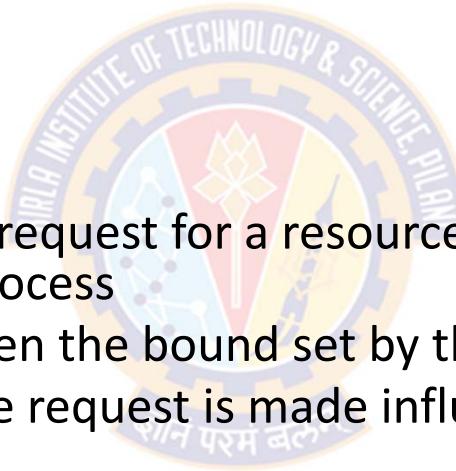




Denial of Service

Components of Denial of Service Models

- Denial of service models have two essential components
 - waiting time policy
 - user agreement
- **Wait time policy**
 - Controls the time between a request for a resource and the allocation of that resource to the requesting process
 - A denial of service occurs when the bound set by this policy is exceeded
 - The environment in which the request is made influences the policy
 - Example:
 - The acceptable waiting time for a pacemaker to take action affecting a patient's heart beating is considerably different than the acceptable waiting time for a purchase from an Internet website to be acknowledged.





Denial of Service

Components of Denial of Service Models

- **User agreement**

- Establishes constraints a process ("user") must meet in order to ensure service
- These are designed to ensure that a process will receive service within the waiting time
- For example:
 - Consider parallel processes accessing a mutually exclusive resource
 - A user agreement for this situation would be that once a process acquires the resource, it must (eventually) release that resource
 - When released, there are enough unallocated resources to enable a process waiting for those resources to proceed

Denial of Service



Components of Denial of Service Models

- These two components (wait time policy & user agreement) in combination ensure that a process meets the conditions needed to receive the resources it needs and not create a denial of service
- It will receive those resources after an acceptable waiting time
- Thus, the process can proceed and not itself be denied service
- Two types of models that formalize these notions are:
 - Constraint-based models
 - State-based models



Trust Models

शोनं परमं बलम्

Trust Models



Overview

- Integrity Models
 - Integrity models deal with changes to entities
 - State conditions under which changes preserve those properties that define "**integrity**"
 - Do not deal with the **confidence** one can have in the initial values or settings of that entity
 - That is, integrity models deal with the preservation of **trustworthiness**, but not with the initial evaluation of whether the contents can be trusted
- Trust models
 - Provide information about the **credibility** of data and entities
 - Deal with **confidence** one can have in the initial values or settings
 - Are concerned with the *initial* evaluation of whether data can be trusted
 - Because trust is subjective, trust models typically express the trustworthiness of one entity in terms of another

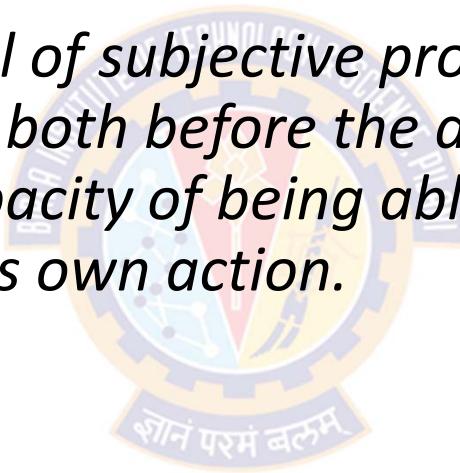
Trust Models



Definition of Trust

A *trusts* B if

A believes, with a level of subjective probability, that B will perform a particular action, both before the action can be monitored (or independently of the capacity of being able to monitor it) and in a context in which it affects A's own action.



Trust Models



Definition of Trust

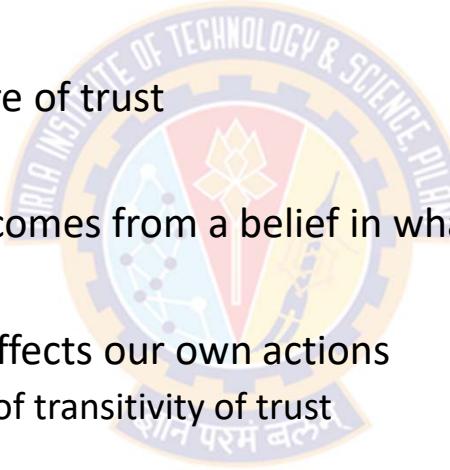
- The above definition involves actors, but it also can apply to the credibility of information
- If you ask whether the data is "trusted" is really asking if a reader of the data believes to some level of subjective probability that the entity providing the data
 - (i) obtained it accurately and without error, and is
 - (ii) providing it accurately and without error
- In the above definition, the reader is A, the provider is B, and the "particular action" is that of gathering and providing the data

Trust Models



Definition of Trust

- This definition captures three important points about trust:
 - First
 - it includes the subjective nature of trust
 - Second
 - it captures the idea that trust comes from a belief in what we do not, or cannot, monitor
 - Third
 - the actions of those we trust affects our own actions
 - This also leads to the notion of transitivity of trust

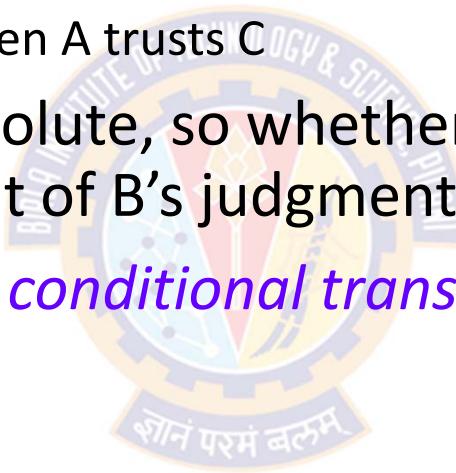


Trust Models



Transitivity of Trust

- *Transitivity of trust:*
 - if A trusts B and B trusts C, then A trusts C
- In practice, trust is not absolute, so whether trust is transitive depends on A's assessment of B's judgment
- This leads to the notion of *conditional transitivity of trust*, which says that A can trust C when:
 - B recommends C to A
 - A trusts B's recommendations
 - A can make judgments about B's recommendations; and
 - Based on B's recommendation, A may trust C less than B does.



Trust Models



Trust Propagation

- *Direct trust:*
 - A trusts C because of A's observations and interactions
- *Indirect trust:*
 - A trusts C because A accepts B's recommendation
- *Trust Propagation:*
 - Indirect trust may take a path involving many intermediate entities
 - This is called trust propagation because the trust propagates among many entities

Trust Models



Types of Beliefs Underlying Trust

- Castelfranchi and Falcone argue that trust is a cognitive property,
 - so only agents with goals and beliefs can trust another agent
- This requires the trusting agent, A, to estimate risk and then decide, based on her willingness to accept (or not accept) the risk, whether to rely on the one to be trusted, B
- This estimation arises from social and technological sources, as well as A's observations and her taking into account recommendations
- They identify several belief types:



Trust Models

Types of Beliefs Underlying Trust

- *Competence*: A believes B to be competent to aid A in reaching her goal
- *Disposition*: A believes the B will actually do what A needs to reach her goal
- *Dependence*: A believes she needs what B will do, depends on what B will do, or that it is better for A to rely on B than not to rely on him
- *Fulfillment*: A believes goal will be reached
- *Willingness*: A believes B has decided to do what A wants
- *Persistence*: A believes B will not change his mind before doing what A wants
- *Self-confidence*: A believes that B knows he can take the action A wants



Thank You!



BITS Pilani
Pilani | Dubai | Goa | Hyderabad

Cyber Security

Introduction to Networks and the Internet

Dr. Ramakrishna Dantu

Associate Professor, BITS Pilani

Disclaimer and Acknowledgement



Disclaimer

- The content for these slides has been obtained from books and various other source on the Internet
- I here by acknowledge all the contributors for their material and inputs.
- I have provided source information wherever necessary
- I have added and modified the content to suit the requirements of the course

Agenda

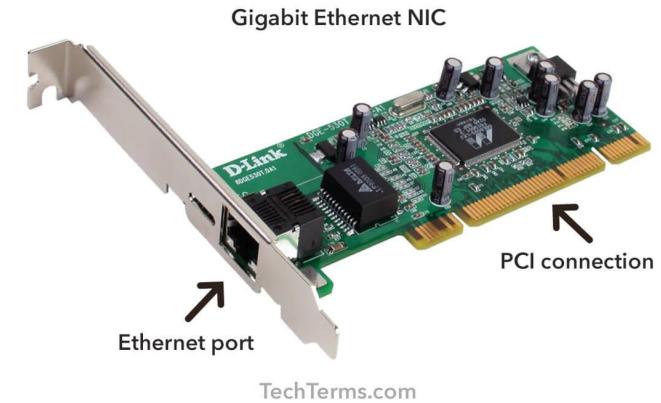
- Introduction
- Network Basics
- How the Internet Works
- History of the Internet
- Basic Network Utilities
- Other Network Devices
- Advanced Network Communications Topics:
 - Network communication types
 - Types of Networks
 - OSI Model
 - Network Protocols



Network Basics

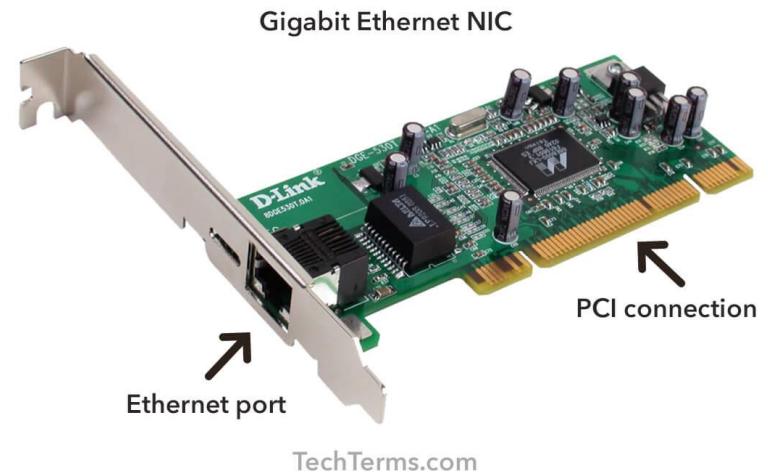
Overview

- Communication among computers
 - Requires connecting them physically through cables or wirelessly
 - Cables are plugged either directly to another computer or into a **device**
 - This device will, in turn, connects to several other computers
- Network Interface Card (NIC)
 - Wireless communication relies on a physical device for transmitting the data
 - This device is called *network interface card* (NIC)



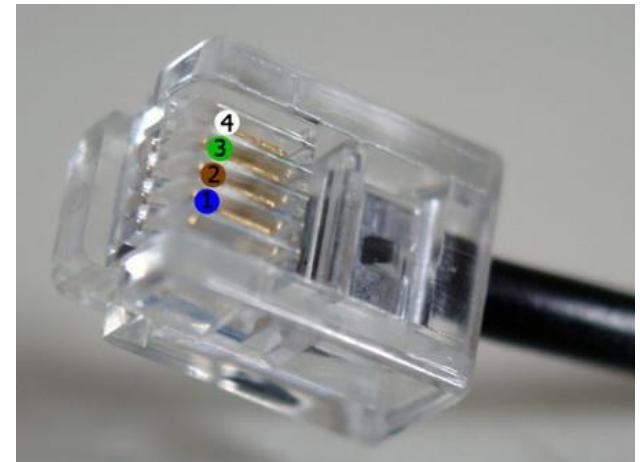
Overview

- Connection slot (Ethernet port)
 - If the connection is through a cable, the part of the NIC that is external to the computer has a **connection slot** that looks like a telephone jack, only slightly bigger
- Radio signals
 - Wireless networks also use a NIC
 - Rather than a slot for connecting a cable, NIC uses radio signals to transmit to a nearby wireless router or hub
- Antenna
 - Wireless routers, hubs, and NICs have an antenna to transmit and receive signals



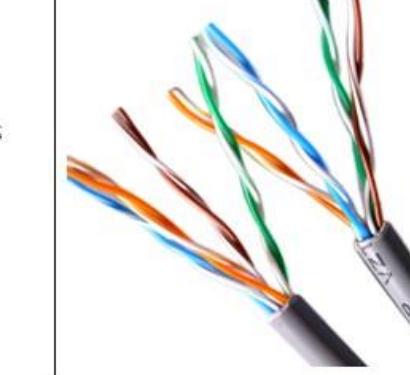
The Physical Connection: Local Networks

- RJ-45
 - The cable connection used with wired NICs is called an RJ-45 connection
 - RJ = Registered Jack, an international industry standard
- RJ-11
 - In contrast to the computer's RJ-45 jacks, standard telephone lines use RJ-11 jacks
- RJ-45 Vs. RJ-11
 - The key difference between jacks is the number of wires in the connector (also called the [terminator](#))
 - Phone lines (RJ-11) have four wires (some have six wires), RJ-45 connectors have eight wires
- This standard connector jack must be on the end of the cable



The Physical Connection: Local Networks

- Cat 5 or Cat 6 Cable
 - The cable used in most networks today is a Category 5 or 6 cable abbreviated as Cat 5 or Cat 6 cable
- Unshielded Twisted-Pair (UTP)
 - The cable used in connecting computers is referred to as *unshielded twisted-pair* (UTP) cable
 - The wires in the cable are in pairs, twisted together without additional shielding
- Shielded Twisted-Pair (STP)
 - There are other types of cable such as *shielded twisted-pair* (STP), but UTP is most commonly used

Cat5e VS Cat6		
Product Name	Cat5e UTP Cable	Cat6 UTP Cable
Speed	10BASE-T, 100BASE-TX(Fast Ethernet), 1000BASE-T (Gigabit Ethernet)	10BASE-T, 100BASE-TX(Fast Ethernet), 1000BASE-T (Gigabit Ethernet), 10G BASE-T (10-Gigabit Ethernet)
Frequency	100 MHz	250 MHz
Performance	Good	Better

The Physical Connection: Local Networks

- Table summarizes various categories of cable and their uses.

Cable Types and Uses		
Category	Specifications	Uses
1	Low-speed analog (less than 1MHz)	Telephone, doorbell
2	Analog line (less than 10MHz)	Telephone
3	Up to 16MHz or 100Mbps (megabits per second)	Voice transmissions
4	Up to 20MHz/100Mbps	Data lines, Ethernet networks
5	100MHz/100Mbps	Most common a few years ago, still widely used
6	1000Mbps (some get 10Gbps)	Most common type of network cable
6a	10Gbps	High-speed networks
7	10Gbps	Very high-speed networks
8	40Gbps	Not yet commonly found

The Physical Connection: Local Networks

- Each subsequent category of cable is somewhat faster and more robust than the last
- Although Cat 4 can be used for networks, it almost never is used, as it is simply slower, less reliable, and an older technology
- We usually see Cat 5 cable and, increasingly, Cat 6
- We are focusing on UTP because that is what is found most often
- Other types of cable such as shielded twisted-pair (STP), but they are not nearly as common as UTP

The Physical Connection: Local Networks

- A key specification for cables is speed
 - measured in Mbps (megabits per second)
- Now a days, Gbps (gigabits per second) speeds are becoming more common
- Data specification for each cable indicated in the table is the maximum that the cable can handle
 - This is called *bandwidth* of a cable
 - E.g., a Cat 5 cable can transmit up to 100 mega (million) bits per second
- If multiple users simultaneously transmit data on a network, that traffic uses up bandwidth rather quickly
 - E.g., a scanned picture can easily take 2 megabytes (2 million bytes, or 16 million bits) or much larger
 - Streaming media, such as videos, are most demanding in terms of bandwidth

The Physical Connection: Local Networks

- Connecting two computers simply requires a cable to go directly from one computer to another
 - What about more than 2 computers or 100 computers?
- Three devices that can help accomplish this task:
 - The hub
 - The switch, and
 - The router
- These devices use Cat 5 or Cat 6 cable with RJ-45 connectors

The Hub

- A hub is a small electronic device into which network cables are plugged
- It can have 4 or more (commonly up to 24) RJ-45 jacks, each called a port
- A hub can connect as many computers as it has ports
 - E.g., an 8-port hub can connect eight computers
- Stacking
 - We can also connect one hub to another
 - This is referred to as "*stacking*" hubs



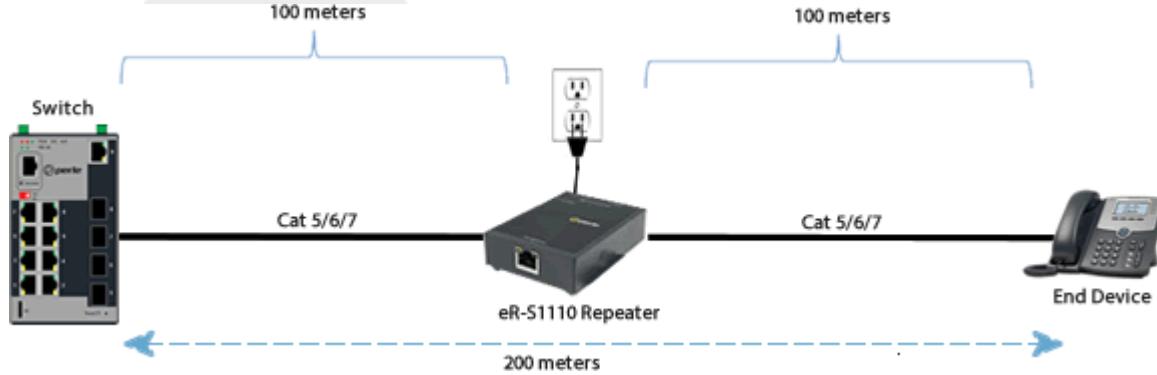
Downside of Hubs

- If a packet (a unit of data transmission) is sent from one computer to another
 - a copy of that packet is actually sent out from every port on the hub
- These copies leads to a lot of unnecessary network traffic
- This occurs because the hub, being a very simple device, has no way of knowing where a packet is supposed to go
- Therefore, it simply sends copies of the packet out all of its ports
- True hubs no longer exist, what we are really getting is a *switch*.



The Repeater

- Is a device used to boost signal
- Basically if the cable needs to go further than the maximum length (which is 100 meters for UTP), then we need a repeater
- There are two types of repeaters: **amplifier** and **signal**
- Amplifier repeaters simply boost the entire signal they receive, including any noise
- Signal repeaters regenerate the signal, and thus don't rebroadcast noise.



The Switch

- A switch is basically an intelligent hub
- It works and looks exactly like a hub
- When a switch receives a packet, it sends that packet only out the port for the computer to which it needs to go
- A switch is essentially a hub that is able to determine where a packet is being sent



The Router

- A router is used to connect two or more *networks*
- A router:
 - a) is similar in concept to a hub or switch, as it does relay packets;
 - b) is far more sophisticated
- Routers can be programmed and controlled how they relay packets
- Most routers have interfaces that allow us to configure them
- The specifics of router programming differs from vendor to vendor
- Unlike using a hub or switch, the two networks connected by a router are still separate networks



Faster Connection Speeds

Internet Connection Types

Connection Type	Speed	Details
DS0	64Kbps	Standard phone line.
ISDN	128Kbps	Two DS0 lines working together to provide a high-speed data connection.
T1	1.54Mbps	Twenty-four DS0 lines working as one. Twenty-three carry data, and one carries information about the other lines. This type of connection has become common for schools and businesses.
T3	43.2Mbps	672 DS0 lines working together. This method is the equivalent of 28 T1 lines.
OC3	155Mbps	All OC lines are optical and do not use traditional phone lines. OC3 lines are quite fast and very expensive. They are often found at telecommunications companies.
OC12	622Mbps	The equivalent of 336 T1 lines, or 8,064 phone lines.
OC48	2.5Gbps	The equivalent of four OC12 lines.

Wireless

- The Institute of Electrical and Electronics Engineers (IEEE) standard 802.11 provides guidelines for wireless networking
- Various letter designations are used to denote different wireless speeds
- The various wireless speeds, starting from the oldest to the most recent, are listed here

Wireless

Designation	Description
802.11a	<ul style="list-style-type: none">This was the first widely used Wi-Fi; it operated at 5GHz and was relatively slow
802.11b	<ul style="list-style-type: none">This standard operated at 2.4GHz and had an indoor range of 125 feet with a bandwidth of 11Mbps
802.11g	<ul style="list-style-type: none">There are still many of these wireless networks in operationWe can no longer purchase new Wi-Fi access points that use 802.11g.This standard includes backward compatibility with 802.11b.802.11g has an indoor range of 125 feet and a bandwidth of 54Mbps
802.11n	<ul style="list-style-type: none">This standard was a tremendous improvement over preceding wireless networksIt provides a bandwidth of 100Mbps to 140Mbps and operates at frequencies of 2.4GHz or 5.0GHz over an indoor range of up to 230 feet
IEEE 802.11n-2009	<ul style="list-style-type: none">This technology provides a bandwidth of up to 600Mbps with the use of four spatial streams at a channel width of 40MHzIt uses multiple-input multiple-output (MIMO), in which multiple antennas coherently resolve more information than is possible using a single antenna

Wireless

Designation	Description
IEEE 802.11ac	<ul style="list-style-type: none">This standard was approved in January 2014It has a throughput of up to 1Gbps and at least 500MbpsIt uses up to 8 multiple-input multiple-output (MIMO)
IEEE 802.11ad Wireless Gigabyte Alliance	<ul style="list-style-type: none">Supports data transmission rates up to 7GbpsThis is more than 10 times faster than the highest 802.11n rate
IEEE 802.11af	<ul style="list-style-type: none">Also referred to as "White-Fi" and "Super Wi-Fi,"This standard was approved in February 2014It allows WLAN operation in TV white space spectrum in the VHF and UHF bands between 54MHz and 790MHz.
IEEE 802.11aj	<ul style="list-style-type: none">It is a rebranding of 802.11adIt is used in the 45GHz unlicensed spectrum available in some regions of the world (specifically China).

Securing Wi-Fi

- The methods for securing Wi-Fi have evolved over the years
 - First there was Wired Equivalent Privacy (WEP)
 - Next, Wi-Fi Protected Access (WPA)
 - Next, Wi-Fi Protected Access2 (WPA2)
 - Currently, Wi-Fi Protected Access3 (WPA3)

Securing Wi-Fi

- Wired Equivalent Privacy (WEP)
 - WEP uses the stream cipher RC4 algorithm to secure the data and a CRC-32 checksum for error checking
 - Standard WEP (known as WEP-40) uses a 40-bit key with a 24-bit initialization vector (IV) to effectively form 64-bit encryption
 - 128-bit WEP uses a 104-bit key with a 24-bit IV
 - Because RC4 is a stream cipher, the same traffic key must never be used twice
 - The purpose of an IV, which is transmitted as plain text, is to prevent any repetition
 - but a 24-bit IV is not long enough to ensure this on a busy network
 - The way its IV is used also opens WEP to a related key attack
 - For a 24-bit IV, there is a 50% probability the same IV will repeat after 5000 packets

Securing Wi-Fi

- Wi-Fi Protected Access (WPA)
 - WPA uses Temporal Key Integrity Protocol (TKIP)
 - TKIP is a 128-bit per-packet key
 - That is, it dynamically generates a new key for each packet
- Wi-Fi Protected Access (WPA2)
 - WPA2 is based on the IEEE 802.11i standard
 - Provides the Advanced Encryption Standard (AES) using the Counter Mode-Cipher Block Chaining (CBC)-Message Authentication Code (MAC) Protocol (CCMP)
 - This provides data confidentiality, data origin authentication, and data integrity for wireless frames

Securing Wi-Fi

- Wi-Fi Protected Access (WPA3)
 - WPA3 requires attackers to interact with your Wi-Fi for every password guess they make, making it much harder and time-consuming to crack
 - However, with WPA3's "Wi-Fi Easy Connect," you can connect a device by merely scanning a QR code on your phone
 - One of the important new security features is that with WPA3, even open networks will encrypt your individual traffic

Bluetooth

- The name comes from king Harald "Bluetooth" Gormsson, a tenth-century Danish king who united the tribes of Denmark.
 - There are different explanations for the king's nickname
 - One is that he had a bad tooth that was blue
 - Another is that he was often clothed in blue
- The idea behind the Bluetooth technology is that it unites communication protocols
- It is a short-distance radio using the 2.4GHz to 2.485GHz frequency
- The IEEE standardized Bluetooth as IEEE 802.15.1, but it no longer maintains the standard
 - This standard enables devices to discover other Bluetooth devices within range
- The speed and range of Bluetooth depends on the version

Version	Bandwidth	Range
3.0	25Mbps	10 meters (33 feet)
4.0	25Mbps	60 meters (200 feet)
5.0	50Mbps	240 meters (800 feet)

Other Wireless Protocols

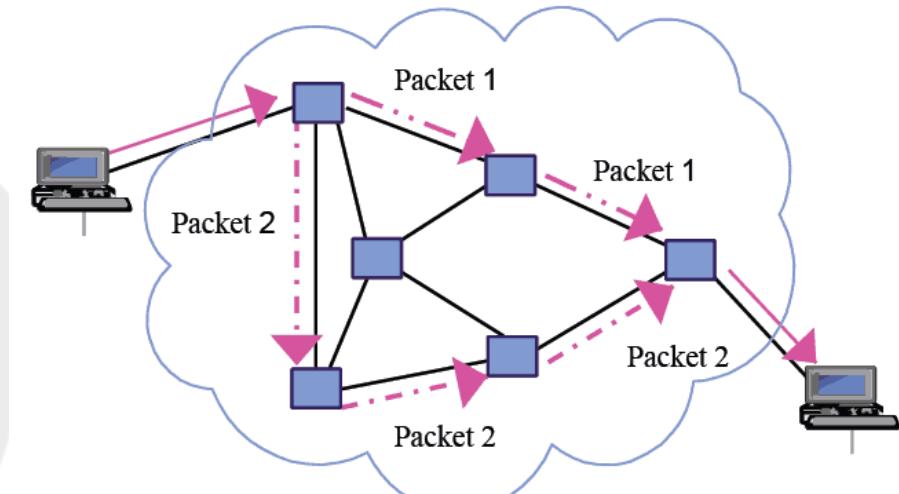
- ANT +:
 - This wireless protocol is often used with sensor data such as in bio sensors or exercise applications
- ZigBee:
 - This standard was developed by a consortium of electronics manufacturers for mainly residential applications of wireless devices related to appliances and security
 - It is based on the 802.15.4 standard
 - This standard is represented by the name "ZigBee" rather than a number
 - The term ZigBee is used similar to the way the term Wi-Fi is used
- Z-Wave:
 - This wireless communications protocol is used primarily for home automation
 - Uses a low-energy radio for appliance-to-appliance communication using a mesh network



Data Transmission

Overview

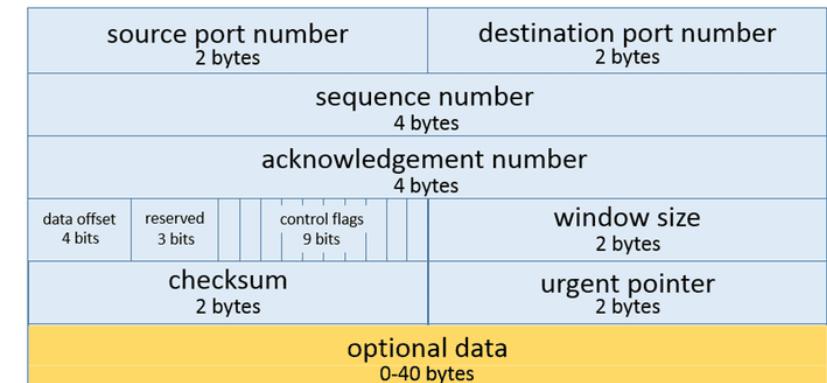
- How is data actually transmitted in the networks?
- To transmit data, a packet is sent
- The basic purpose of a cable is to transmit packets from one machine to another
- It does not matter whether that packet is part of a document, a video, an image, or just some internal signal from the computer



Overview

- Now, what exactly is a packet?
- A packet is a certain number of bytes divided into a header and a body
- The header is 20-60 bytes at the beginning of the packet that tells where the packet is coming from, where it is going, and more
- The body contains the actual data, in binary format
- The routers and switches read the header portion of the packets that come to them and determine where the packet should be sent

Transmission Control Protocol (TCP) Header
20-60 bytes



Protocols

- There are different types of network communications for different purposes
- These network communications are called *protocols*
- A *protocol* is, essentially, an agreed-upon method of communication
- In fact, this definition is exactly how the word protocol is used in standard, non-computer usage, too
- Each protocol has a specific purpose and normally operates on a certain port

Protocols

- Some of the most important, and most commonly used, protocols are listed in table below (see next slide)
- All of these protocols are part of a suite of protocols referred to as TCP/IP (Transmission Control Protocol/Internet Protocol)
- But no matter what protocol is used, all communication on networks takes place via packets
- These packets are transmitted according to certain protocols, depending on the type of communication that is occurring

Data Transmission

innovate

achieve

lead

Protocols

Protocol	Purpose	Port(s)
FTP (File Transfer Protocol)	For transferring files between computers	20 & 21
TFTP (Trivial File Transfer Protocol)	A quicker but less reliable form of FTP	69
SSH (Secure Shell)	Used to securely connect to a remote system	22
Telnet	Used to remotely log on to a system. You can then use a command prompt or shell to execute commands on that system. Popular with network administrators	23
SMTP (Simple Mail Transfer Protocol)	Sends email	25
Whois	A query and response protocol that provides information about the registered Domain Names, an IP address block, Name Servers, etc.	43
DNS (Domain Name System)	Translates URLs into web addresses.	53
HTTP (Hypertext Transfer Protocol)	Displays web pages	80
POP3 (Post Office Protocol version 3)	Retrieves email	110

Data Transmission

innovate

achieve

lead

Protocols

Protocol	Purpose	Port(s)
NNTP (Network News Transfer Protocol)	Used for network newsgroups (Usenet newsgroups). You can access these groups over the Web via www.google.com and selecting the Groups tab	119
NetBIOS	An older Microsoft protocol that is for naming systems on a local network	137, 138, or 139
IMAP (Internet Message Access Protocol)	More advanced protocol for receiving email. Widely replacing POP3	143
IRC (Internet Relay Chat)	Used for chat rooms	194
SMB (Server Message Block)	Used for Windows Active Directory	445
HTTPS	Encrypted HTTP; used for secure websites	443
SMTPS	Simple Mail Transfer Protocol Secure; Encrypted SMTP	465
POP3S	Post Office Protocol version 3 Secure; Encrypted POP3	995
IMAPS	Internet Message Access Protocol Secure; Encrypted IMAP	993

Ports

- In a physical sense, ports are the connection locations on the back of our computer
 - E.g., serial ports, parallel ports, and RJ-45 and RJ-11 ports
- In networking terms, a port is a connection point
- It is a numeric designation for a particular pathway of communications
- It can be thought of as a channel number on our television
- We may have one cable coming into our TV, but you can tune to a variety of channels

Ports

- Regardless of the type of computer or operating system, there are 65,535 network communications ports on our computer
- The combination of our computer's IP address and port number is referred to as a *socket*
- All network communication (regardless of the port used) comes into our computer via the connection on our NIC
- So, a network consists of computers connected to each other via cables, hubs, switches, or routers
- These networks transmit binary information in packets using certain protocols and ports



How Internet Works

How the Internet Works

innovate

achieve

lead

Overview

- The Internet is essentially a large number of networks that are connected to each other
- These networks are connected into main transmission lines called *backbones*
- The points where the backbones connect to each other are called *network access points* (NAPs)
- The Internet works exactly the same way as a local network
- It sends the same sort of data packets, using the same protocols
- When we log on to the Internet, we typically use an *Internet service provider* (ISP)
- The ISP has a connection either to the Internet backbone or to yet another provider that has a backbone
- So, logging on to the Internet is a process of connecting the computer to ISP's network, which is, in turn, connected to one of the backbones on the Internet

IP Addresses

- When tens of thousands of networks and millions of individual computers communicate,
 - how to ensure that the data packets go to the correct computer?
- This task is accomplished in much the same way as traditional "snail" letter mail is delivered to the right person: **via an address**
- In network communications, this address is referred to as an "IP" address
- An IP address can be IP version 4 or version 6

IPv4

- An IP address is a series of four values, separated by periods
 - E.g., 107.22.98.198
- Each of the three-digit numbers must be between 0 and 255
 - For example, an address of 107.22.98.466 is not a valid one
- These addresses are actually four binary numbers; we just see them in decimal format
- Each of these numbers (being a decimal representation of 8 bits), are often referred to as octets
- A 8-bit binary number converted to decimal format will be between 0 and 255
- So there are four octets in an IPv4 address
- This rule gives a total of over 4.2 billion possible IP addresses
- There are methods already in place to extend the use of addresses

How the Internet Works

innovate

achieve

lead

IPv4

- To extend the reach of the IPv4 address space, companies have turned to using **private IPv4** addresses through a public-to-private address translation technique known as network address translation (NAT).
- The public IP addresses are for computers connected to the Internet
- Public IP addresses cannot be duplicate
- A private IP address, such as one on a private company network, only has to be unique in that network
- Often network administrators use private IP addresses that begin with a 10, such as 10.102.230.17.

How the Internet Works



IPv4

- An ISP typically buys a pool of public IP addresses and assign them to us when we log on
- An ISP might own 1,000 public IP address and have 10,000 customers
- The ISP simply assigns an IP address to a customer when he logs on, and the ISP un-assigns the IP address when the customer logs off
- The IP address of a computer tells us a lot about that computer
- The first byte (or the first decimal number) in an address tells you to what class of network that machine belongs

How the Internet Works



IPv4

- Table below summarizes the five network classes

Network Classes

Class	IP Range for the First Byte	Use
A	0–126	Extremely large networks. No Class A network IP addresses are left. All have been used.
B	128–191	Large corporate and government networks. All Class B IP addresses have been used.
C	192–223	The most common group of IP addresses. Your ISP probably has a Class C address.
D	224–247	These are reserved for multicasting (transmitting different data on the same channel).
E	248–255	Reserved for experimental use.

How the Internet Works

innovate

achieve

lead

IPv4

- The IP range of 127 (not listed in the table) is reserved for testing
- The IP address of 127.0.0.1 designates the machine you are on, regardless of that machine's assigned IP address
- This address is often referred to as the *loopback address*
- That address is often used in testing our machine and our NIC
- In these network classes, one part of the address represents the network and the other part represents the node
- For example:
 - In Class A address, the first octet represents the network, and the remaining three represent the node
 - In Class B address, the first two octets represent the network, and the second two represent the node
 - In Class C address, the first three octets represent the network, and the last represents the node

How the Internet Works

innovate

achieve

lead

IPv4

- Special purpose IP addresses
 - IP 127.0.0.1, or the loopback address is used for referring to the network interface card of the machine we are on
 - Certain range of private IP addresses have been designated for use within networks
 - These cannot be used as public IP addresses but can be used for internal workstations and servers
 - 10.0.0.10 to 10.255.255.255
 - 172.16.0.0 to 172.31.255.255
 - 192.168.0.0 to 192.168.255.255
- The gateway router performs *network address translation* (NAT)
- NAT takes the private IP address on outgoing packets and replaces it with the public IP address of the gateway router
 - This allows the packet to be routed through the Internet

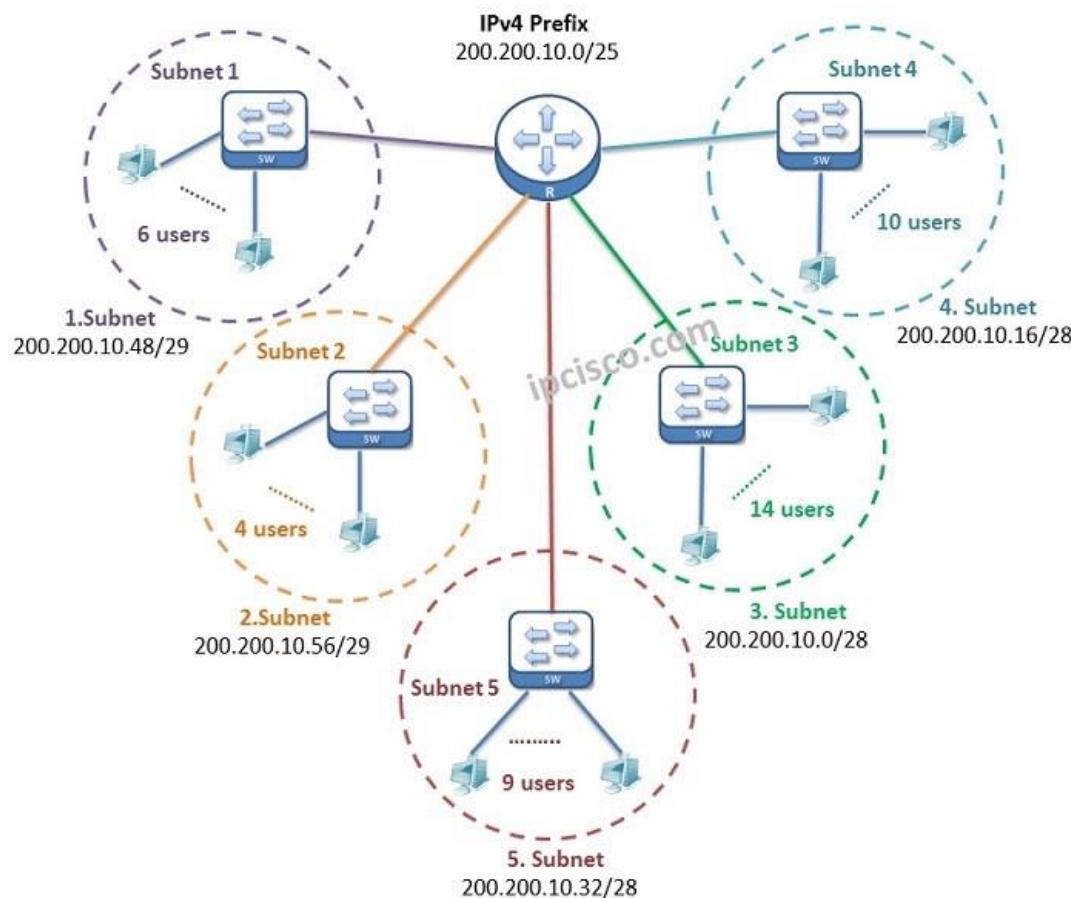
How the Internet Works

innovate

achieve

lead

Subnetting



- Subnetting is simply slicing a network into smaller portions
- For example, consider a network using the IP address 192.168.1.X (x being whatever the address is for the specific computer), then we have allocated 255 possible IP addresses
- If we wish to divide this IP into two separate subnetworks, subnetting is the way to go
- More technically, the subnet mask is a 32-bit number that is assigned to each host to divide the 32-bit binary IP address into network and node portions

How the Internet Works



Subnetting

- We already have a subnet mask even if you have not been subnetting
 - If we have a Class C IP address, then our network subnet mask is 255.255.255.0.
 - If we have a Class B IP address, then our subnet mask is 255.255.0.0.
 - If we have a Class A IP address, then our subnet mask is 255.0.0.0.
- The decimal value 255 converts to 11111111 in binary
- So we are literally "masking" the portion of the network address that is used to define the network, and the remaining portion is used to define individual nodes

Subnets/Hosts				
Network	Host	Host	Host	Host
255	.	0	.	0
Subnets/Hosts				
Network	Network	Host	Host	Host
255	.	255	.	0
Subnets/Hosts				
Network	Network	Network	Host	Host
255	.	255	.	0

Subnetting

- Now if we want fewer than 255 nodes in our subnet, then we need something like 255.255.255.240 for our subnet
- If we convert 240 to binary, it is 11110000
- That means the first three octets and the first 4 bits of the last octet define the network
- The last 4 bits of the last octet define the node
- That means we could have as many as 1111 (in binary) or 15 (in decimal) nodes on this subnetwork
- This is the basic essence of subnetting

How the Internet Works



CIDR

- Subnetting only allows a certain, limited subnets
- Another approach is *Classless InterDomain Routing* (CIDR)
- Rather than define a subnet mask, we have the IP address followed by a slash and a number
- That number can be any number between 0 and 32, which results in IP addresses like these:
 - 192.168.1.10/24 (basically a Class C IP address)
 - 192.168.1.10/31 (much like a Class C IP address with a subnet mask)
- When we use this, rather than having classes with subnets, we have *Variable-Length Subnet Masking* (VLSM) that provides classless IP address
- This is the most common way to define network IP addresses today

How the Internet Works

innovate

achieve

lead

Subnetting

- Class A (CDIR Value = /8) = Classless Inter Domain Routing = Total number of network bits
 - IP Address: 1-126
 - Default Subnet Mask: 255.0.0.0
 - 8 bits are reserved for network and the remaining 24 bits are reserved for the host

How the Internet Works

innovate

achieve

lead

Subnetting

- Class B (CDIR Value = /16) = Classless Inter Domain Routing = Total number of network bits
 - IP Address: 128-191
 - Default Subnet Mask: 255.255.0.0
 - 16 bits are reserved for network and the remaining 16 bits are reserved for the host

How the Internet Works

innovate

achieve

lead

Subnetting

- Class C (CDIR Value = /24) = Classless Inter Domain Routing = Total number of network bits
 - IP Address: 192-223
 - Default Subnet Mask: 255.255.255.0
 - 24 bits are reserved for network and the remaining 8 bits are reserved for the host

Network Bits	Host Bits
24	8
1 0 0 0 0 0 0 0	0

How the Internet Works



Subnetting – Example – Class C

- 192.168.1.0/24

255								255								255								0							
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
$2^7+2^6+2^5+2^4+2^3+2^2+2^1+2^0$								$2^7+2^6+2^5+2^4+2^3+2^2+2^1+2^0$								$2^7+2^6+2^5+2^4+2^3+2^2+2^1+2^0$								0							
8 Bits								8 Bits								8 Bits								8 Bits							
Block 1								Block 2								Block 3								Block 4							

- Default subnet mask for class C = 255.255.255.0
- CIDR Value = 24 = Total number of network bits
- We can calculate the subnet mask only from the network bits not the host bits

How the Internet Works



Subnetting – Example – Class C

- 192.168.1.0/25

255								255								255								128											
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0				
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
$2^7+2^6+2^5+2^4+2^3+2^2+2^1+2^0$								$2^7+2^6+2^5+2^4+2^3+2^2+2^1+2^0$								$2^7+2^6+2^5+2^4+2^3+2^2+2^1+2^0$								2^7											
8 Bits								8 Bits								8 Bits								8 Bits											
Block 1								Block 2								Block 3								Block 4											

- Default subnet mask for class C = 255.255.255.0
- But, CIDR Value = 25. So, we need one extra bit. We borrow that from host
- The new subnet mask = 255.255.255.128

How the Internet Works



Subnetting – Example – Class C

- 192.168.1.0/25

255								255								255								128											
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0				
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0

- Number of networks
 - 2^n (Where, n = number of bits borrowed from the host)
 - $2^1 = 2$ (We can create only two networks)
- Number of IP addresses on each network
 - 2^b (Where, b = number of remaining host bits)
 - $2^7 = 128$ (On each network we can have 128 IP addresses)
- Number of hosts on each network (IPs that can be assigned to devices)
 - $2^b - 2$ (Where, b = number of remaining host bits)
 - $2^7 - 2 = 126$ (We can assign 126 IP addresses to devices)

Note:

In every network, the first IP address is reserved for the network ID and the last IP address is reserved for broadcast ID

How the Internet Works

innovate

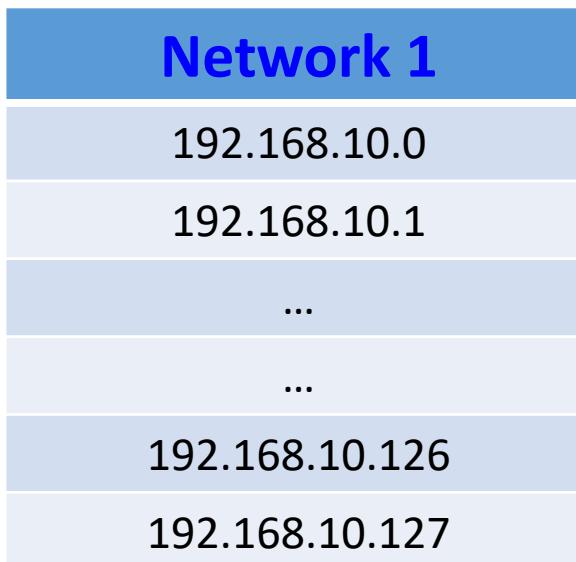
achieve

lead

Subnetting – Example – Class C

- 192.168.1.0/25

255								255								255								128											
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0				
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0



Network ID
IP Addresses that can be assigned
Broadcast ID

How the Internet Works

innovate

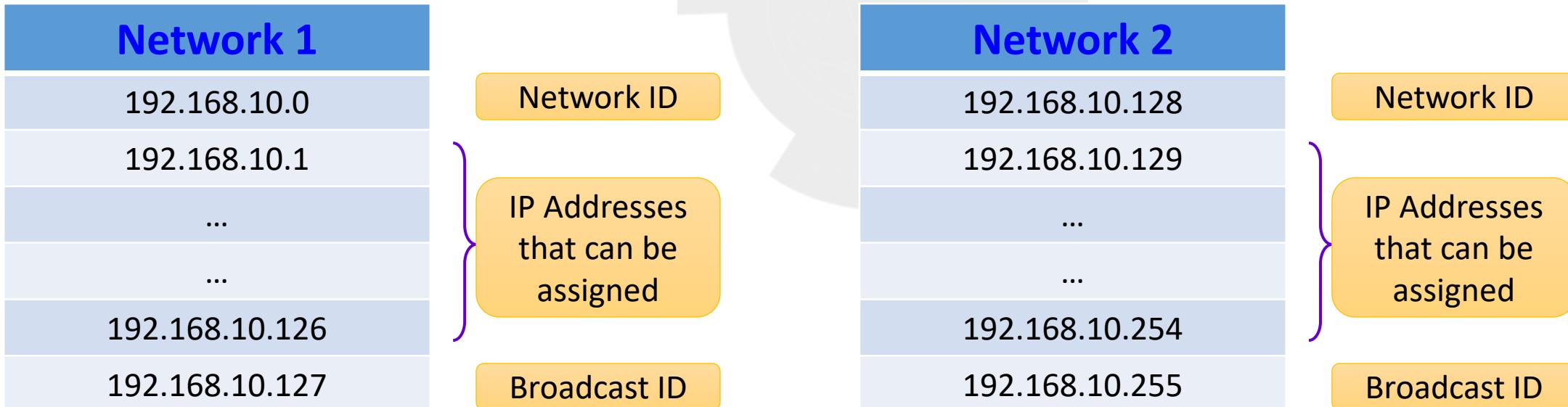
achieve

lead

Subnetting – Example – Class C

- 192.168.1.0/25

255								255								255								128											
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0				
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0



How the Internet Works



Subnetting – Example – Class C

- 192.168.1.0/26

255								255								255								192											
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0				
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0

- Number of networks
 - 2^n (Where, n = number of bits borrowed from the host)
 - $2^2 = 4$ (We can create only two networks)
- Number of IP addresses on each network
 - 2^b (Where, b = number of remaining host bits)
 - $2^6 = 64$ (On each network we can have 64 IP addresses)
- Number of hosts on each network (IPs that can be assigned to devices)
 - $2^b - 2$ (Where, b = number of remaining host bits)
 - $2^6 - 2 = 62$ (We can assign 62 IP addresses to devices)

Note:

In every network, the first IP address is reserved for the **network ID** and the last IP address is reserved for **broadcast ID**

How the Internet Works



Subnetting – Example – Class C

- 192.168.1.0/26

255								255								255								128										
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0			
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0

Network No	Network ID	Number of IPs	Broadcast ID
1	192.168.10.0	192.168.10.1 - 192.168.10.62	192.168.10.63
2	192.168.10.64	192.168.10.65 - 192.168.10.126	192.168.10.127
3	192.168.10.128	192.168.10.129 - 192.168.10.190	192.168.10.191
4	192.168.10.192	192.168.10.193 - 192.168.10.254	192.168.10.255

Subnetting

- The first value of a subnet mask must be 255
- The remaining three values can be 255, 254, 252, 248, 240, or 224
- The computer will take our network IP address and the subnet mask and use a binary AND operation to combine them

How the Internet Works

innovate

achieve

lead

IPv6

- IPv6 is an extension of IPv4
- IP version 4 is limited to 4.2 billion IP addresses
- Even with the use of private IP addresses, we will run out of available IP addresses
 - Consider all the computers, printers, routers, servers, smart phones, tablets, and so on connected to the Internet
- IP version 6 was designed to alleviate this problem
- IPv6 utilizes a 128-bit address (instead of 32), so there is no chance of running out of IP addresses in the foreseeable future
- IPv6 also utilizes a hex numbering method in order to avoid long addresses such as 132.64.34.26.64.156.143.57.1.3.7.44.122.111.201.5
- The hex address format will appear in the form of 3FFE:B00:800:2::C, for example.

IPv6

- IPv6 involves no subnetting, but it does use CIDR
- The network portion is indicated by a slash followed by the number of bits in the address that are assigned to the network portion
 - For example: /48 /64
- There is a loopback address for IPv6, and it can be written as ::/ 128
- Loopback address
 - An address that sends outgoing signals back to the same computer for testing
 - In a TCP/IP network, the loopback IP address is 127.0.0.1, and pinging this address will always return a reply unless the firewall prevents it
 - The loopback address allows a network administrator to treat the local machine as if it were a remote machine
 - The standard domain name for the address is localhost

IPv4 Vs. IPv6

- Link/machine-local address:
 - This is the IPv6 version of IPv4's APIPA (Automatic Private IP Addressing) address
 - If a machine is configured for dynamically assigned addresses and cannot communicate with a DHCP server, it assigns itself a generic IP address
 - DHCP, or Dynamic Host Configuration Protocol, is used to dynamically assign IP addresses within a network
 - IPv6 link/machine-local IP addresses all start with fe80::
 - So if your computer has this address, that means it could not get to a DHCP server and therefore made up of its own generic IP address.

How the Internet Works



IPv4 Vs. IPv6

- Site/ network-local address:
 - This is the IPv6 version of the IPv4 private address
 - Site/ network-local addresses are real IP addresses, but they only work on the local network and are not routable on the Internet
 - All site/ network-local IP addresses begin with FE and have C to F for the third hexadecimal digit: FEC, FED, FEE, or FEF
- The managed address configuration flag (M flag):
 - When the M flag is set to 1, the device should use DHCPv6 to obtain a stateful IPv6 address
- Other stateful configuration flag (O flag):
 - When the O flag is set to 1, the device should use DHCPv6 to obtain other TCP/ IP configuration settings
 - In other words, it should use the DHCP server to set things like the IP address of the gateway and DNS servers
- M flag:
 - This indicates that the machine should use DHCPv6 to retrieve an IP address.

Uniform Resource Locator (URL)

- When we visit websites, we type names rather than IP addresses in the browser's address bar
 - For example, www.yahoo.com
- This name (called a URL) needs to be translated into an IP address
- The DNS protocol handles this translation process
- If the address is found, the browser sends a packet (using HTTP) to port 80
- If that target computer has software that listens and responds to such requests, then the target computer will respond to our browser's request, and communication will be established
 - The software is web server software such as Apache or Microsoft Internet Information Server

Uniform Resource Locator (URL)

- Error Messages

- There are a series of error messages that the web server can send back to our web browser to indicate different situations
- The browser handles many of these errors itself; we never see the error message
- Error 400 series
 - All error messages in the 400 series are client errors
 - That is something is wrong on our side, not with the web server
 - E.g., Error 404
 - Refers to File Not Found
 - Indicates that our browser received back a packet (from the web server) with error code 404, denoting that requested page could not be found

Uniform Resource Locator (URL)

- Error Messages
 - Error 500 series
 - These are server errors, meaning, there is a problem on the web server
 - Error 100 series
 - These are simply informational
 - Error 200 series
 - These indicate success
 - We usually do not see these, the browser simply processes them
 - Error 300 series
 - These are re-directional, meaning the page you are seeking has moved, and your browser is then directed to the new location

Uniform Resource Locator (URL)

- Emails
 - Using email works the same way as visiting websites
 - Our email client will seek out the address of your email server
 - Then our email client will use either Post Office Protocol version 3 (POP3) to retrieve the incoming email or Simple Mail Transfer Protocol (SMTP) to send the outgoing email
 - The email server (probably at our ISP or our company) will then try to resolve the address we are sending to
 - If we send something to joe@yahoo.com, the email server will translate that email address into an IP address for the email server at yahoo.com
 - Then our server will send our email there
 - There is another protocol called Internet Message Access Protocol (IMAP) for retrieving emails from remote server, but POP3 is still the most commonly used

Uniform Resource Locator (URL)

- Chat Rooms
 - A chat room (like the other communication methods), works with packets
 - We first find the address of a chat room, and then connect
 - The difference here is that our computer's chat software is constantly sending packets back and forth
 - Whereas email only sends and receives when we tell it to
 - or on a predetermined time interval
 - The packet header section contains our IP address and the destination IP address (as well as other information)

How the Internet Works

innovate

achieve

lead

What is a Packet?

- Network traffic is really a lot of 1s and 0s that are transmitted as
 - voltages (over *unshielded twisted-pair* (UTP))
 - light wave (over optic cable) or
 - radio frequencies (over Wi-Fi)
- The data is divided into small chunks called packets
- A packet is divided into three sections:
 - The header, the data, and the footer
- The header contains information about how to address the packet, what kind of packet it is, and related data
- The data portion is the information we want to send
- The footer serves both to show where the packet ends and to provide error detection

What is a Packet?

- Header
 - There are usually at least three headers
 - Ethernet header, TCP header, and IP header
 - Each contains different information, in combination they have several pieces of information that will be interesting for forensic investigations
- TCP header
 - Contains information related to the transport layer of the OSI model
 - Contains the source and destination port for communications
 - It also has the packet number, such as packet 10 of 21

What is a Packet?

- IP header
 - Contains the source IP address, the destination IP address, and the protocol
 - The IP header also has a version number (4.0 or 6.0) for the IP packet
 - The size variable describes how large the data segment is
- Ethernet header
 - Contains information regarding the source MAC address and destination MAC address
 - When a packet gets to the last network segment in its journey, MAC address is used to find the NIC that the packet is being sent to

Basic Communications

- The packet headers also contain some signal bits
- These are single bit flags that are turned on to indicate some type of communication
- A normal network conversation starts with one side sending a packet with the SYN (synchronize) bit turned on
- The target responds with both SYN and ACK (acknowledge) bits turned on
- Then the sender responds with just the ACK bit turned on, and communication commences
- To end the communication, the original sender terminates the communication by sending a packet with the FIN (finish) bit turned on

Reference

- Easttom, Chuck. Computer Security Fundamentals (Pearson IT Cybersecurity Curriculum (ITCC)) – 4th Edition



Thank You!



BITS Pilani
Pilani | Dubai | Goa | Hyderabad

Cyber Security

Introduction to Networks and the Internet

Dr. Ramakrishna Dantu

Associate Professor, BITS Pilani

Disclaimer and Acknowledgement



- The content for these slides has been obtained from books and various other source on the Internet
- I here by acknowledge all the contributors for their material and inputs.
- I have provided source information wherever necessary
- I have added and modified the content to suit the requirements of the course



The OSI Model

The OSI Model



Overview

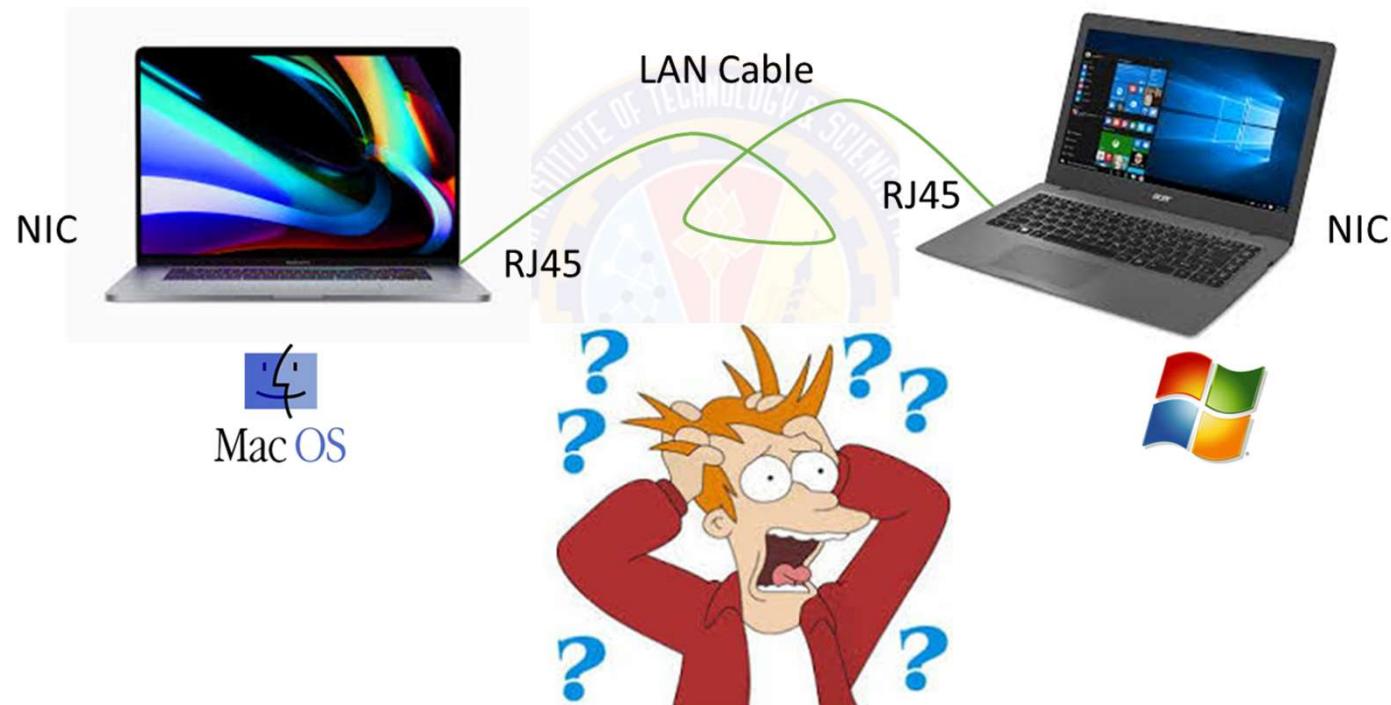
- Open Systems Interconnection (OSI) model describes how computers communicate with each other on a network
- It outlines the various protocols and activities, and tells how the protocols and activities relate to each other



The OSI Model



Overview

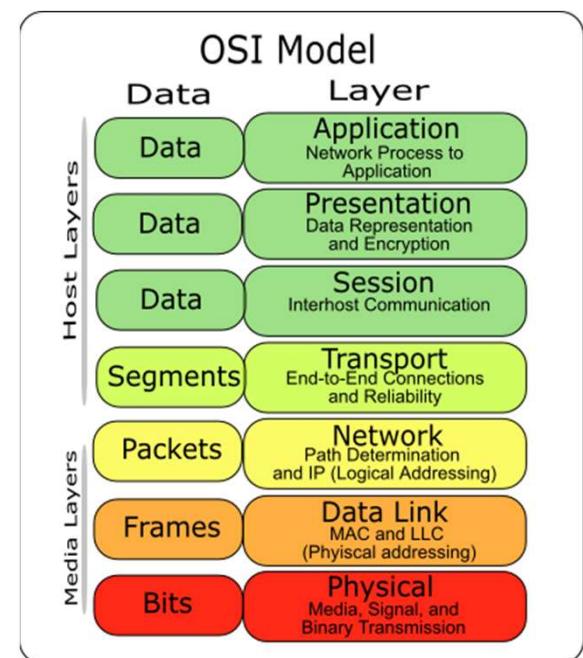
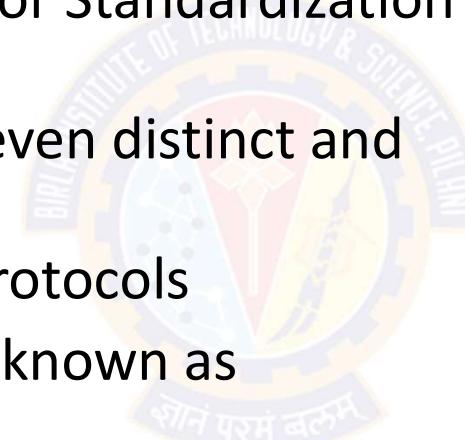


The OSI Model



Overview

- This model was originally developed by the International Organization for Standardization in 1984
- The model is divided into seven distinct and separate layers
- Each layer is a package of protocols
- Each layer possesses a trait known as 'successive dependence.'
 - This means that the successively higher layers in the model depend on the services and characteristics of the preceding lower layers



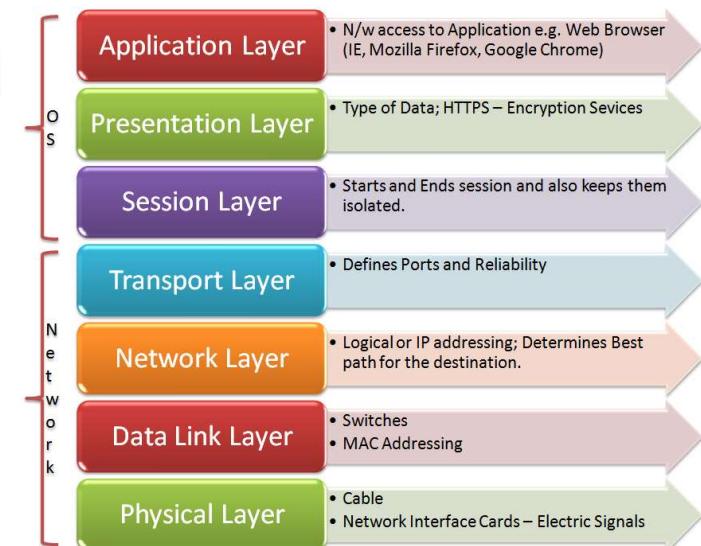
Open Systems Interconnection (OSI) Model

The OSI Model



Layer 7: Application Layer

- This doesn't mean applications such as chrome, email client, word processor, etc.,.
- The application layer is the end user's access to the network
- This layer includes protocols to make these applications work correctly
- These are applications that rely on the Internet to work
- For Example:
 - Chrome, Skype, Outlook, etc.,.

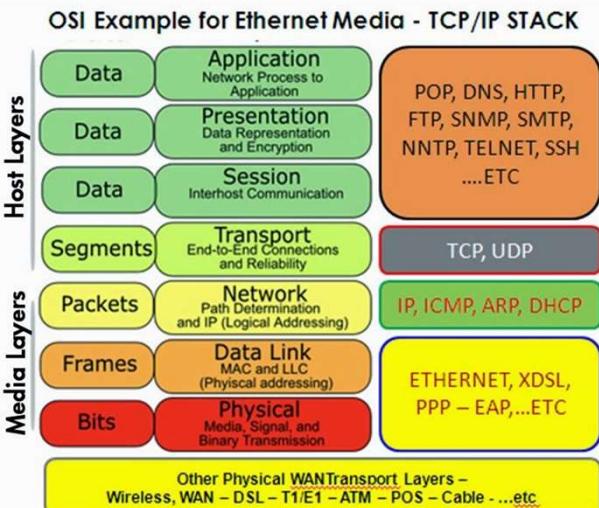
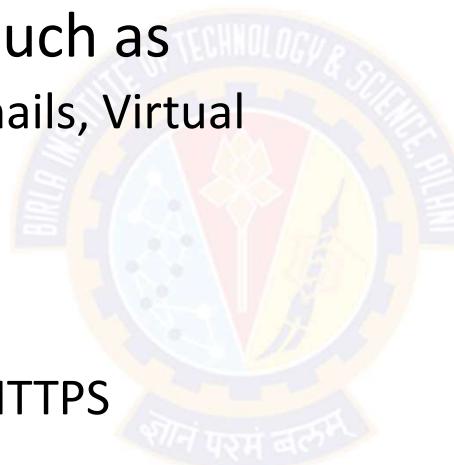


The OSI Model



Layer 7: Application Layer

- These protocols form the basis for various network services such as
 - File transfer, Web surfing, Emails, Virtual terminals, etc.,,
- For example:
 - File transfer relies on FTP
 - Web surfing relies on HTTP/HTTPS
 - Emails use SMTP
 - Virtual terminals use Telnet

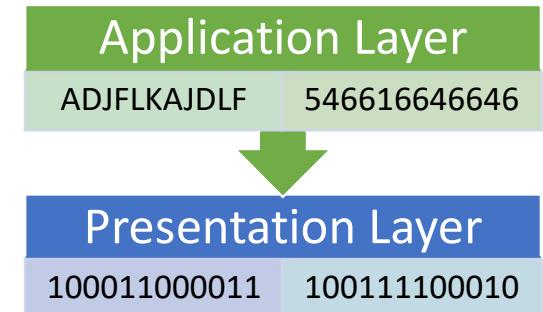


The OSI Model



Layer 6: Presentation Layer

- This layer receives data from the Application Layer
- This data is in the form of characters and numbers
- The presentation layer formats the data converts this data into machine understandable binary format (0's and 1's)
 - E.g., conversion of ASCII to EBCDIC
 - Extended Binary Coded Decimal Interchange Code
 - This process is called **translation**
- Before the data is transmitted, the presentation layer reduces the number of bits used to represent the original data
 - $100011000011 \rightarrow 10010011$
 - This reduction of data is called data **compression**
- Data compression can be lossy or lossless

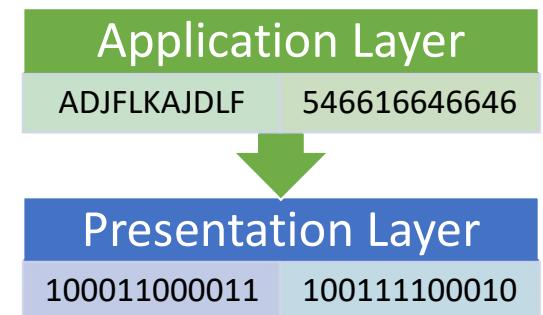


The OSI Model



Layer 6: Presentation Layer

- Data compression reduces the amount of space required to store the original file
 - E.g., 5MB → 3MB
- As the file size is reduced, data transmission can happen faster
- Data compression is useful in real-time audio and video streaming
- Data is **encrypted** before transmission to maintain the integrity/security of the data
- At the receiver side, data is **decrypted**, before presenting
- Secure Socket Layer (SSL) protocol is used in the presentation layer for encryption and decryption
- Essentially, presentation layer performs three functions:
 - Translation, Compression, & Encryption/Decryption



The OSI Model



Layer 5: Session Layer

- Suppose we decide to have a party at our home



- We hire an event management team to help us organize the party
- Helpers will help us with setting up, assisting, cleaning, and closing the party



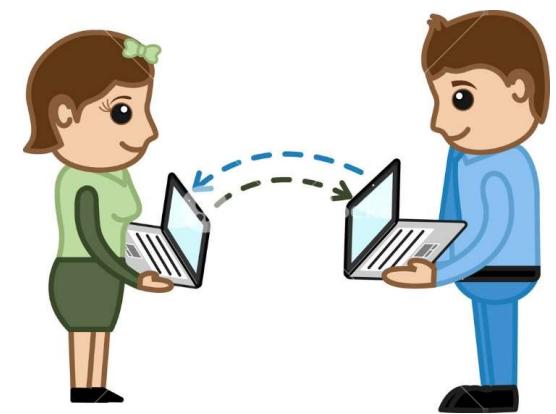
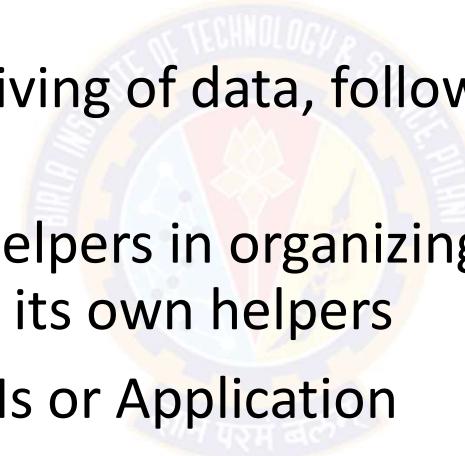
- The Session Layer performs a similar function

The OSI Model



Layer 5: Session Layer

- The session layer is responsible for setting up and managing all connections or sessions
- It enables sending and receiving of data, followed by the termination of connections or sessions
- Similar to the way we had helpers in organizing the party, session layer also has its own helpers
- These helpers are called APIs or Application Programming Interfaces
 - E.g., NETBIOS – Network Basic Input/Output System allows applications on different computers to communicate with each other

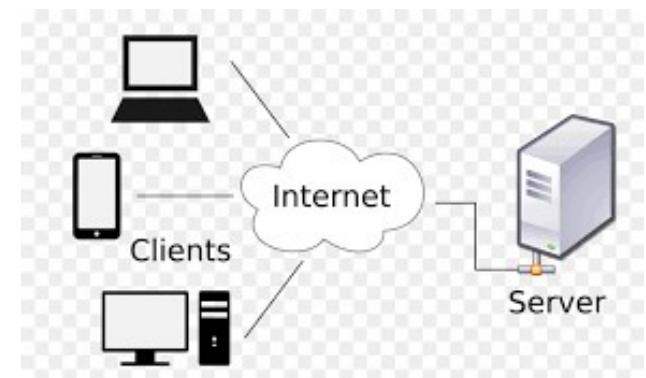


The OSI Model



Layer 5: Session Layer

- Session Layer performs two functions – **Authentication & Authorization**
- Before establishing a session or connection, the server performs a function called **authentication**
- **Authentication** process verifies the identity of the client where the user name and password are matched
- Once authenticated, a session or connection is established between the computer and the server
- After authenticating the user, **authorization** is checked
- **Authorization** is the process where the server checks if the user has permission to access a file or any resource
 - If not, users gets a message saying "Access Denied"

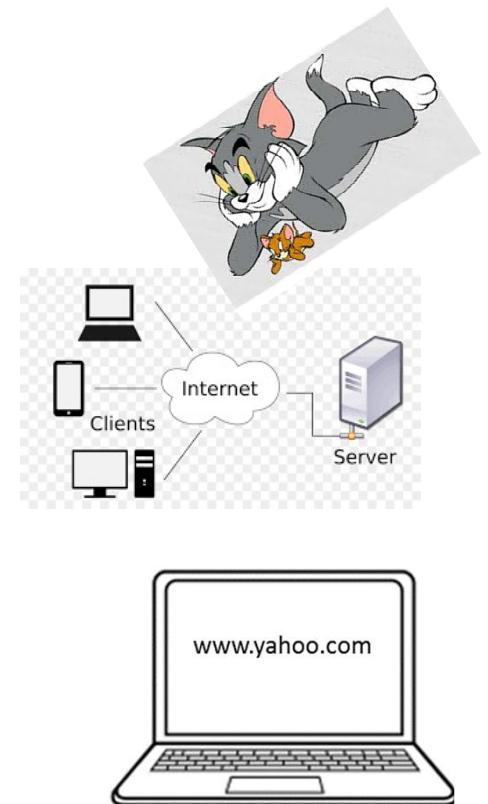
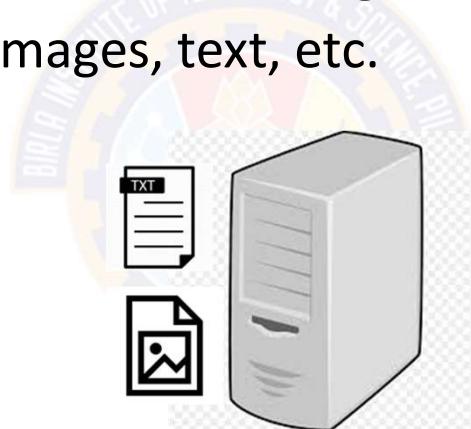


The OSI Model



Layer 5: Session Layer

- Thus session layer also performs **session management**
- Session layer keeps track of files that are being downloaded
- For e.g., a web page contains images, text, etc.



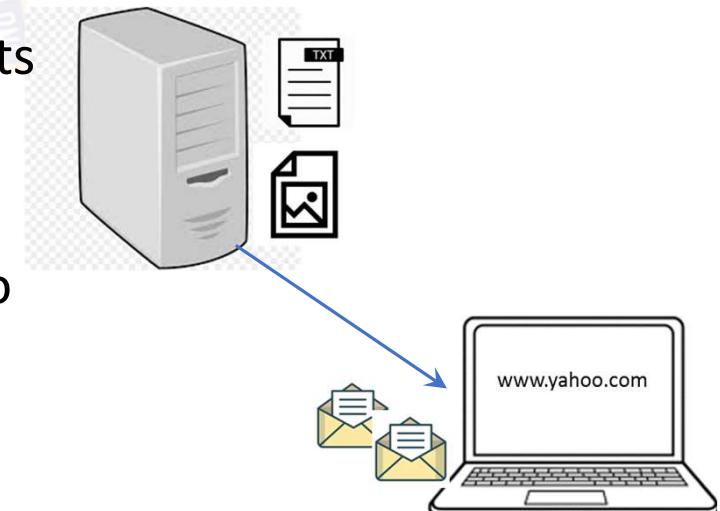
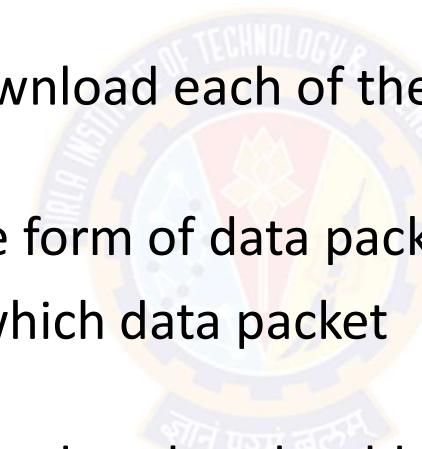
- These images and text are stored as separate files on the web server

The OSI Model



Layer 5: Session Layer

- When we request a web page, web browser opens a separate connection or session with the server
- This session enables us to download each of these text and image files separately
- These files are received in the form of data packets
- Session layer keeps track of which data packet belongs to which file
- It also tracks where the received packet should go (the destination)
 - in this case, it goes to web browser
- Thus session layer helps in **session management**



The OSI Model



Layer 5: Session Layer

- Thus, the session layer performs three key functions
 - Authentication
 - Authorization
 - Session Management
- Our web browser performs all these functions of:
 - Session layer
 - Presentation layer, and
 - Application layer

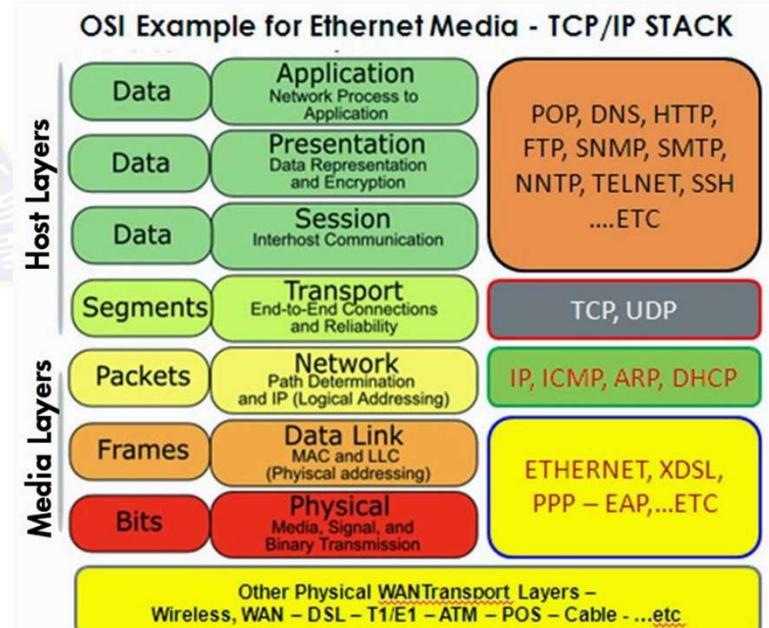
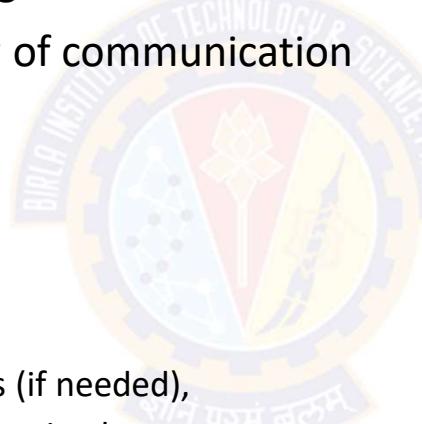


The OSI Model



Layer 4: Transport Layer

- The transport layer deals with end-to-end issues, such as procedures for entering and departing from the network
- Transport layer controls the reliability of communication through
 - Segmentation
 - Flow Control
 - Error Control
- It is responsible for:
 - breaking a large data into smaller packets (if needed),
 - ensuring that all the packets have been received,
 - eliminating duplicate packets, and
 - performing flow control to ensure that no computer is overwhelmed by the number of messages it receives



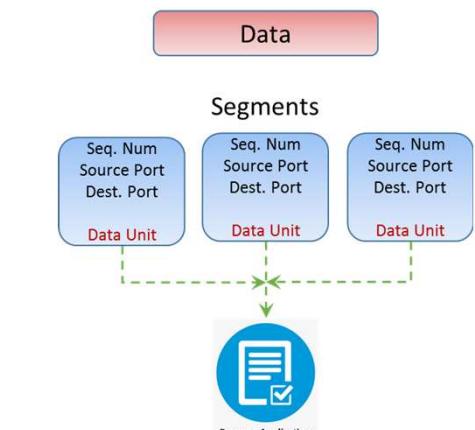
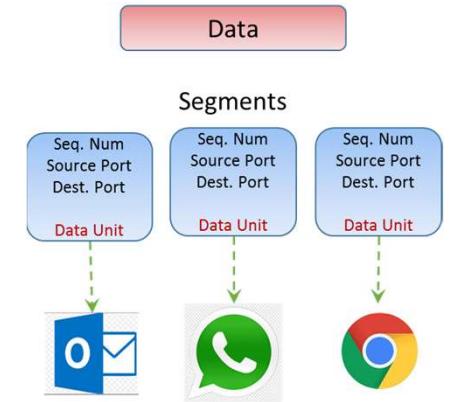
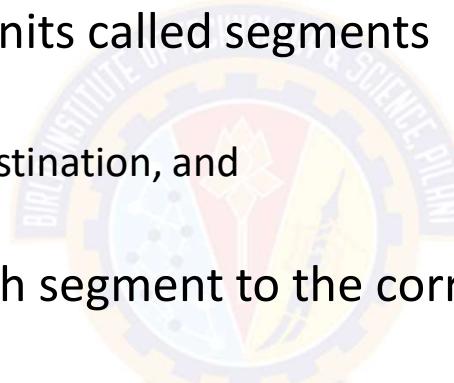
The OSI Model



Layer 4: Transport Layer

- **Segmentation**

- Data is divided into smaller units called segments
- Each segment contains:
 - port number of source and destination, and
 - sequence number
- Port number helps direct each segment to the correct application
- Sequence number helps to reassemble the segments in correct order to form correct message at the receiver
 - so that it can be delivered to the correct application in that order



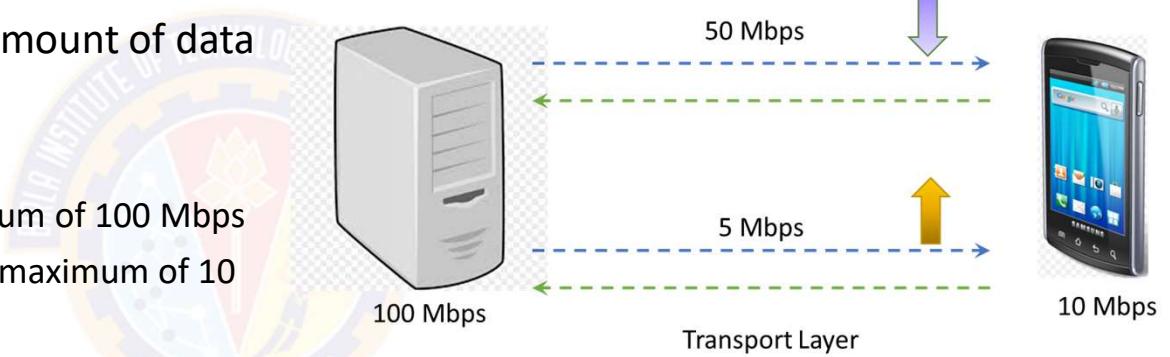
The OSI Model



Layer 4: Transport Layer

- Flow Control

- Here, transport layer controls the amount of data being transmitted
- Consider that the
 - server can transmit data at a maximum of 100 Mbps
 - mobile phone can process data at a maximum of 10 Mbps
- Server sends data at 50 Mbps
 - This is more than the processing capacity of mobile phone
 - Mobile phone with the help of transport layer can tell the server to slow down data transmission rate to 10 Mbps so there is no data loss
- Server sends data at 5 Mbps
 - Mobile phone tells the server to increase the speed to 10 Mbps to maintain system performance

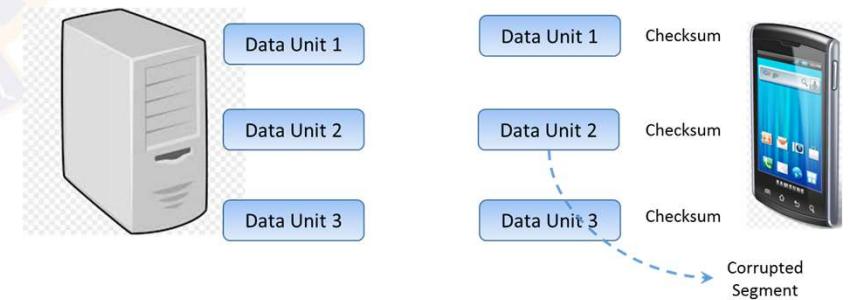
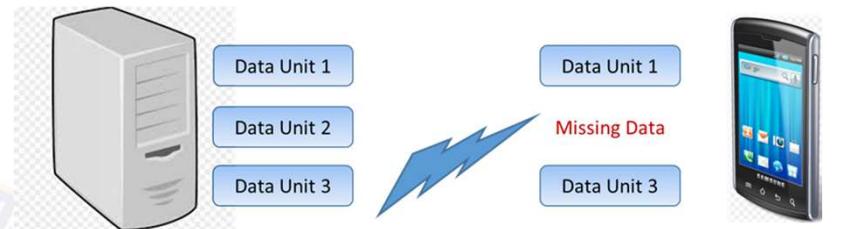


The OSI Model



Layer 4: Transport Layer

- Error control
 - Transport layer also performs error control
 - If a data packet doesn't arrive at the destination, transport layer uses **Automatic Repeat Request** scheme to retransmit the lost or corrupted data
 - A group of bits called checksum is added to each segment by the transport layer to find out received corrupted segment



The OSI Model



Layer 4: Transport Layer

- Transport layer protocols are
 - Transmission Control Protocol (TCP)
 - User Datagram Protocol (UDP)
- Transport layer performs two types of transmission services
 - Connection-oriented Transmission
 - Done using TCP
 - Connectionless Transmission
 - Done using UDP

The OSI Model



Layer 4: Transport Layer

- UDP Vs. TCP

- UDP is faster than TCP, because
 - UDP does not provide any feedback
 - TCP provides feedback so that lost data can be retransmitted
- UDP is used where it doesn't matter if we received all data
 - E.g., Online video streaming, Songs, Games, VOIP
- TCP is used where full data delivery is must
 - E.g., WWW, Email, FTP, etc.,

TCP vs UDP

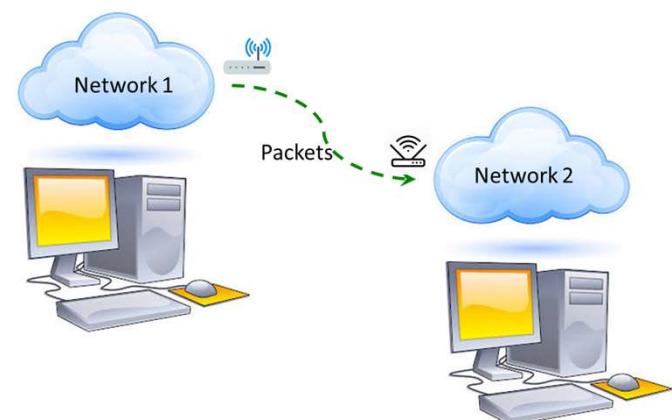
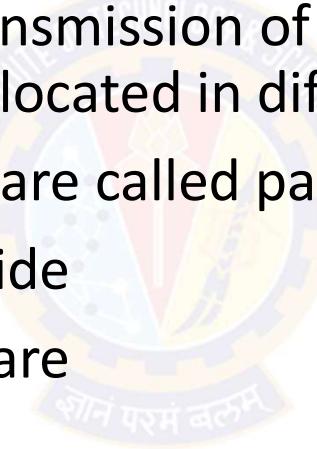
- | TCP | vs | UDP |
|--|-----------|---|
| <ul style="list-style-type: none">• Connected• State Memory• Byte Stream• Ordered Data Delivery• Reliable• Error Free• Handshake• Flow Control• Relatively Slow• Point to Point• Security: SSL/TLS | | <ul style="list-style-type: none">• Connectionless• Stateless• Packet/Datagram• No Sequence Guarantee• Lossy• Error Packets Discarded• No Handshake• No Flow Control• Relatively Fast• Supports Multicast• Security: DTLS |

The OSI Model



Layer 3: Network Layer

- Transport layer passes data segments to Network Layer
- Network layer works for the transmission of the received data segments from one computer to another located in different networks
- Data units in the network layer are called packets
- It is the layer where routers reside
- Key functions of network layer are
 - Logical addressing
 - Routing
 - Path determination

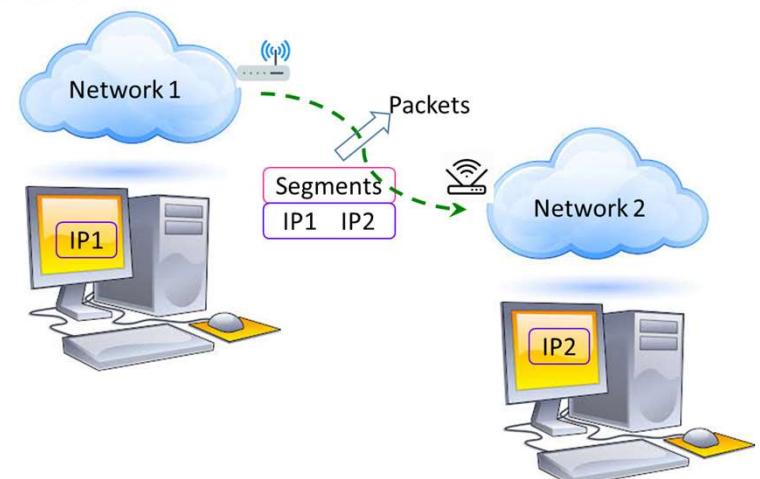


The OSI Model



Layer 3: Network Layer

- Logical addressing
 - IP addressing (IPv4 & IPv6) done in network layer is called Logical Addressing
 - Every computer in a network has a unique IP address
 - Network layer assigns sender's and receiver's IP address to each segment to form an IP packet
 - IP addresses are assigned to ensure that each data packet reaches the correct destination



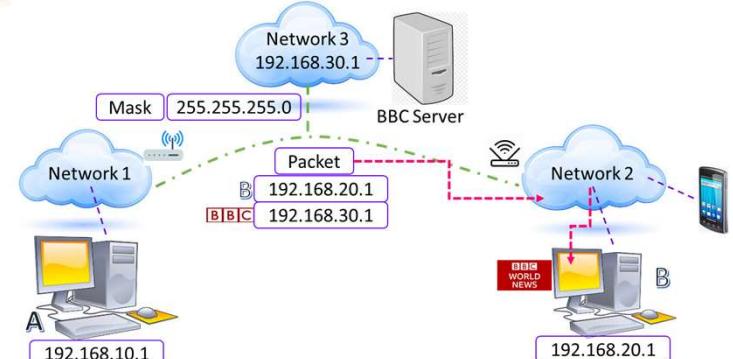
The OSI Model



Layer 3: Network Layer

- **Routing**

- Routing is a method of moving data packets from source to destination
- Based on IP address and mask, routing decisions are made in a computer network
- It is based on the logical address format of IPv4 or IPv6 and subnet mask
- Suppose computers A & B are connected to networks 1 & 2 respectively
- From computer B we requested to access BBC NEWS website
- There is a reply from BBC server to computer B in the form of a packet
- This packet must be delivered to computer B only



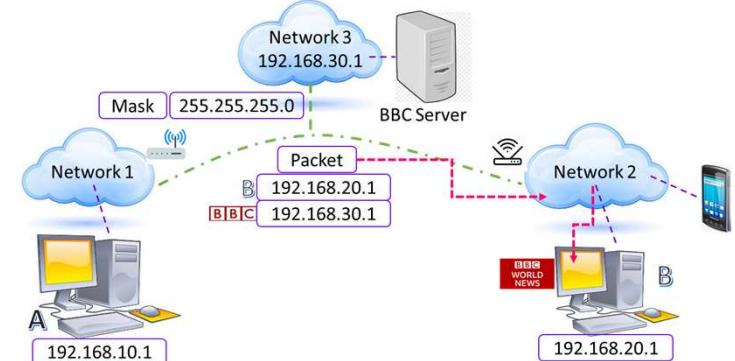
The OSI Model



Layer 3: Network Layer

- **Routing**

- As we know, both computers A & B have their unique IP addresses
- Network layer of the BBC server adds sender and receiver's IP address in the packet
- The mask 255.255.255.0 tells that the first three octets (**192.168.20.1**) of the IP address represent network, while the last octet represents host or computer B
- Based on the IP address format, the received data packet will first move to network2 and then to computer B
- Based on IP address and mask, routing decisions are made in a computer network

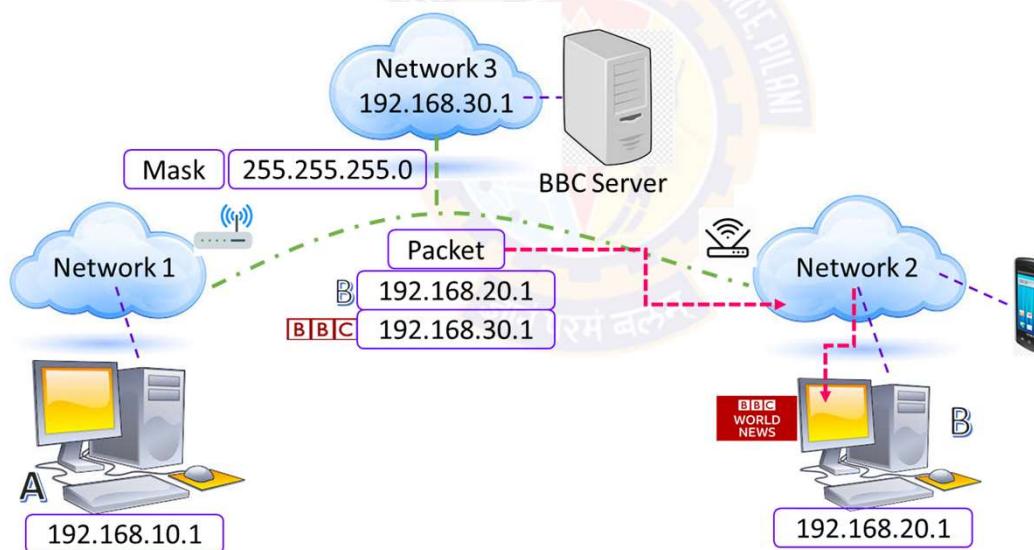


The OSI Model



Layer 3: Network Layer

- Routing
 - Based on IP address and mask, routing decisions are made in a computer network

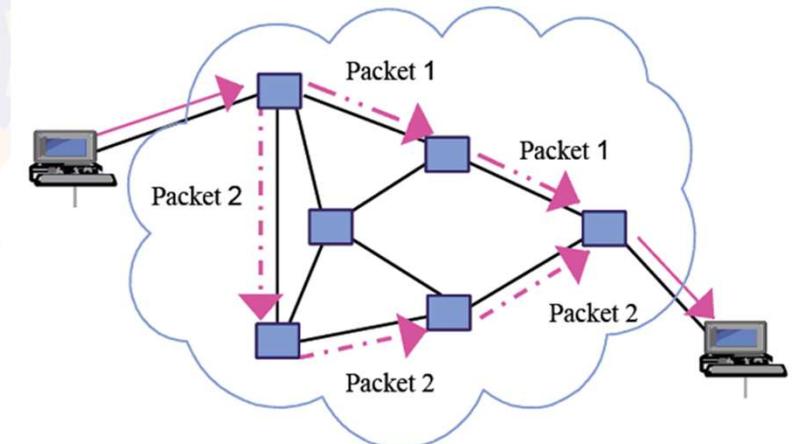


The OSI Model



Layer 3: Network Layer

- Path Determination
 - A computer can be connected to an Internet server in a number of ways
 - Choosing the best possible path for data delivery from source to destination is called path determination
 - Layer 3 devices use protocols such as:
 - OSPF - Open Shortest Path First
 - BGP – Border Gateway Protocol
 - IS-IS - Intermediate System to Intermediate SystemTo determine the best possible path for data delivery

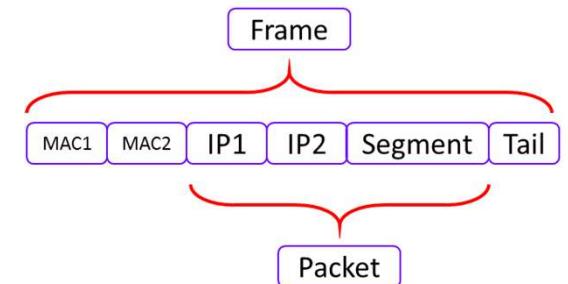


The OSI Model



Layer 2: Data Link Layer

- Data Link Layer receives data packets from the Network Layer
- Data unit in the data link layer is called a **frame**
- Data packets contain IP addresses of the sender and the receiver
- There are two kinds of addressing
 - Logical addressing
 - Done in the network layer where sender's and receiver's IP address are assigned to each segment to form a data packet
 - Physical addressing
 - Done in the data link layer, where MAC address of sender and receiver are assigned to each data packet to form a frame
 - MAC address is a 12 digit alpha-numeric number embedded in the NIC of a computer

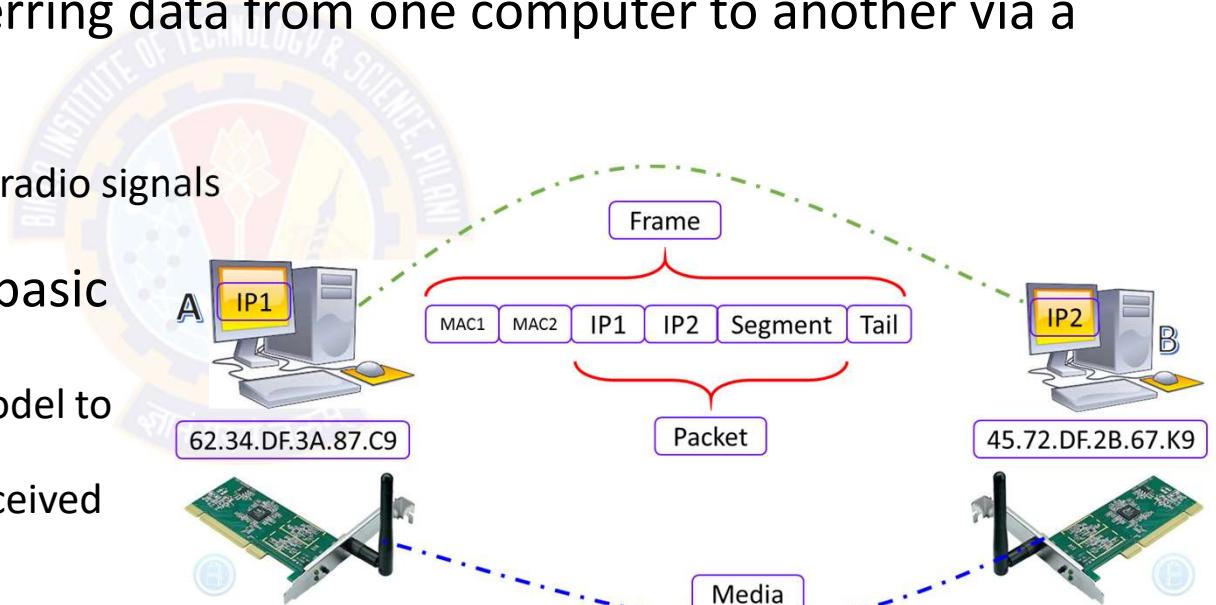


The OSI Model



Layer 2: Data Link Layer

- Data Link Layer is embedded as software in the NIC of the computer
- It provides a means for transferring data from one computer to another via a local media
- Local media includes:
 - copper wire, fiber optics, or air for radio signals
- Data Link Layer performs two basic functions:
 - it allows the upper layers of OSI model to access media
 - controls how data is placed and received from the media using such as
 - Media Access Control (MAC)
 - Error Detection

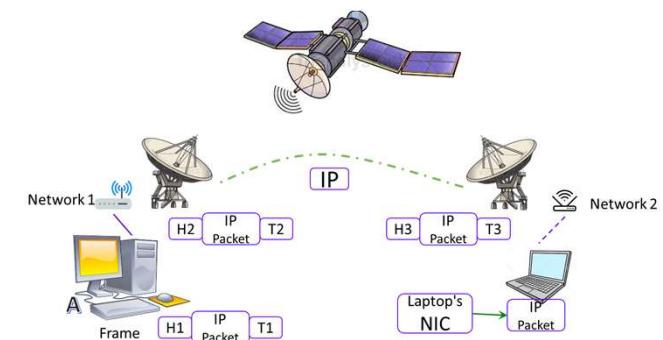


The OSI Model



Layer 2: Data Link Layer

- Consider two distant hosts:
 - A desktop and a Laptop communicating with each other
- As laptop and desktop are connect to two different networks
 - they will be using network layer protocols (E.g., IP) to communicate with each other
- Desktop is connected to router R1 via an Ethernet cable
- Laptop is connected to router R2 via a wireless link
- Router R1 and R2 are connected via a satellite link



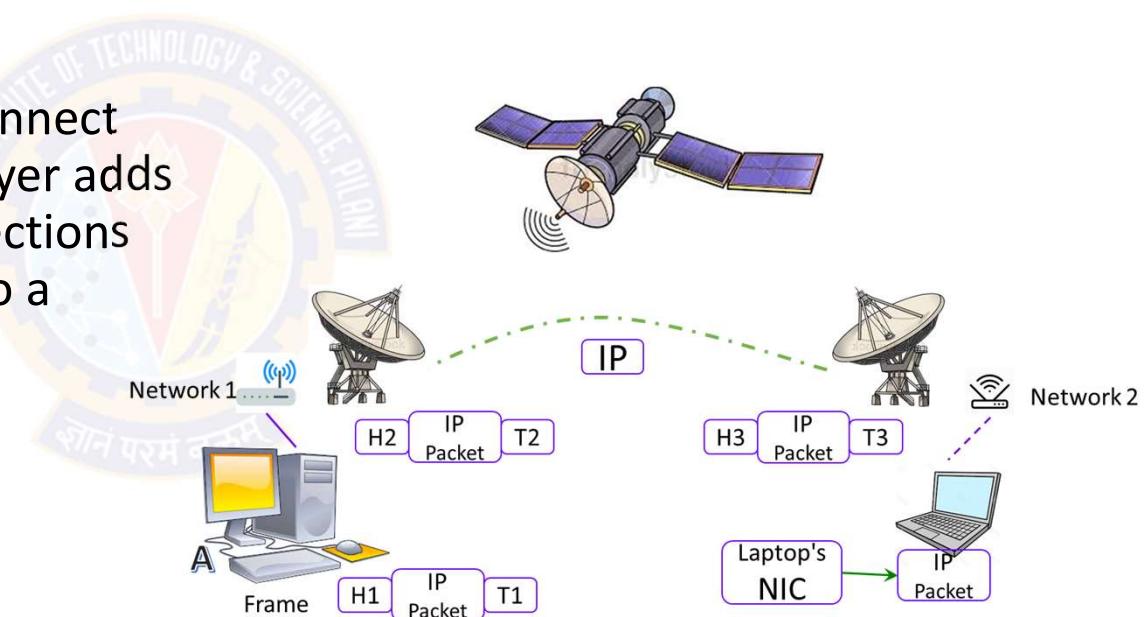
The OSI Model



Layer 2: Data Link Layer

- Desktop wants to send some data to laptop

- Based on the medium used to connect desktop to router R1, data link layer adds some data in the head and tail sections of the IP packet and converts it to a frame (E.g., Ethernet frame)

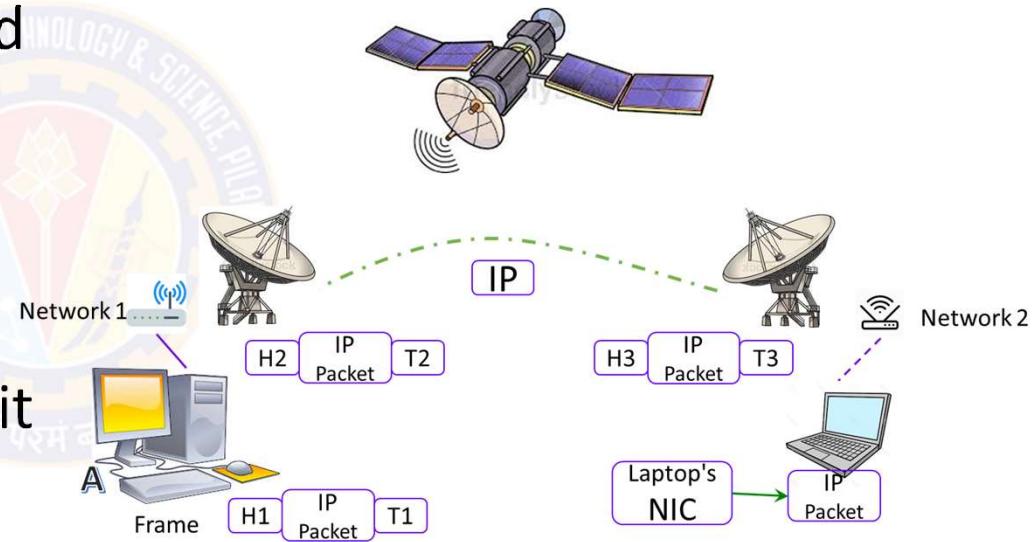


The OSI Model



Layer 2: Data Link Layer

- Router R1 receives this frame, decapsulates it to an IP Packet and then encapsulates it again to a frame so that it can cross the satellite link to reach router R2
- Router R2 again decapsulates the received frame and encapsulates it again to form a wireless data link frame

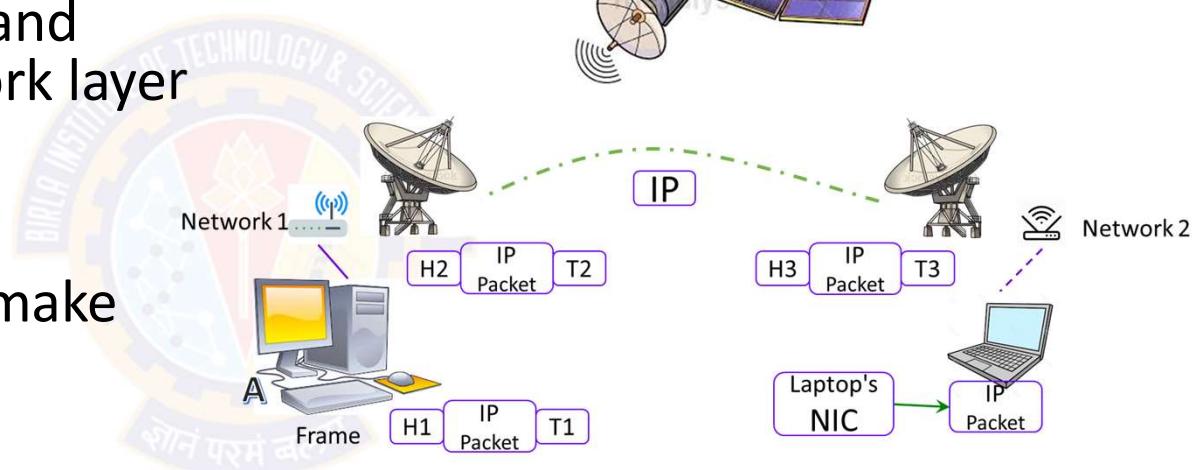


The OSI Model



Layer 2: Data Link Layer

- Laptop receives this wireless data link frame, decapsulates it, and forwards IP packet to network layer
- Finally data arrives at the application layer
- Application layer protocols make the received data visible on computer screen
- Higher level layers are able to transfer data over the media with the help of data link layer



The OSI Model



Layer 2: Data Link Layer

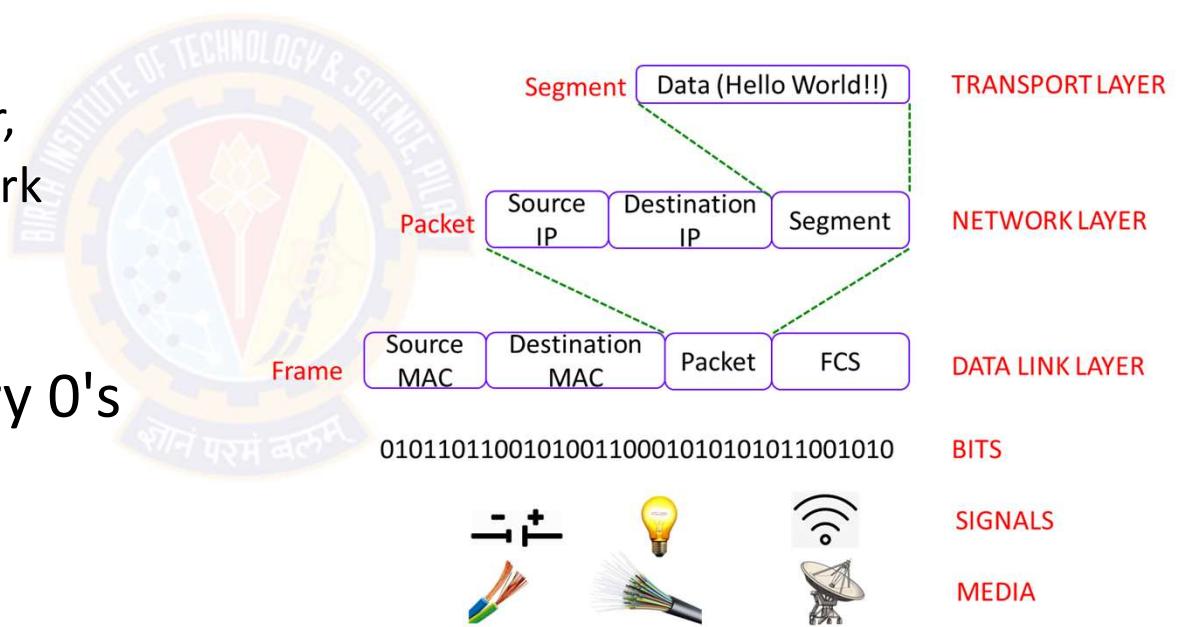
- Media Access Control
 - Data link layer also controls how the data is placed and received from the media
 - The technique used to get the frame on and off the media is called Media Access Control
 - There may be a number of devices connected to a common media
 - If two or more devices connected to same media send data simultaneously, there may be collisions of data packets resulting in loss of data
 - To avoid this situation, data link layer keeps an eye on when the shared media is free so that devices can transmit data for the receiver
 - This is called Carrier Sense Multiple Access (CSMA)
- Error Control
 - Tail of each frame contains bits which are used to check for errors in the received frame
 - Errors occur due to certain limitations of the media used for transmitting data

The OSI Model



Layer 1: Physical Layer

- Till now, data from application layer has been
 - segmented by transport layer,
 - placed into packets by network layer, and
 - framed by data link layer
- This is a sequence of binary 0's and 1's

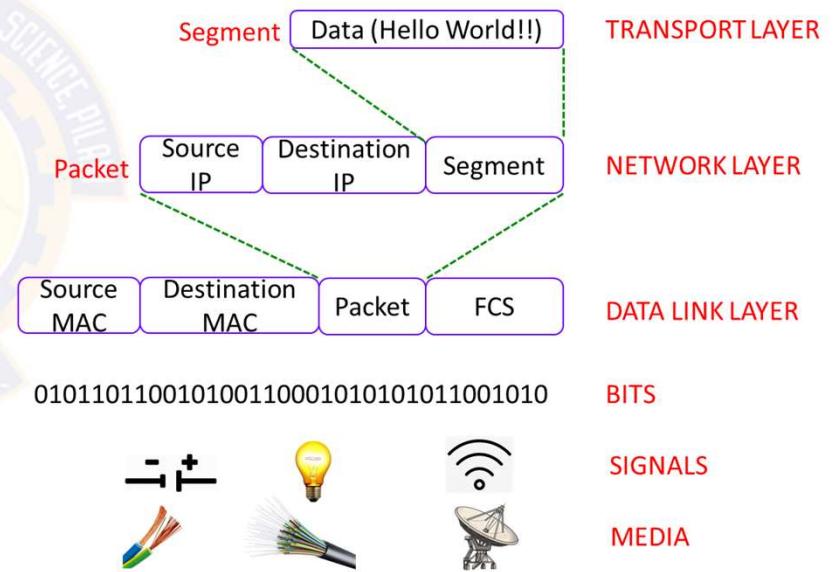
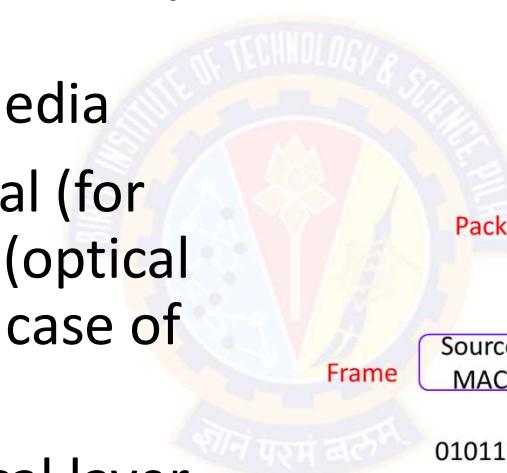


The OSI Model



Layer 1: Physical Layer

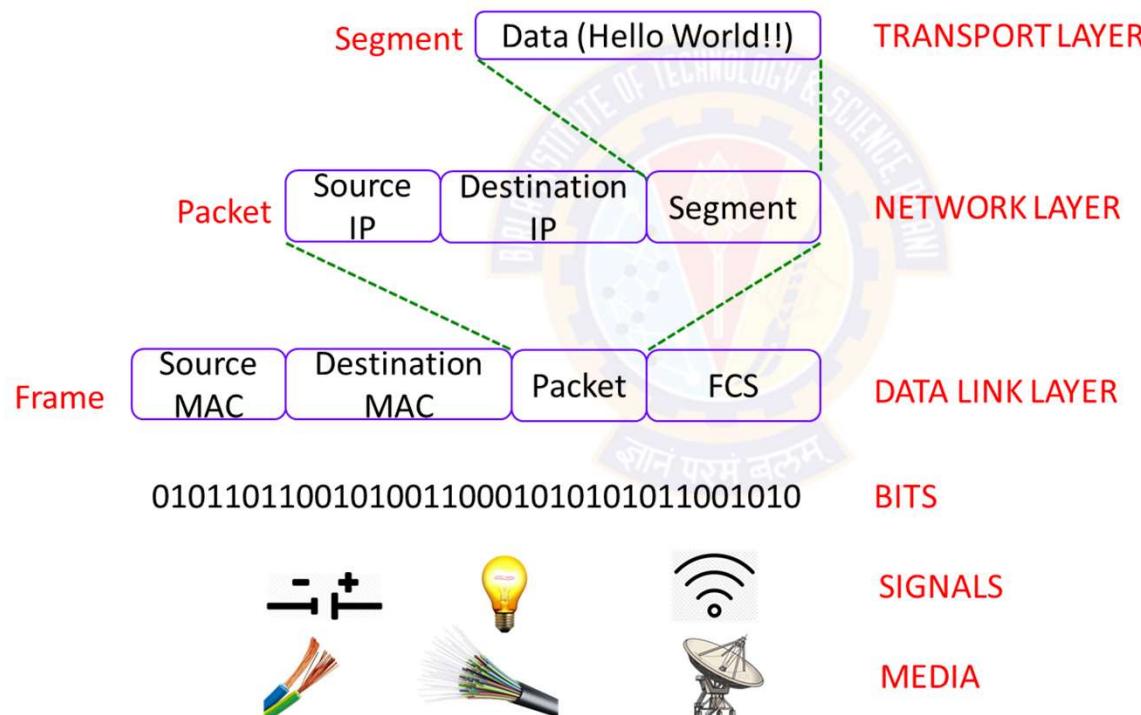
- Physical layer converts this binary sequence into signals and transmits over the local media
- It can be an electrical signal (for copper cable), light signal (optical fiber), and radio signal (in case of air)
- Signal generated by physical layer depends on the type of media used to connect two devices



The OSI Model



Layer 1: Physical Layer



The OSI Model



Seven Layers of the OSI Networking Model

Layer	Description	Protocols
Application Layer	This layer controls and mediates the interaction of the network with the Operating System and the applications installed on this OS It basically defines how the applications handle the communications in which the system becomes involved when connected to a network	POP, SMTP, DNS, FTP, and so on
Presentation Layer	Performs data compression/ decompression and encryption/decryption	
Session Layer	Defines the connection between two computers as either a client-server connection or a peer-to-peer connection The term 'session' is used to describe this virtual network connection between computers	NetBIOS
Transport Layer	Mediates the movement of data between all the other layers	TCP, UDP



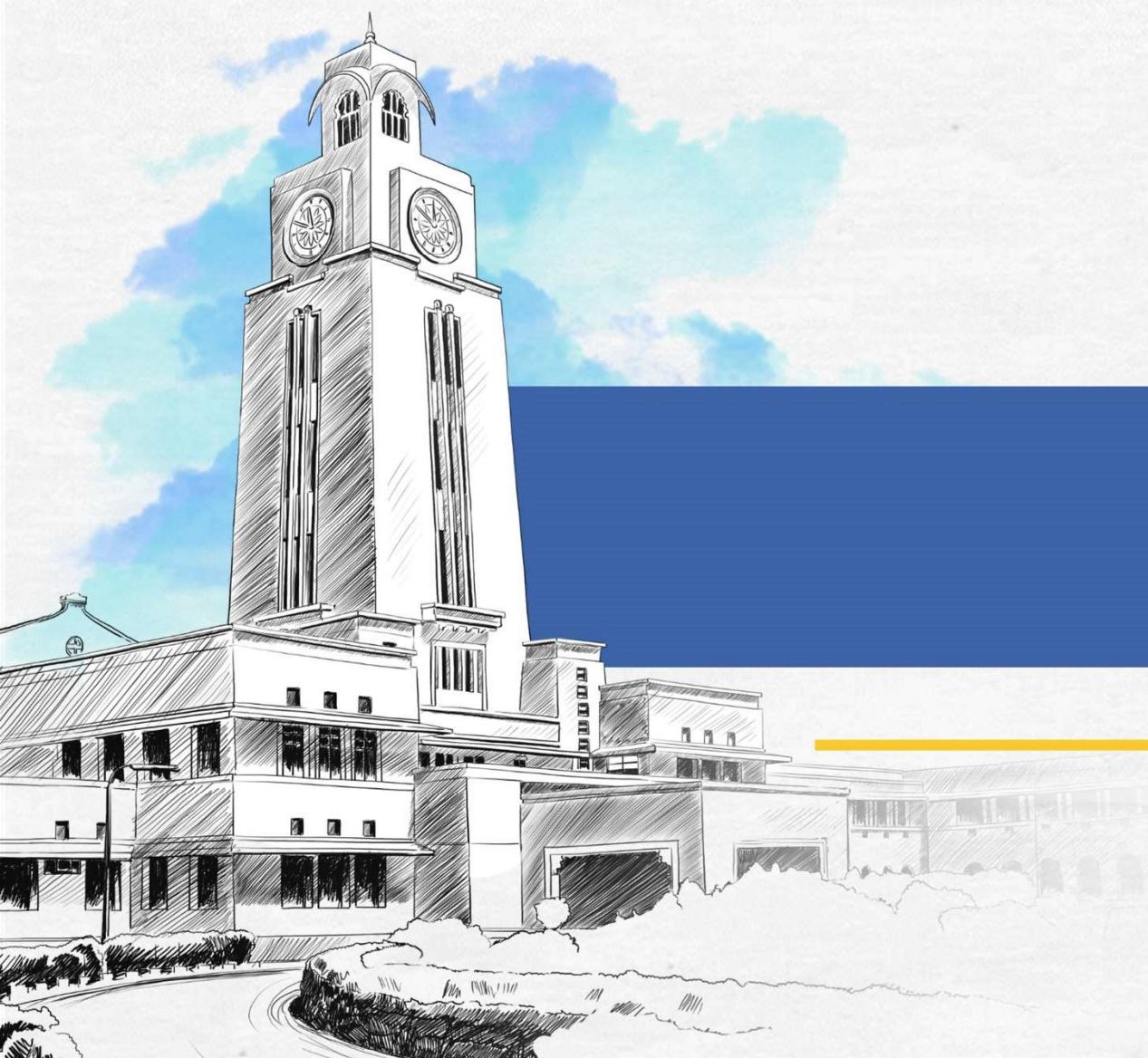
The OSI Model

Seven Layers of the OSI Networking Model

Layer	Description	Protocols
Network Layer	<p>Defines the route through which the data packets will travel from node to node</p> <p>For this purpose, the transport layer masks the characteristics of lower layers from the upper layers in the OSI model</p>	IP, Internet Control Message Protocol
Data Link Layer	Bridges the connection between the third layer (network layer) and the first layer (physical layer) by defining and implementing a protocol through which the network layer transmits its data to the physical layer	Address Resolution Protocol, Serial Line Internet Protocol, Point-to-Point Protocol
Physical Layer	Specifies the network cable, the router, the DSU/CSU box, and the other physical mediums involved.	None



Thank You!



BITS Pilani
Pilani | Dubai | Goa | Hyderabad

Cyber Security

Cyber Threat Landscape and Common Cyber Attacks

Dr. Ramakrishna Dantu

Associate Professor, BITS Pilani

Disclaimer and Acknowledgement



- The content for these slides has been obtained from books and various other source on the Internet
- I here by acknowledge all the contributors for their material and inputs.
- I have provided source information wherever necessary
- I have added and modified the content to suit the requirements of the course

Cyber Threat Landscape and Common Cyber Attacks



Agenda

- The Threat Landscape
- Understanding Vulnerabilities
- Common Cyber Attacks
 - Stages and Patterns
 - Targeted and Non-targeted Attacks
 - Reducing exposure to Cyber Attacks
- Essential Cyber Security Controls
 - Boundary firewalls and Internet gateways
 - Secure configuration
 - Whitelisting and execution control
 - User access control
 - Password policy
 - Content checking





The Threat Landscape

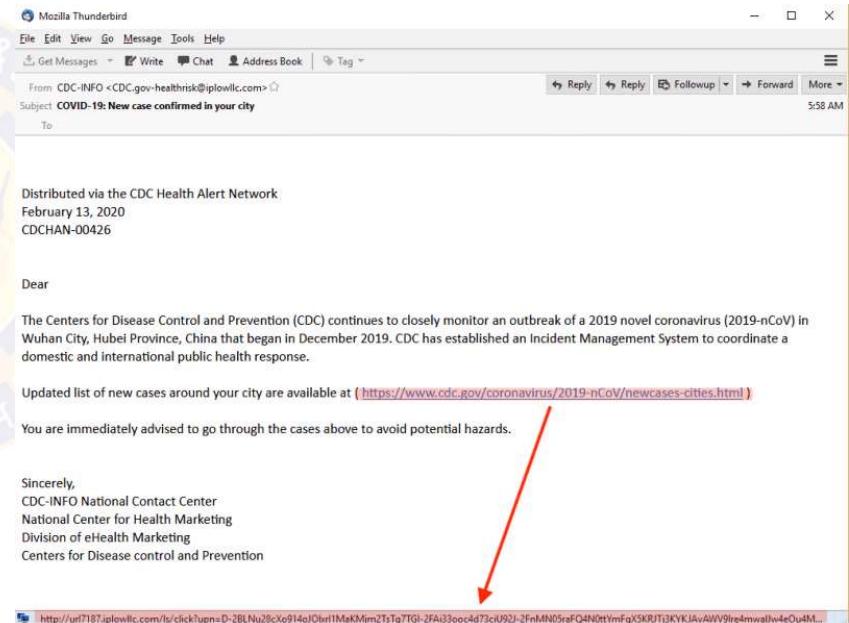
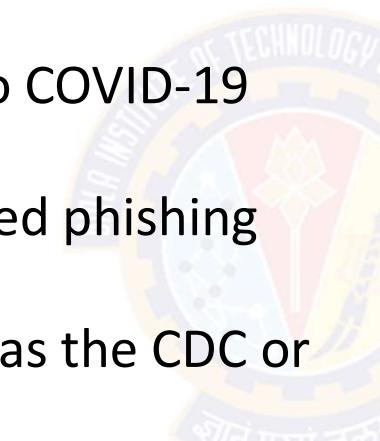
१०८ परमं बलूः

The Threat Landscape



Scenario

- Before we take a look at the cyber security threat landscape, let's look at this scenario
- Cybercrime Up 600% Due To COVID-19 Pandemic
- There is a rise in sophisticated phishing emails due to COVID-19
- Malicious actors are posing as the CDC or WHO representatives
- These emails are designed to deceive and trick recipients into taking an action:
 - clicking a malicious link, or opening an attachment with a virus



CDC = Center for Disease Control and Prevention
WHO = World Health Organization



The Threat Landscape

Key Industry Trends

- The cyber threat landscape is complex and constantly changing
- Cybersecurity has never been more important than before
- COVID-19 has forced companies to create remote workforces and operate off cloud-based platforms
- The rollout of 5G has made connected devices more connected than ever
- Some industry trends to watch for in 2021 and beyond
 - Remote workers will continue to be a target for cybercriminals
 - As a side effect of remote workforces, cloud breaches will increase
 - The cybersecurity skills gap will remain an issue
 - As a result of 5G increasing the bandwidth of connected devices, IoT devices will become more vulnerable to cyber attacks



The Threat Landscape

Some Facts

Fact	Source
95% of cybersecurity breaches are caused by human error	Cybint
The worldwide information security market is forecast to reach \$170.4 billion in 2022	Gartner
88% of organizations worldwide experienced spear phishing attempts in 2019	Proofpoint
68% of business leaders feel their cybersecurity risks are increasing	Accenture
On average, only 5% of companies' folders are properly protected	Varonis
Data breaches exposed 36 billion records in the first half of 2020	RiskBased
86% of breaches were financially motivated and 10% were motivated by espionage	Verizon
45% of breaches featured hacking, 17% involved malware and 22% involved phishing	Verizon
Between January 1, 2005, and May 31, 2020, there have been 11,762 recorded breaches	ID Theft Resource Center
The top malicious email attachment types are .doc and .dot which make up 37%, the next highest is .exe at 19.5%	Symantec
An estimated 300 billion passwords are used by humans and machines worldwide	Cybersecurity Media

The Threat Landscape



Some Facts

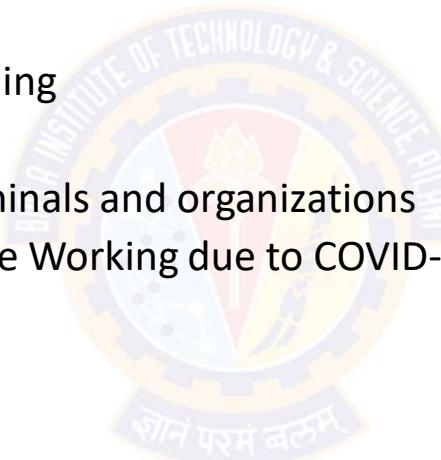
Fact
There is a hacker attack every 39 seconds
43% of cyber attacks target small business
The global average cost of a data breach is \$3.9 million across SMBs
9.7 Million Records healthcare records were compromised in September 2020 alone
Approximately \$6 trillion is expected to be spent globally on cybersecurity by 2021
Connected IoT devices will reach 75 billion by 2025
Unfilled cybersecurity jobs worldwide is already over 4 million
More than 77% of organizations do not have a Cyber Security Incident Response plan
Most companies take nearly 6 months to detect a data breach, even major ones
Share prices fall 7.27% on average after a breach
Total cost for cybercrime committed globally will reach \$6 trillion by 2021



The Threat Landscape

Some Perspectives of Security

- Let's understand some perspectives of security
 - Technology is the cause of attack
 - Risk-Reward Ratio and Ease of stealing
 - Cyber crime Vs. Physical crime
 - Information is an asset to both criminals and organizations
 - Personal Computing Assets (Remote Working due to COVID-19)
 - The Digital Divide
 - The Growing Internet of Things
 - Increasing use of Social Media





The Threat Landscape

Technology is the cause of attack

- In today's world the growth and prominence of technologies and data are showing no signs of slowing down
- The technology changes in unimaginable ways
 - We can be attacked both physically and virtually
- For today's organizations that rely heavily on technology (particularly the Internet) for doing their business
 - Virtual attacks are far more threatening
- For every vulnerability fixed, another pops up, ripe for exploitation
- When a vulnerability is identified, a tool that can exploit it is often developed and used within hours
 - This is faster than the time it normally takes for the vendor to release a patch, and certainly quicker than the time many organizations take to install that patch
- The adoption of new innovations creates an environment where threat landscapes can change quickly



The Threat Landscape

Risk-Reward Ratio & Ease of Stealing

- The technology gives attackers a huge advantage over the defenders
 - They attack anyone, anywhere, from the comfort of their home
 - They often have automated tools to identify their victims – and their vulnerabilities
- From an attacker's perspective, there is often a very good risk-to-reward ratio:
 - For the victim, it can be hard enough to detect that the attack happened at all, never mind trace who was behind it
- It is the very nature of the digital information that we are trying to protect that is easy to copy
- In fact, stealing the information does not require removing it from its original location at all
 - meaning that the owner of that information may never realize that the theft happened



The Threat Landscape

Cyber crime Vs. Physical crime

- Committing crimes over the Internet can also be very lucrative
- Physical pickpocketing compared with digitally targeting someone
 - Stealing cash and credit cards can only be beneficial for short term
 - Stealing a person's identity can get credit cards issued in the victim's name
- Upscale that to targeting businesses
 - A criminal might get access to thousands or even millions of credit card details and personal information
 - They can use the information for themselves or sell it on the dark web
 - where you can buy virtually anything, from drugs and organs to hacking software and stolen credentials
- The profits are certainly far greater compared to a physical crime conducted in the same timescale and with the same manpower



The Threat Landscape

Information is asset to both criminals and organizations

- Information 'assets' – by definition, someone else wants to get hold of them
- Individuals normally go through the proper channels – but not everyone will take the legal route
- Everyone is a target because virtually every organization (even a small business) holds valuable information (often in huge quantities)
- Being the most important asset, organizations cannot do business if they lose access to that information
- The fact that criminals can extract significant value from this information means that it is an asset to them too



The Threat Landscape

Personal Computing Assets (Remote Working)

- Threat landscapes commonly prioritize corporate and governmental networks assets as high priorities
 - Personal networks and resources are treated as lower-level threats
- Covid-19 pandemic resulted in over 40% of people working from home
 - This requires a reassessment of prioritization levels
- This change enabled bad actors with more opportunities to prey on remote workers
 - This forces reassessment of the risk level of home networks
- Today's threat landscape must also include personal computing assets as high-risk and high-value targets
 - This is because often-sensitive data being accessed outside of the protected corporate networks

The Threat Landscape



The Digital Divide

- The changing threat landscape has made a large segment of society to use technology securely
- People who may lack skills needed to protect themselves from security attacks now use their computers for education, work, and play
- In many situations, multiple family members utilize the same electronic device, greatly increasing the chance for exposure to malware
- Educational institutions are now required to quickly transition to online learning without implementing necessary training and cybersecurity protocols
 - These training and protocols are part of traditional online models
- Educators who may have not previously utilized technology are now sharing files as part of their daily online classroom interactions
 - This could introduce malware onto their devices

The Threat Landscape



Growing Internet of Things

- 
- Connected appliances
 - Smart home security systems
 - Autonomous farming equipment
 - Wearable health monitors
 - Smart factory equipment
 - Wireless inventory trackers
 - Ultra-high speed wireless internet
 - Biometric cybersecurity scanners
 - Shipping container and logistics tracking
 - Connected Cars
 - Connected Homes
 - Connected Agriculture
 - Connected Retail
 - Connected Hospitality
 - Connected Health
 - Connected Manufacturing
 - Connected Cities



The Threat Landscape

Growing Internet of Things

- A growing Internet of Things (IoT) has exposed devices to cyberattacks
 - A few years ago these would never have been included in most threat landscape models
- More healthcare and fitness apps for people to manage their health
 - This increases scope for attack surfaces
 - An attack on such apps, exposes large amounts of personal data and puts personal lives at risk
- Large number of payment apps such as GooglePay & PayTM
 - Such apps expose the possibility of stealing credit card and bank account information
- Modern agriculture equipment incorporates large amounts of technology
 - including data centers, networks, satellites and even artificial intelligence (AI)
 - a successful large-scale attack by either a lone individual or an organized group could potentially damage our food supply

The Threat Landscape



Increasing use of Social Media

- Greater numbers of individuals use social media as a news source
 - More than half of Americans receive their news by social media (Forbes)
- The manipulation of video using techniques such as **deepfake** make it increasingly difficult to recognize altered videos in social media
 - <https://youtu.be/EfREntgxmDs>
 - <https://edition.cnn.com/interactive/2019/01/business/pentagons-race-against-deepfakes/>
- Conspiracy theories are often shared online as facts, introducing yet more confusion in actual messaging to users looking for current news
- The risk of wireless technology remains constant
 - In addition, widespread use of 5G has introduces additional vulnerabilities



The Threat Landscape

Wireless Technology

- Previous mobile network topology provided for fewer pieces of hardware at which point traffic could be monitored
- The decentralized nature of 5G requires implementation of monitoring and security solutions at an exponentially greater number of devices
- The increased bandwidth and ability to add large numbers of IoT devices will require security solutions that are scalable and able to respond rapidly in order to provide a secure computing environment
- Understanding today's threat landscape is critical to developing strategies and solutions to establish a strong cybersecurity framework
- It is critical for both organizations and individuals to not become complacent and remain vigilant, regularly defending their threat landscape

The Threat Landscape



References

- The Cyber Security Handbook – Prepare for, respond to and recover from cyber attacks by Alan Calder *Published by IT Governance Publishing, 2020*
- UK Department for Digital, Culture, Media & Sport, “Cyber Security Breaches Survey 2020”, March 2020, <https://www.gov.uk/government/publications/cyber-security-breaches-survey-2020/cyber-security-breaches-survey-2020>.



Understanding Vulnerabilities

साने परमं बलं

Understanding Vulnerabilities



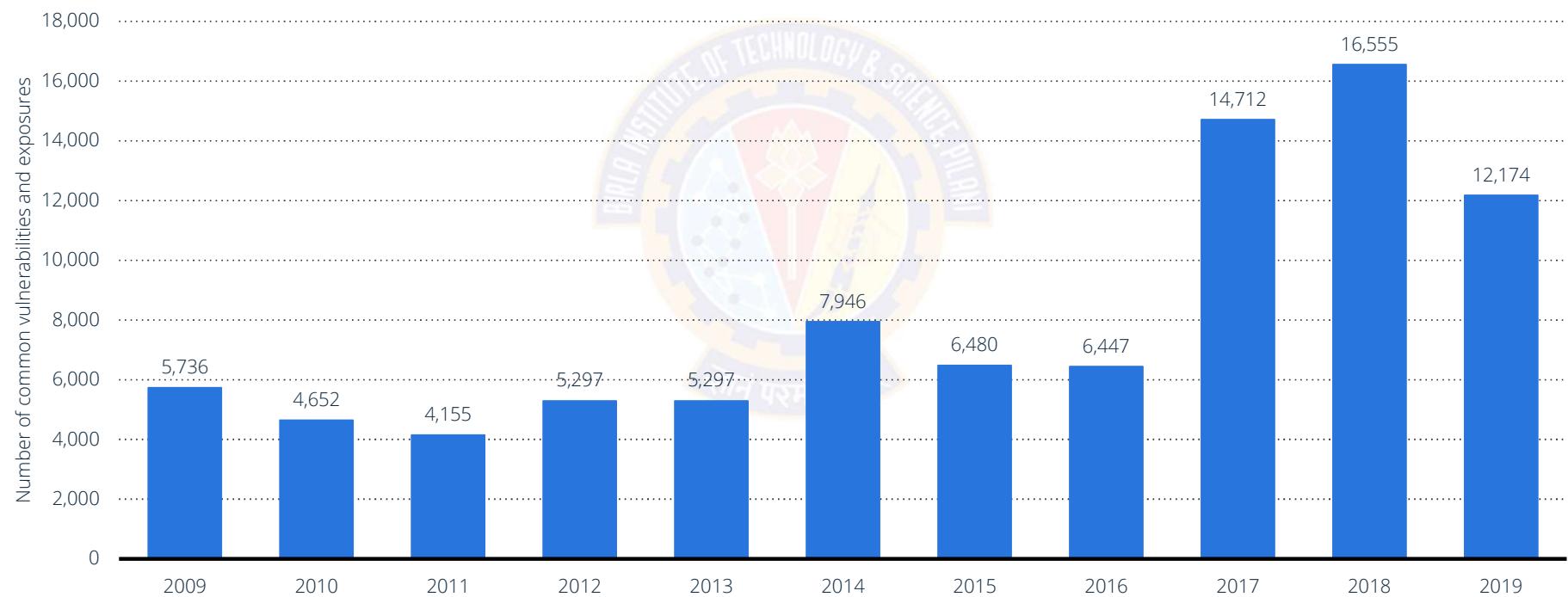
Overview

- A vulnerability is a weakness in an IT system that can be exploited by an attacker to deliver a successful attack
- Attackers will look to exploit any of them, often combining one or more, to achieve their end goal
- To exploit an existing vulnerability, an attacker needs to have at least one tool that connects to a system weakness:
 - The vulnerability then becomes what is known as the "attack surface".

Understanding Vulnerabilities



Common IT vulnerabilities and exposures worldwide 2009-2019



Note(s): Worldwide; 2009 to 2019
Source(s): Website (cvedetails.com); ID 500755

statista

Understanding Vulnerabilities



Vulnerability Categories

- Virtually, there can be 1000s of vulnerabilities
- However, they can be broadly grouped into following categories
 - Server and Host Vulnerabilities
 - Network Vulnerabilities
 - Virtualization Vulnerabilities
 - Web Application Vulnerabilities
 - Internet of Things Vulnerabilities
 - Database Vulnerabilities



Source: CompTIA CySA+ (CS0-001): Complete Course and Practice Exam
Source: <https://www.ncsc.gov.uk/information/understanding-vulnerabilities>

Understanding Vulnerabilities



About MITRE

- The MITRE Corporation is an American not-for-profit organization based in Bedford, Massachusetts, and McLean, Virginia
- It manages federally funded R&D centers (FFRDCs) supporting several U.S. government agencies
- MITRE maintains the Common Vulnerabilities and Exposures (CVE) system and the Common Weakness Enumeration (CWE) project
- Since 1999, the MITRE Corporation has been functioning as editor and primary numbering authority of the CVEs
- CVE is now the industry standard for vulnerability and exposure names
- It provides reference points for data exchange so that information security products and services can interoperate with each other

Source: <https://www.rapid7.com/fundamentals/vulnerabilities-exploits-threats/>

Understanding Vulnerabilities



Common Computer Security Vulnerabilities - 2020

- The Common Weakness Enumeration (CWE) identified the Top 25 Most Dangerous Software Errors
- The CWE Top 25 provides insight into the most severe and current security weaknesses
- This is a demonstrative list of the most common and impactful issues experienced over the previous two calendar years
- While the list remains comprehensive, there are many other threats that leave software vulnerable to attack
- These weaknesses are dangerous because they are often easy to find, exploit, and can allow adversaries to completely take over a system, steal data, or prevent an application from working

Understanding Vulnerabilities



Common Computer Security Vulnerabilities

Vulnerability	Score	Vulnerability	Score
Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')	46.82	Exposure of Sensitive Information to an Unauthorized Actor	19.16
Out-of-bounds Write	46.17	Use After Free	18.87
Improper Input Validation	33.47	Cross-Site Request Forgery (CSRF)	17.29
Out-of-bounds Read	26.50	Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')	16.44
Improper Restriction of Operations within the Bounds of a Memory Buffer	23.73	Integer Overflow or Wraparound	15.81
Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')	20.69	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')	13.67

2020 CWE Top 25 Most Dangerous Software Weaknesses

https://cwe.mitre.org/top25/archive/2020/2020_cwe_top25.html

Understanding Vulnerabilities



Common Computer Security Vulnerabilities

Vulnerability	Score	Vulnerability	Score
NULL Pointer Dereference	8.35	Use of Hard-coded Credentials	5.19
Improper Authentication	8.17	Deserialization of Untrusted Data	4.93
Unrestricted Upload of File with Dangerous Type	7.38	Improper Privilege Management	4.87
Incorrect Permission Assignment for Critical Resource	6.95	Uncontrolled Resource Consumption	4.14
Improper Control of Generation of Code ('Code Injection')	6.53	Missing Authentication for Critical Function	3.85
Insufficiently Protected Credentials	5.49	Missing Authorization	3.77
Improper Restriction of XML External Entity Reference	5.33		

2020 CWE Top 25 Most Dangerous Software Weaknesses

https://cwe.mitre.org/top25/archive/2020/2020_cwe_top25.html

Understanding Vulnerabilities



Causes of Vulnerabilities

- They can occur through:
 - Flaws
 - Features
 - User error
 - Zero-day vulnerabilities



Source: CompTIA CySA+ (CS0-001): Complete Course and Practice Exam
Source: <https://www.ncsc.gov.uk/information/understanding-vulnerabilities>

Understanding Vulnerabilities



Causes of Vulnerabilities

- Flaws
 - A flaw is an unintended functionality
 - This may either be a result of poor design or through mistakes made during implementation (coding)
 - Flaws may go undetected for a significant period of time
 - The majority of common attacks we see today exploit these types of vulnerabilities
 - Between 2014 and 2015, nearly 8,000 unique and verified software vulnerabilities were disclosed in the US National Vulnerability Database (NVD)
 - Vulnerabilities are actively pursued and exploited by the full range of attackers
 - Consequently, a market has grown in software flaws, with 'zero-day' vulnerabilities fetching hundreds of thousands of dollars

Understanding Vulnerabilities



Causes of Vulnerabilities

- Features

- A feature is intended functionality which can be misused by an attacker to breach a system
- Features may improve the user's experience, help diagnose problems or improve management, but they can also be exploited by an attacker
- Example:
 - Microsoft introduced macros into their Office suite in the late 1990s. They soon became the vulnerability of choice
 - E.g., Melissa virus in March, 1999
 - It was a mass-mailing macro virus. It targeted Microsoft Word and Outlook-based systems, and created considerable network traffic
 - The virus would infect computers via Email, the email being titled "Important Message From", followed by the current username
 - Upon clicking the message, the body would read: "Here's that document you asked for. Don't show anyone else ;)." Attached was a Word document titled list.doc containing a list of pornographic sites and accompanying logins for each
 - It would then mass mail itself to the first 50 people in the user's contact list and then disable multiple safeguard features on Microsoft Word and Microsoft Outlook

Understanding Vulnerabilities



Causes of Vulnerabilities

- Features
 - Macros are still exploited today
 - The Dridex banking Trojan that was spreading in late 2014 relies on spam to deliver Microsoft Word documents containing malicious macro code, which then downloads Dridex onto the affected system.
 - JavaScript, widely used in dynamic web content, continues to be used by attackers
 - E.g., Diverting the user's browser to a malicious website and silently downloading malware, and hiding malicious code to pass through basic web filtering.

Understanding Vulnerabilities



Causes of Vulnerabilities

- User Error
 - Users can be a significant source of vulnerabilities
 - They make mistakes, such as choosing a common or easily guessed password, or leaving their laptop or mobile phone unattended
 - Even the most cyber aware users can be fooled into giving away their password, installing malware, or divulging confidential information
 - These details would allow an attacker to target and time an attack appropriately
 - A carefully designed and implemented computer system can minimize vulnerabilities
 - Such efforts can be easily undone
 - E.g., an inexperienced system administrator who enables vulnerable features, fails to fix a known flaw, or leaves default passwords unchanged

Understanding Vulnerabilities



Causes of Vulnerabilities

- Zero-day vulnerabilities
 - The term "zero-day" refers to a newly discovered software vulnerability
 - Because the developer has just learned of the flaw, it also means an official patch or update to fix the issue hasn't been released
 - So, "zero-day" refers to the fact that the developers have "zero days" to fix the problem that has just been exposed — and perhaps already exploited by hackers
 - Once the vulnerability becomes publicly known, the vendor has to work quickly to fix the issue to protect its users.
 - But the software vendor may fail to release a patch before hackers manage to exploit the security hole
 - That's known as a zero-day attack.

Understanding Vulnerabilities



Causes of Vulnerabilities

- Vulnerabilities are not just software-based
- Vulnerabilities can be found on software, hardware, network, even the users — impacting all assets across an organization
- Vulnerabilities can come from many sources, complexity, misconfiguration, connectivity, software bugs, etc.
- The most common source of vulnerabilities is the human user
 - Which poses a significant risk for organizations and their security posture.

Understanding Vulnerabilities



Common Vulnerability Scoring System (CVSS)

- While software bugs aren't inherently harmful (except for potential performance issues), many can be taken advantage of by "bad" actors
 - These are known as vulnerabilities.
- Vulnerabilities can be leveraged to force software to act in ways it's not intended to
 - E.g., gleaning information about the current security defenses in place.
- Once a bug is determined to be a vulnerability, it is registered by MITRE as a CVE, or common vulnerability or exposure
- The vulnerability is assigned a Common Vulnerability Scoring System (CVSS) score to reflect the potential risk it could introduce to the organization
- This central listing of CVEs serves as a reference point for vulnerability scanners

Understanding Vulnerabilities



Scanning Vulnerabilities

- A vulnerability scanner is a computer program designed to assess computers, networks or applications for known weaknesses
- They are used to identify and detect vulnerabilities arising from mis-configurations or flawed programming within a network-based asset such as a firewall, router, web server, application server, etc.,
- A vulnerability scanner scans and compares an organization's environment against a vulnerability database, or a list of known vulnerabilities
- Once the vulnerabilities are detected, developers can use penetration testing as a means to see where the weaknesses are
- These problems can be fixed and future mistakes can be avoided
- Frequent and consistent scanning will enable us to see common threads between vulnerabilities and a better understanding of the system

Understanding Vulnerabilities



Threat-Vulnerability-Risk

- Before we discuss identifying vulnerabilities, we need to understand threat and risk
- Threat
 - A potential for an attacker to exploit a vulnerability in some asset, which if it occurs may compromise the security of the asset and cause harm to the asset's owner
- Risk
 - The potential for loss computed as the combination of the *likelihood* that an attacker exploits some vulnerability to an asset, and the *magnitude* of harmful consequence that results to the asset's owner

Understanding Vulnerabilities



Threat-Vulnerability-Risk

- Not all vulnerabilities are a security risk
- For example:
 - The risk of a vulnerability can depend on the potential impact that it could have on the business, in relation to which asset it impacts
- If the vulnerability is on a low-risk asset then it is much less likely of posing a significant risk
- The risk also depends on the time a vulnerability has existed
- A vulnerability which has been identified and quickly addressed poses much less risk than one that goes undetected for days, weeks, or even months

Understanding Vulnerabilities



Threat-Vulnerability-Risk

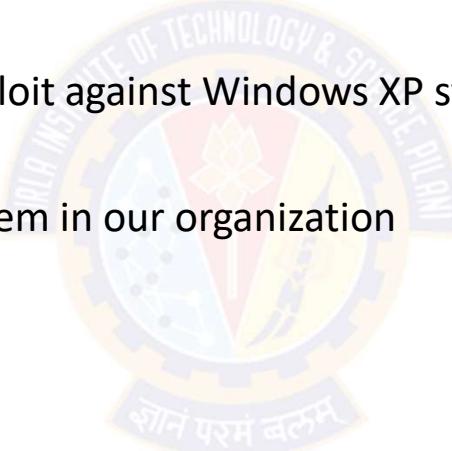
- Identifying potentially significant risks to the assets requires answering the following questions for each asset:
 - Who or what could cause it harm?
 - This involves identifying potential threats to assets
 - How could this occur?
 - This involves identifying flaws or weaknesses in the organization's IT systems or processes that could be exploited by a threat source
- Mere existence of some vulnerability does not mean harm will be caused to an asset
 - There must also be a threat source for some threat that can exploit the vulnerability
- The combination of a *threat* and a *vulnerability* creates a risk to an asset

Understanding Vulnerabilities



Threat-Vulnerability-Risk

- If you have a threat without a vulnerability, it isn't a risk
 - Threat
 - Hackers are using zero-day exploit against Windows XP systems
 - Vulnerability
 - We don't use Windows XP system in our organization
 - Risk
 - None

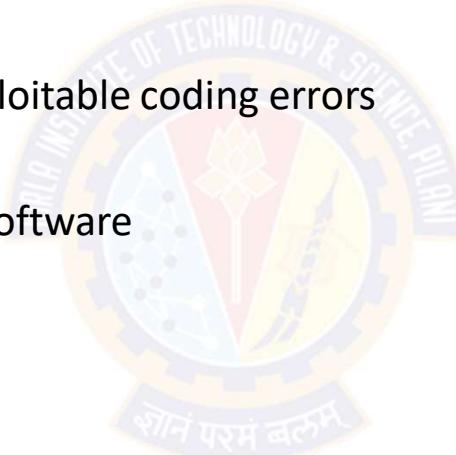


Understanding Vulnerabilities



Threat-Vulnerability-Risk

- If you have a vulnerability without a threat, it isn't a risk
 - Threat
 - Hackers haven't found any exploitable coding errors
 - Vulnerability
 - Unpatched operating system software
 - Risk
 - None





Sources of Vulnerability Information

Security Mailing Lists

- The following mailing lists contain interesting and useful discussion relating to current security vulnerabilities and issues
 - BugTraq (<http://www.securityfocus.com/archive/1>)
 - Full Disclosure (<http://seclists.org/fulldisclosure/>)
 - Pen-Test (<http://www.securityfocus.com/archive/101>)
 - Web Application Security (<http://www.securityfocus.com/archive/107>)
 - Honeypots (<http://www.securityfocus.com/archive/119>)
 - CVE Announce (<http://archives.neohapsis.com/archives/cve/>)
 - Nessus development (<http://list.nessus.org>)
 - Nmap-hackers (<http://seclists.org/nmap-hackers/>)
 - VulnWatch (<http://www.vulnwatch.org>)



Sources of Vulnerability Information

Vulnerability Databases

- The following vulnerability databases and lists can be searched to enumerate vulnerabilities in specific technologies and products:
 - MITRE CVE (<http://cve.mitre.org>)
 - NIST NVD (<http://nvd.nist.gov>)
 - ISS X-Force (<http://xforce.iss.net>)
 - OSVDB (<http://www.osvdb.org>)
 - BugTraq (<http://www.securityfocus.com/bid>)
 - CERT vulnerability notes (<http://www.kb.cert.org/vuls>)
 - FrSIRT (<http://www.frsirt.com>)

Sources of Vulnerability Information

Underground Web Sites

- The following underground web sites contain useful exploit scripts and tools that can be used during penetration tests:

- | | |
|--|---|
| <ul style="list-style-type: none">• Milw0rm (http://www.milw0rm.com)• Raptor's labs (http://www.0xdeadbeef.info)• H D Moore's pages (http://www.metasploit.com/users/hdm/)• The Hacker's Choice (http://www.thc.org)• Packet Storm (http://www.packetstormsecurity.org)• Insecure.org (http://www.insecure.org)• Top 100 Network Security Tools (http://sectools.org)• IndianZ (http://www.indianz.ch)• Zone-H (http://www.zone-h.org)• Phenoelit (http://www.phenoelit.de)• Uninformed (http://uninformed.org) | <ul style="list-style-type: none">• Astalavista (http://astalavista.com)• cqure.net (http://www.cqure.net)• TESO (http://www.team-teso.net)• ADM (http://adm.freelsd.net/adm/)• Hack in the box (http://www.hackinthebox.org)• cnhonker (http://www.cnhonker.com)• Soft Project (http://www.s0ftpj.org)• Phrack (http://www.phrack.org)• LSD-PLaNET (http://www.lsd-pl.net)• w00w00 (http://www.w00w00.org)• Digital Offense (http://www.digitaloffense.net) |
|--|---|



Sources of Vulnerability Information

Vulnerability Databases

- <https://cve.mitre.org/>
 - Common Vulnerabilities and Exposures (CVE®) is a list of common identifiers for publicly known cybersecurity vulnerabilities
- <https://nvd.nist.gov/>
 - The National Vulnerability Database (NVD) is the U.S. government repository of standards based vulnerability management data represented using the Security Content Automation Protocol (SCAP)
 - This data enables automation of vulnerability management, security measurement, and compliance
 - The NVD includes databases of security checklist references, security related software flaws, misconfigurations, product names, and impact metrics.



Thank You!