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

innovate achieve lead

Agenda

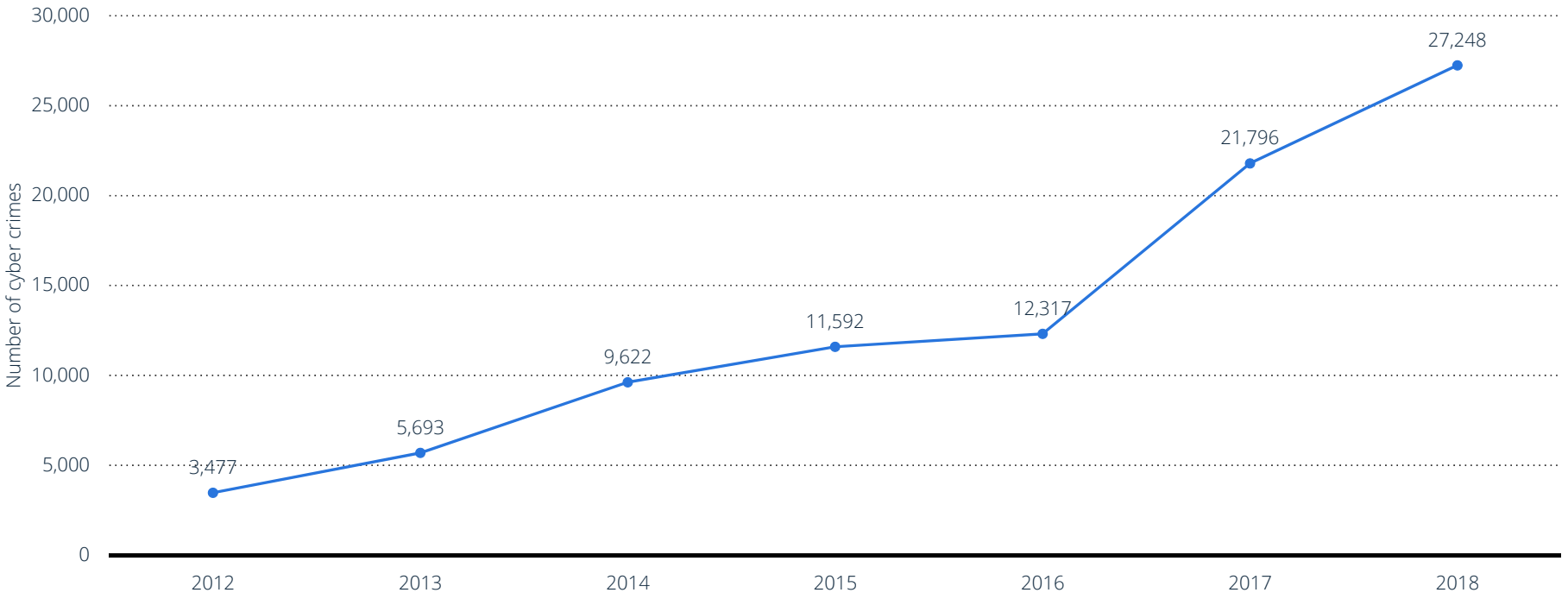
- Computer Security Concepts
- Threats, Attacks, and Assets
- Security Functional Requirements
- Fundamental Security Design Principles
- Attack Surfaces and Attack Trees
- Computer Security Strategy
- Standards

Computer Security Concepts

Some Facts

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

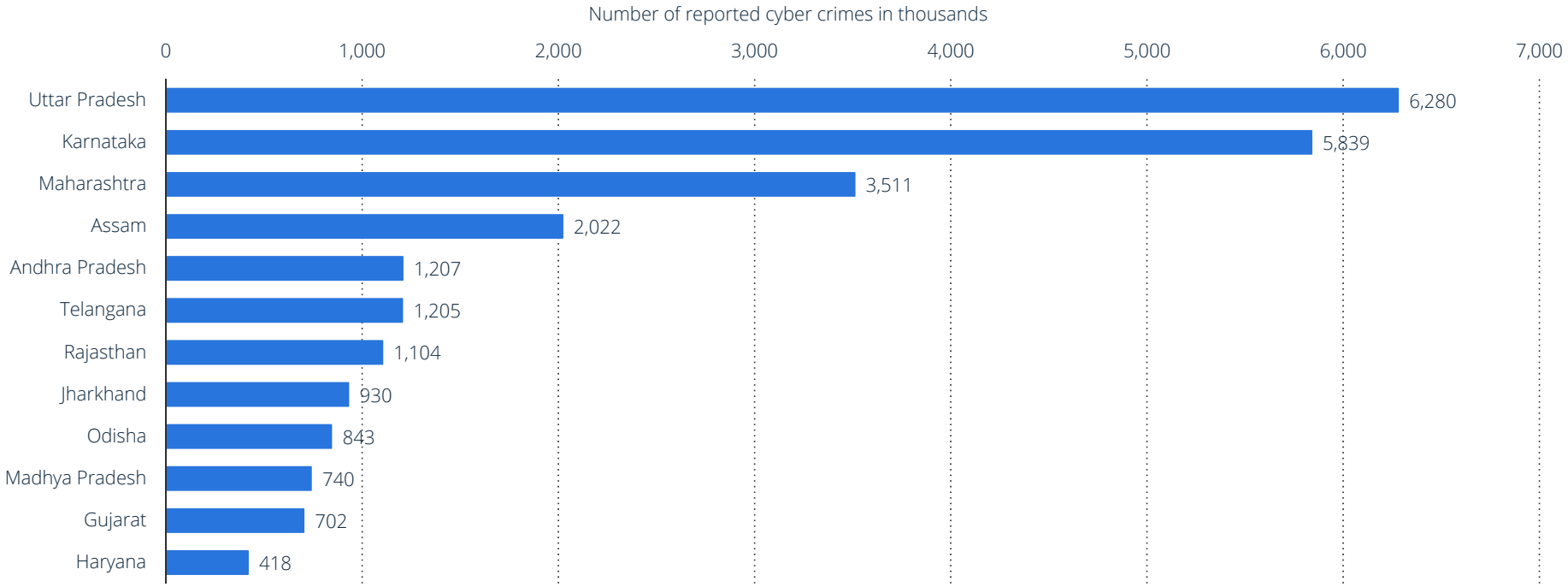
Total number of cyber crimes reported in India 2018



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)

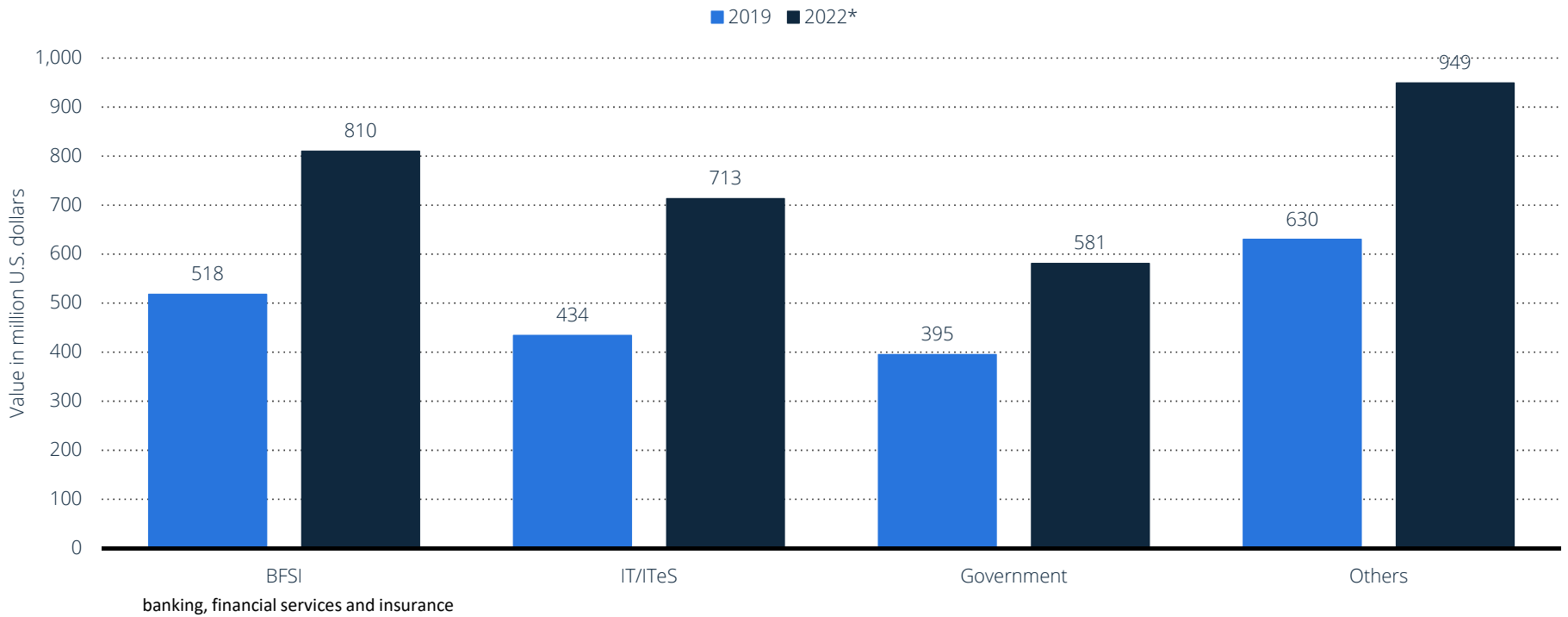
Number of cyber crimes reported in India 2018 by leading state



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

innovate achieve lead

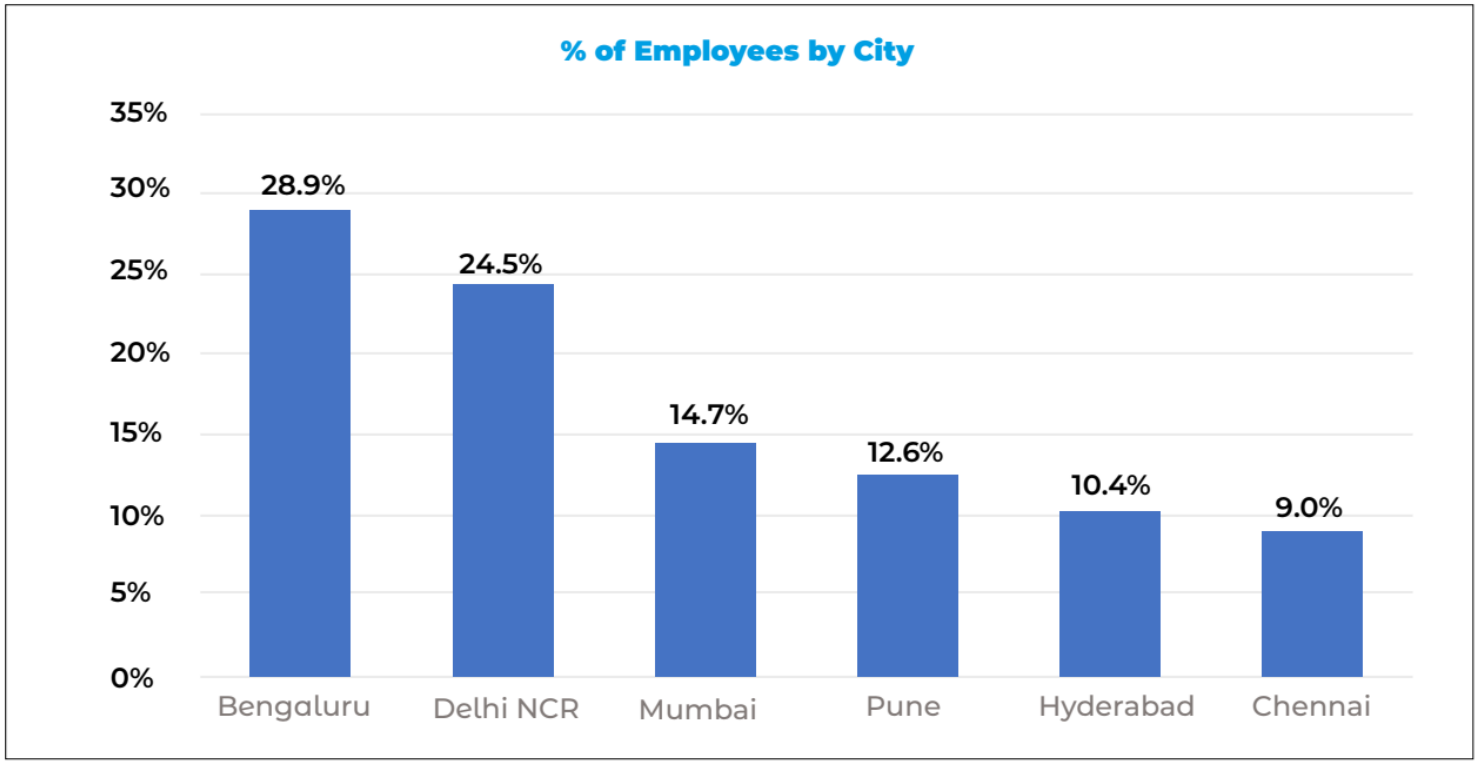
Cyber Security Expenditure in India: 2019-2022

- 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

innovate achieve lead

Cyber Security Employee Distribution: 2020

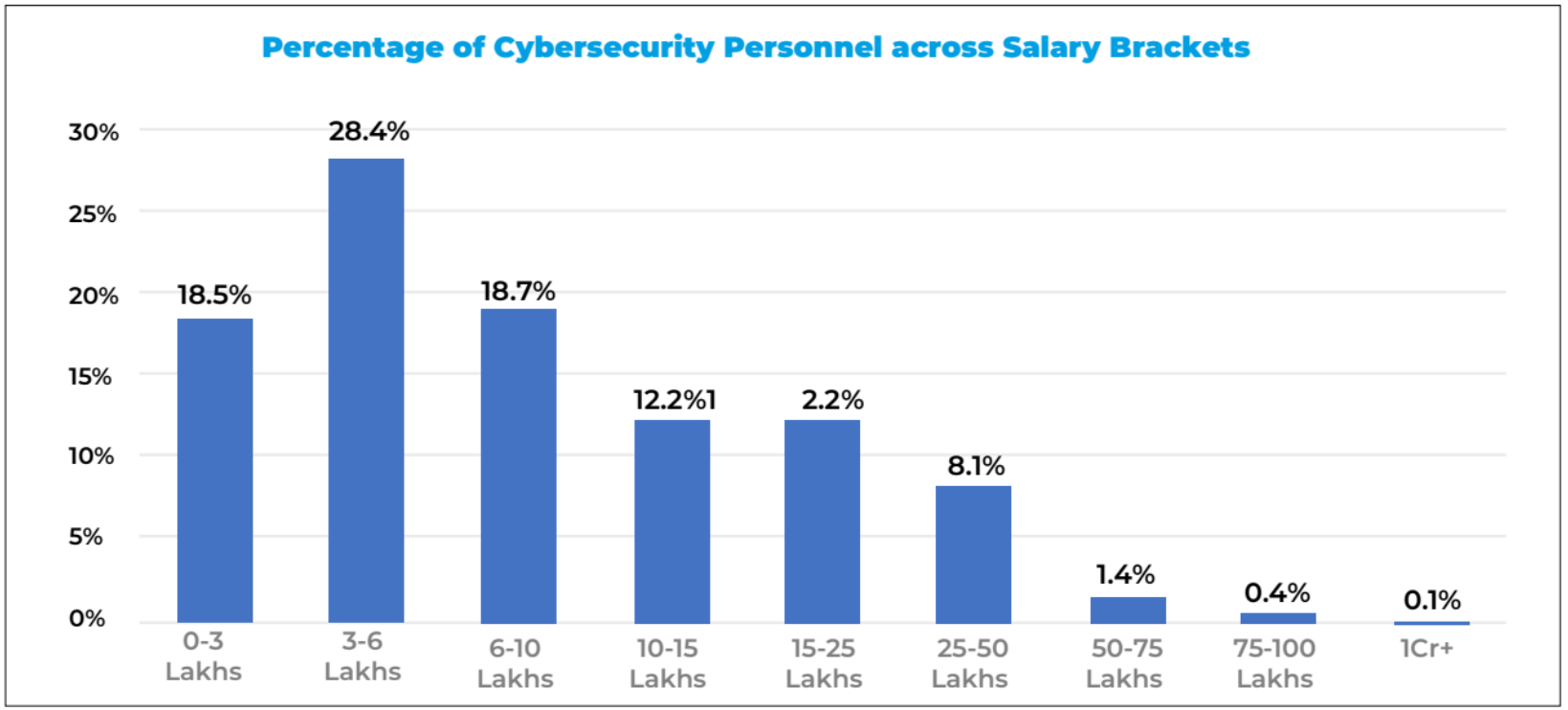


Source: State of Cyber Security in India by Jigsaw Academy & AIMResearch

Some Facts

innovate achieve lead

Cyber Security Personnel Salary Brackets: 2020

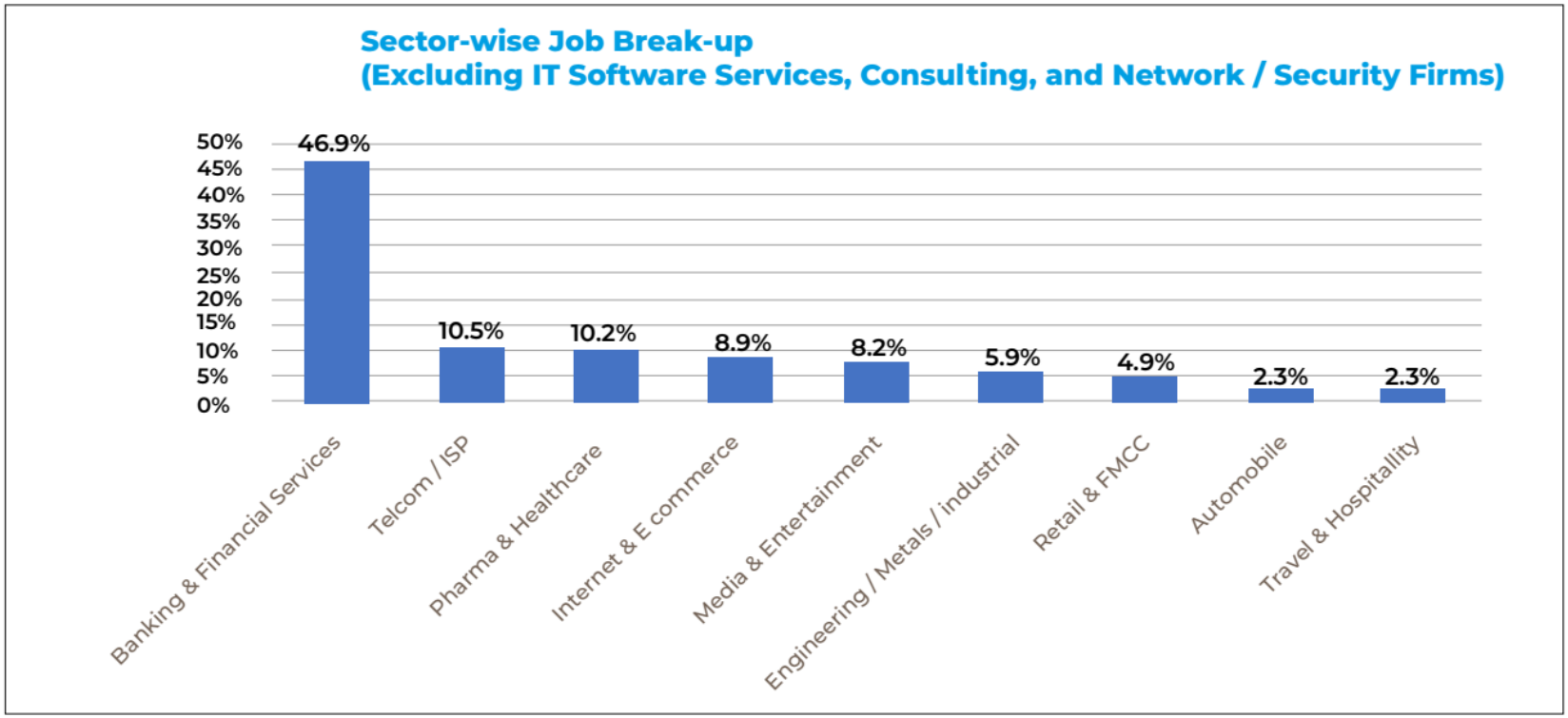


Source: State of Cyber Security in India by Jigsaw Academy & AIMResearch

Some Facts

innovate achieve lead

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

innovate achieve lead

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

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

innovate achieve lead

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)"*

NIST: National Institute of Standards and Technology

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

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

innovate achieve lead

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

https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.199.pdf
https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.200.pdf

Computer Security Concepts

innovate achieve lead

Key objectives of Computer Security

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

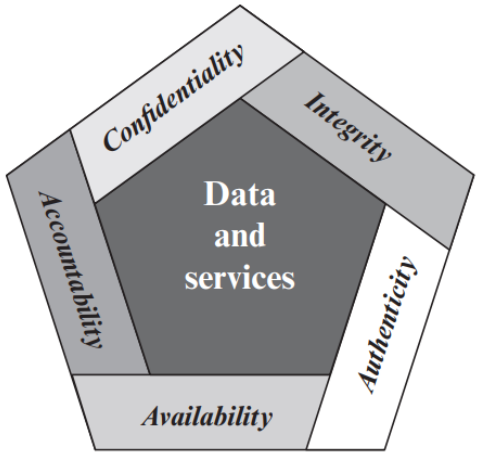
Category	Requirement	Definition of a loss of security
Confidentiality	• Preserving authorized restrictions on information access and disclosure • Includes means for protecting personal privacy and proprietary information	• A loss of confidentiality is the unauthorized disclosure of information
Integrity:	• Guarding against improper modification or destruction of information • Includes ensuring information nonrepudiation and authenticity	• A loss of integrity is the unauthorized modification or destruction of information
Availability:	• Ensuring timely and reliable access to and use of information.	• A loss of availability is the disruption of access to or use of information or an Information System

FIPS 199: Standards for Security Categorization of Federal Information and Information Systems

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

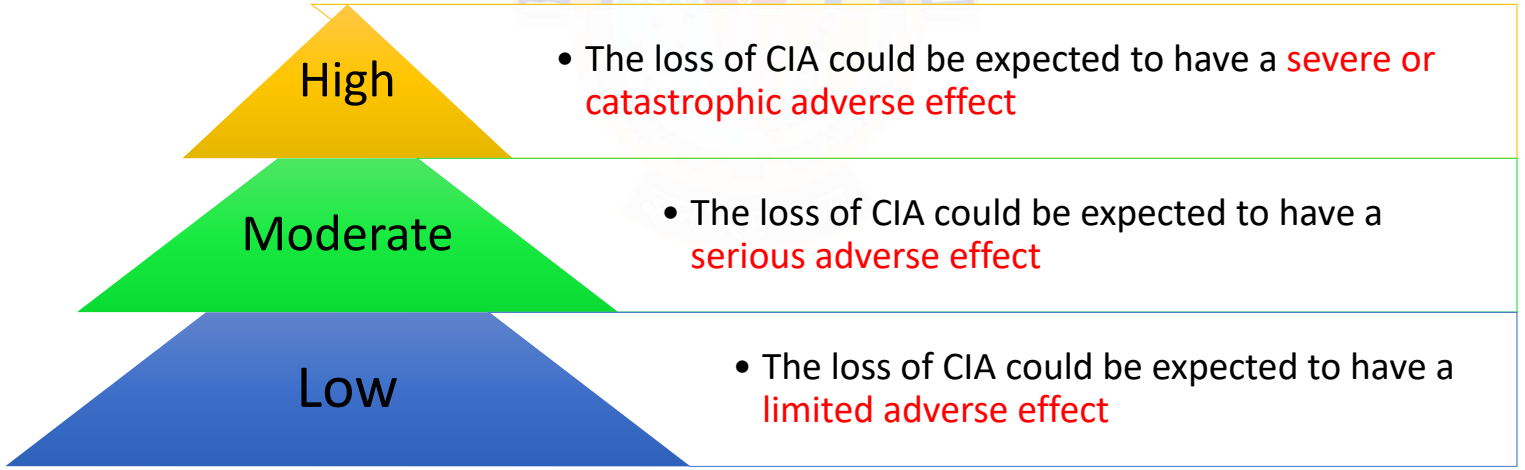


Essential Network and Computer Security Requirements

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

	Breach of Security		
Effect on	**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)	• 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	• This information is seen by more people on a daily basis • Is less likely to be targeted than grade information • Results in less damage if disclosed
Low	Directory information	Not covered by FERPA	• E.g., lists of students or faculty or departmental lists • This 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	• The doctor should be able to trust that the information is correct and current • Now suppose that a nurse who is authorized to access this information deliberately falsifies the data to cause harm to the hospital • The database needs to be restored to a trusted basis quickly • It should be possible to trace the error back to the person responsible • Inaccurate information could result in serious harm or death to a patient and expose the hospital to massive liability
Moderate	Web site	• Offers a forum to registered users to discuss specific topics • Either a registered user or a hacker could falsify some entries or deface the Web site • If 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 severe • The Web master may experience some data, financial, and time loss
Low	Anonymous online poll	• Many Web sites (E.g., news organizations), run polls for their users with very few safeguards • However, the inaccuracy and unscientific nature of such polls is well understood.

Loss to CIA Triad – Availability – Example

Availability	Example	Details

High	A system that provides authentication services for critical systems, applications, and devices	• An interruption of service results in the inability for
		  • customers to access computing resources
		  • staff 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	• The Web site provides information for current and prospective students and donors • Such 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	• The temporary loss of the application may be an annoyance, but • There are other ways to access the information, such as a hardcopy directory or the operator

Challenges in Computer Security

Computer Security Concepts

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.

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

TCP/IP (Transmission Control Protocol/Internet Protocol)

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

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

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

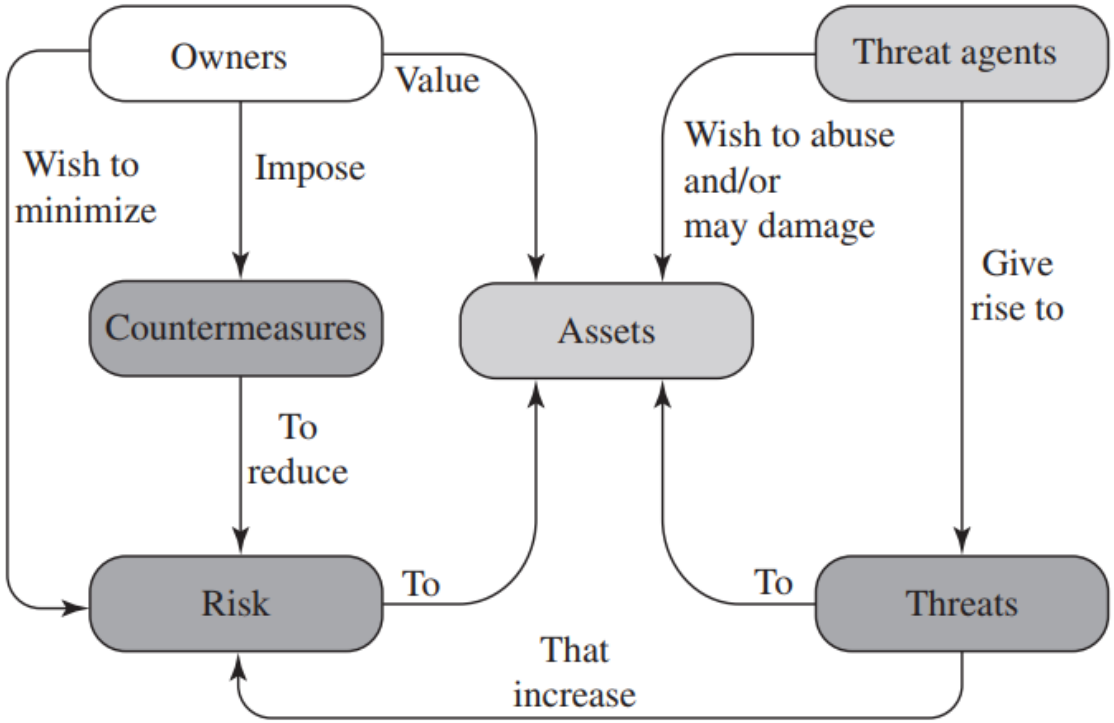
Terminology

- 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

innovate achieve lead

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

innovate achieve lead

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

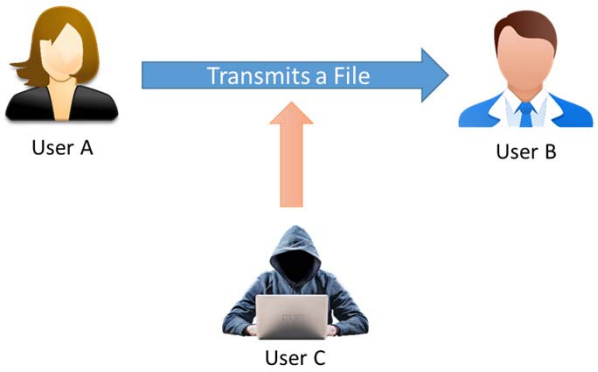
Threats & Attacks

innovate achieve lead

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 a;., 2019, Traffic Analysis Attack for Identifying Users' Online Activities, Published in IT Professional 2019

Threats & Attacks

innovate achieve lead

Unauthorized Disclosure

Threat Action	Types of Threat Actions	Description
Exposure *A threat action whereby sensitive data is directly released to an unauthorized entity.*	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

innovate achieve lead

Unauthorized Disclosure

Threat Action	Types of Threat Actions	Description
Intrusion *A threat action whereby an unauthorized entity gains access to sensitive data by circumventing a system's security protections.*	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 *A threat action whereby an unauthorized entity directly accesses sensitive data travelling between authorized sources and destinations*	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

innovate achieve lead

Unauthorized Disclosure

Threat Action	Types of Threat Actions	Description
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 byproducts of communications*	Traffic Analysis	Gaining knowledge of data by observing the characteristics of communications that carry the data.
	Signal Analysis	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.

Threats & Attacks

innovate achieve lead

Deception

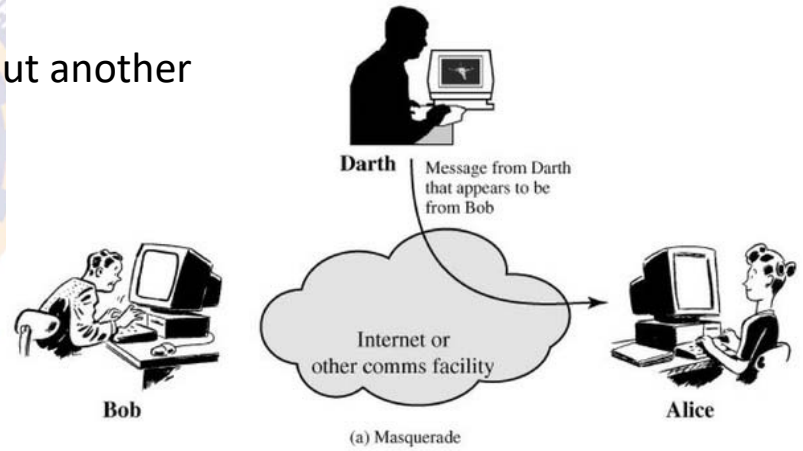
- A circumstance or event that may result in an authorized entity receiving false data and believing it to be true
- The following threat actions can cause deception:
  - Masquerade
    - A threat action whereby an unauthorized entity gains access to a system or performs a malicious act by posing as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

innovate achieve lead

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



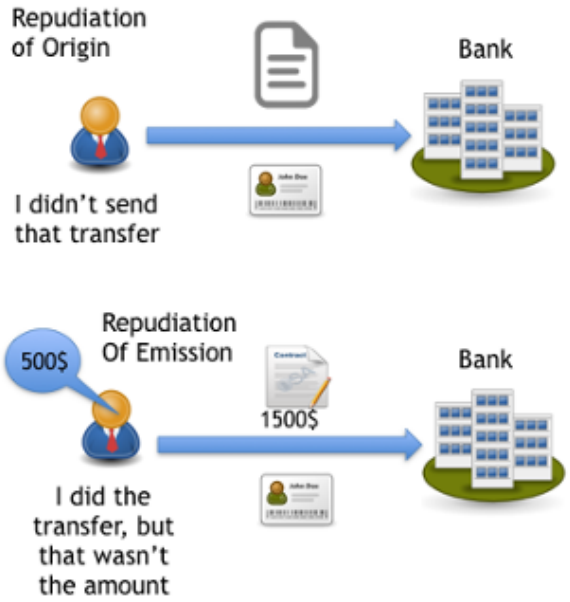
Active Attack – Masquerade

Threats & Attacks

innovate achieve lead

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 *A threat action whereby an unauthorized entity gains access to a system or performs a malicious act by posing as an authorized entity.*	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 *A threat action whereby false data deceives an authorized entity*	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 *A threat action whereby an entity deceives another by falsely denying responsibility for an act.*	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

innovate achieve lead

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

innovate achieve lead

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 *Prevents or interrupts system operation by disabling a system component*	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

innovate achieve lead

Disruption

Threat Action	Types of Threat Actions	Description
Corruption *A threat action that undesirably alters system operation by adversely modifying system functions or data.*	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 *A threat action that interrupts delivery of system services by hindering system operations.*	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 *An entity assumes unauthorized logical or physical control of a system resource.*	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 *A threat action that causes a system component to perform a function or service that is detrimental to system security.*	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 *A circumstance or event whereby an entity gains access to data for which the entity is not authorized*	Exposure: Sensitive data are directly released to an unauthorized entity. Interception: An unauthorized entity directly accesses sensitive data traveling between authorized sources and destinations. 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. Intrusion: An unauthorized entity gains access to sensitive data by circumventing a system's security protections.
Deception *A circumstance or event that may result in an authorized entity receiving false data and believing it to be true*	Masquerade: An unauthorized entity gains access to a system or performs a malicious act by posing as an authorized entity. Falsification: False data deceive an authorized entity. Repudiation: An entity deceives another by falsely denying responsibility for an act.

Threats & Attacks

innovate achieve lead

Summary

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

Threats & Assets

Threats & Assets

Categories

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

Threats & Assets

innovate achieve lead

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

innovate achieve lead

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

innovate achieve lead

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

innovate achieve lead

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.

Threats & Assets

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

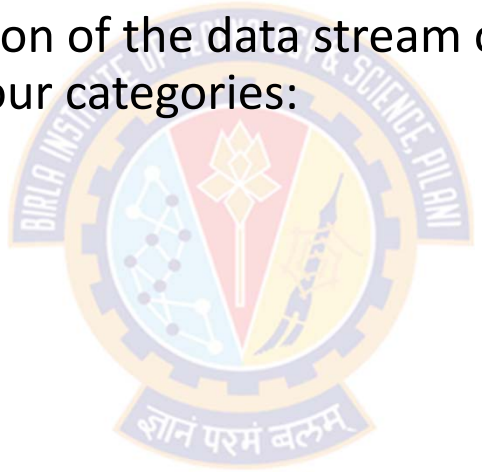
Communication Lines and Networks

- 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

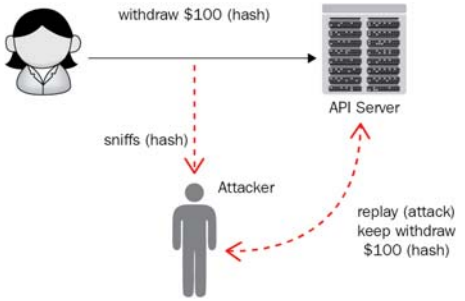
- Active attacks
  - They involve some modification of the data stream or the creation of a false stream and can be subdivided into four categories:
    - Replay
    - Masquerade
    - Modification of messages, and
    - Denial of service.



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

Threats & Assets

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

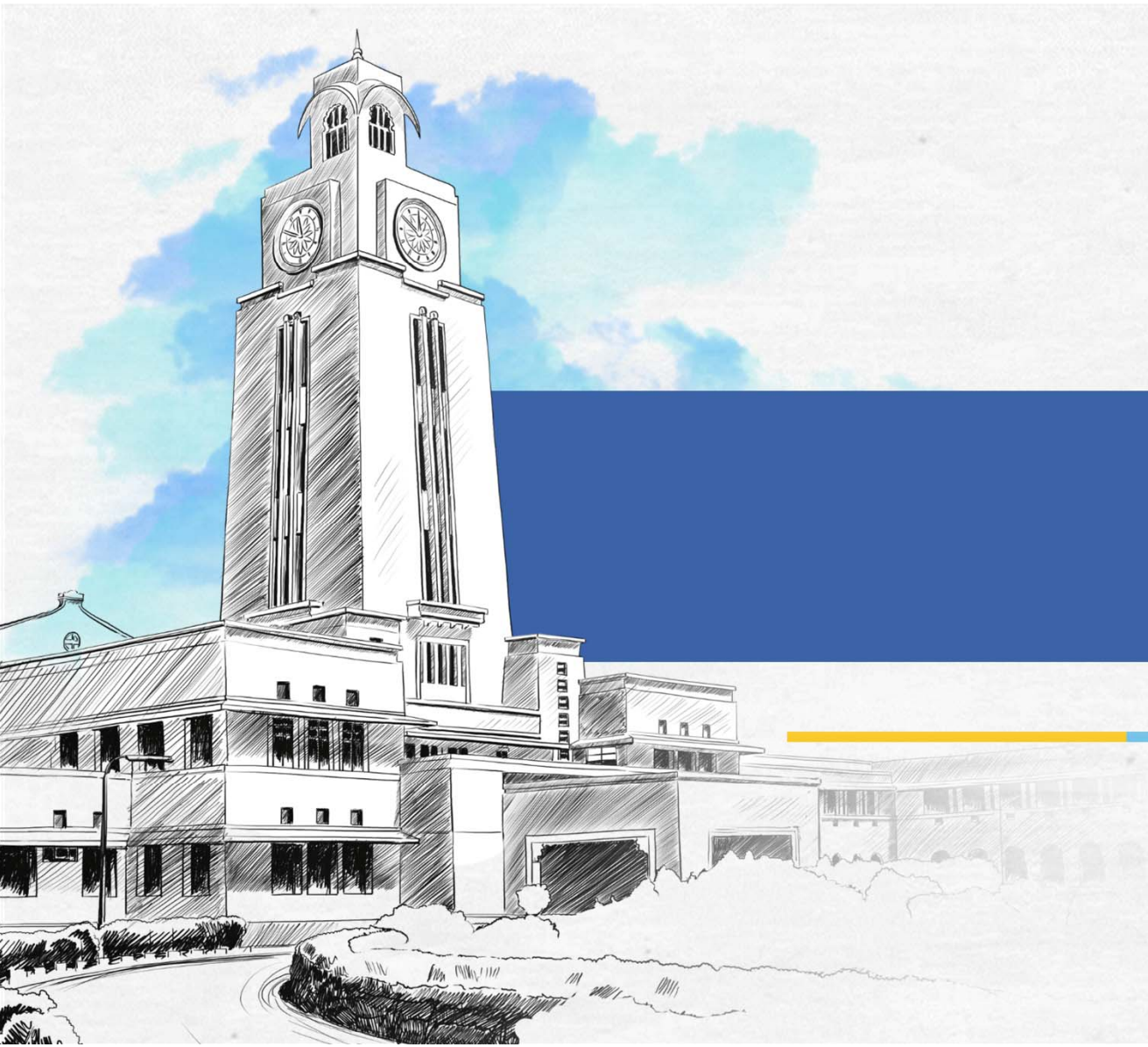
References

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


innovate
achieve
lead

BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI
ज्ञानं परमं बलम्

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

innovate achieve lead

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

innovate achieve lead

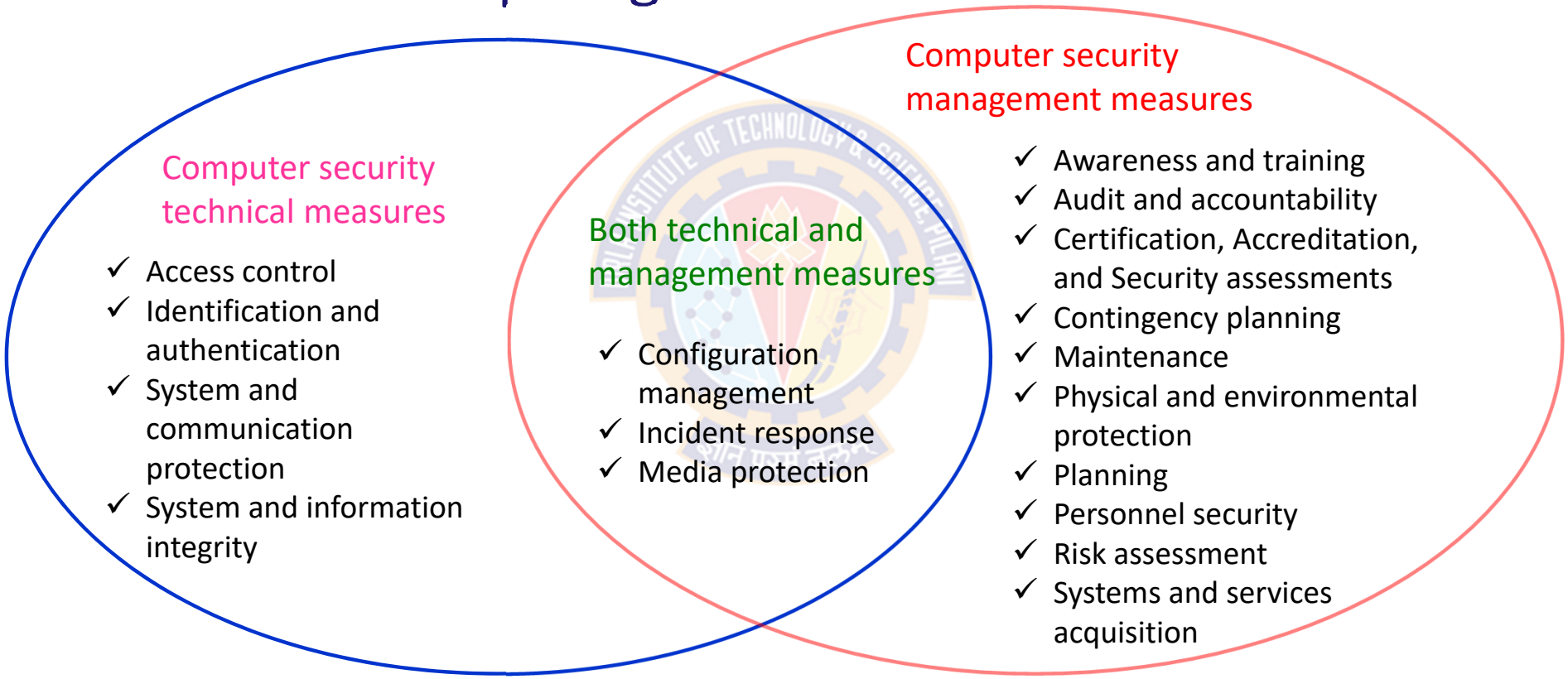
Classifying & Characterizing Countermeasures

- 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

innovate achieve lead

Functional Areas Requiring...



Security Functional Requirements

innovate achieve lead

Functional areas involving technical measures

Term	Description
Access Control	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
Identification and Authentication	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
System and Communication Protection	(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
System and Information Integrity	(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.

OIS = Organizational Information Systems; IS = Information System; OC = Organizational Communication

Security Functional Requirements

innovate achieve lead

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

OIS = Organizational Information Systems; IS = Information System;

Security Functional Requirements

innovate achieve lead

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

innovate achieve lead

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

OIS = Organizational Information Systems; IS = Information System;

Security Functional Requirements

innovate achieve lead

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.

OIS = Organizational Information Systems; IS = Information System;

Fundamental Security Design Principles

Ordering Pizza

innovate achieve lead

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

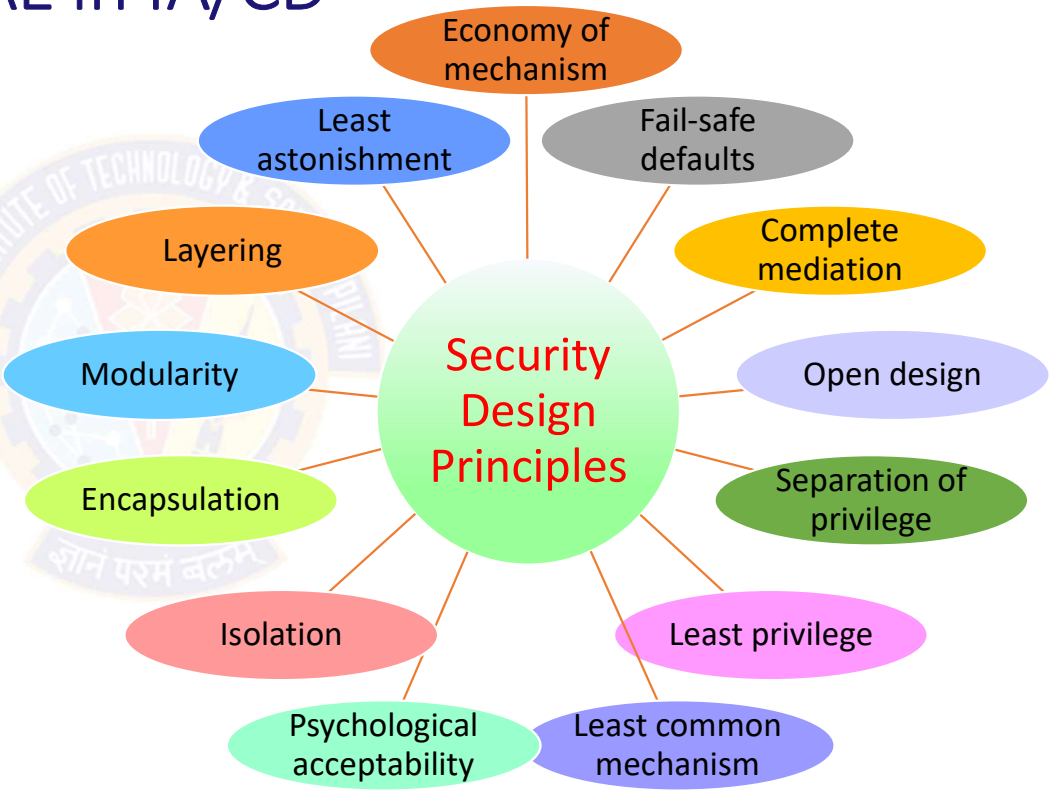
innovate achieve lead

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



NCAE: National Centers of Academic Excellence in Information Assurance/Cyber Defense

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

innovate achieve lead

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	Default is permit access
Involves explicitly giving permission	Involves explicitly excluding access
Exhibits better failure mode than the default permit access approach	Exhibits poor failure mode than the default lack of access
Implementation mistake (giving explicit permission) only results in refusing permission, which is a safe situation and can be quickly detected	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 of 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

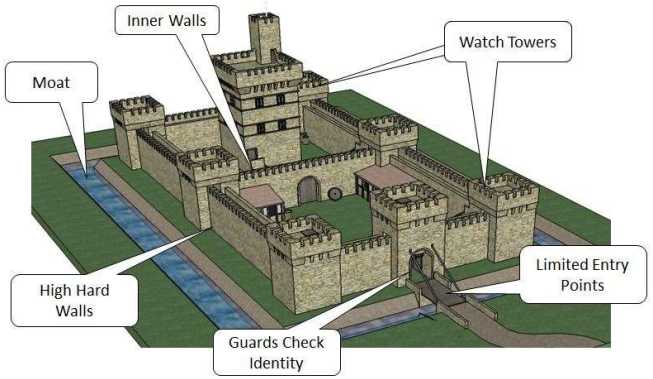
NIST = National Institute of Standards and Technology

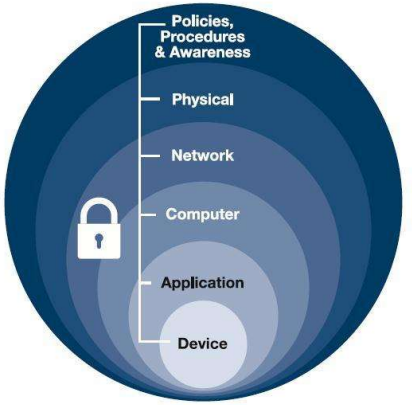
Security Design Principles

innovate achieve lead

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



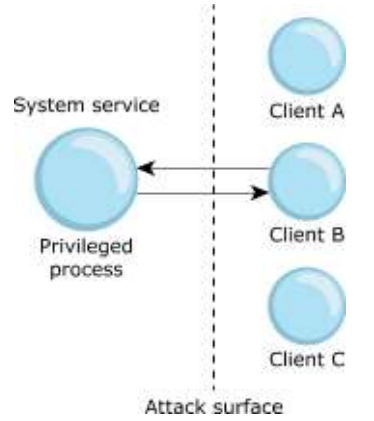


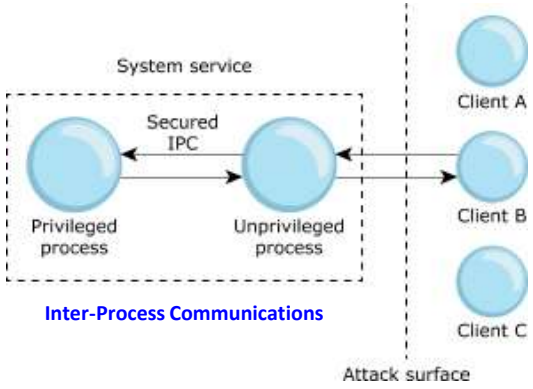
Security Design Principles

innovate achieve lead

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.





http://www.qnx.com/developers/docs/qnxcar2/index.jsp?topic=%2Fcom.qnx.doc.neutrino.prog%2Ftopic%2Fprocess_Privilege_separation.html

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

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

innovate achieve lead

Modularity

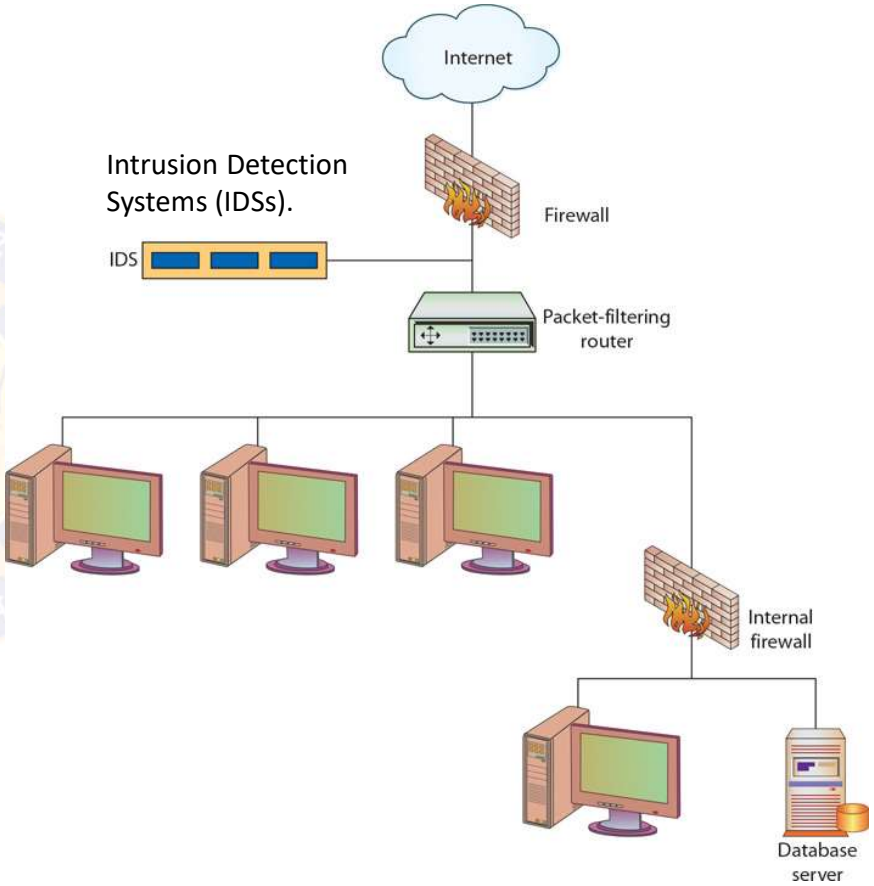
- Modularity principle says that the security mechanism must be developed:
  - as separate and protected modules, and
  - using the modular architecture.
- The design goal here is to provide security functions and services, such as cryptographic functions, as common modules
- For example:
  - numerous 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

innovate achieve lead

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

innovate achieve lead

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

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

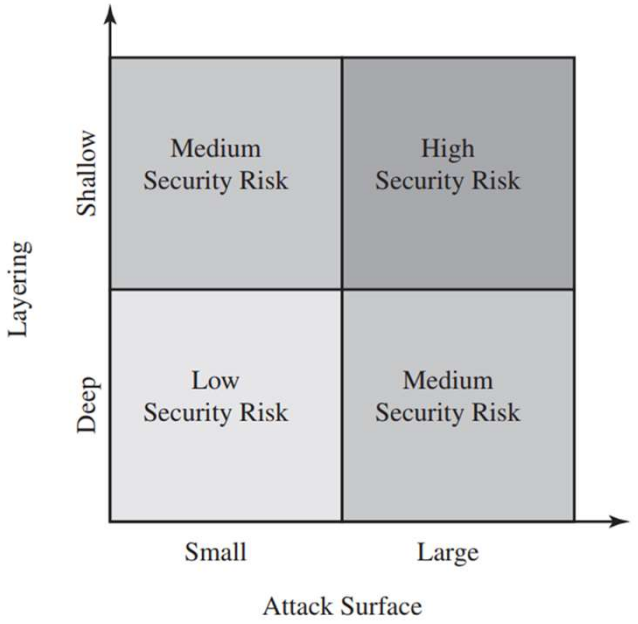
Attack Surfaces

- 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

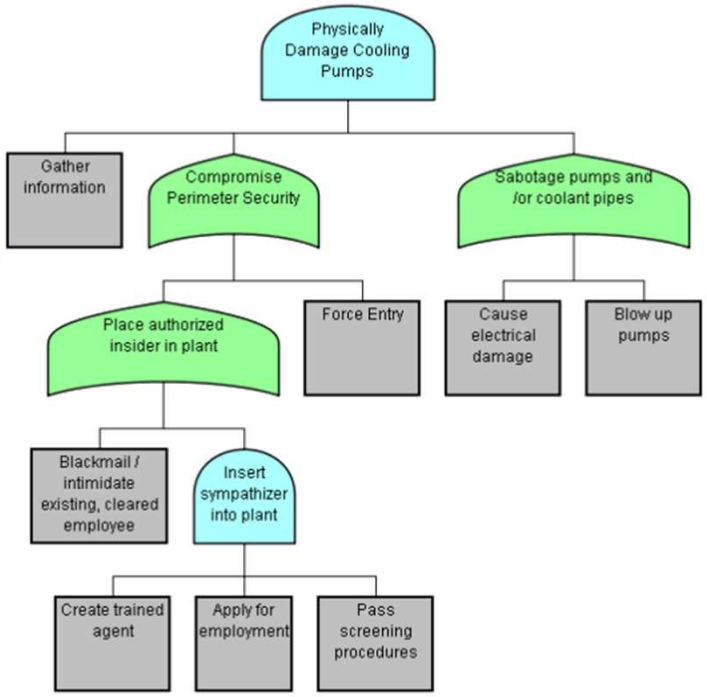
- Is a useful technique for assessing the scale and severity of threats to a system
- A systematic analysis of vulnerable points makes security analysts aware of where security mechanisms are required
- Once an attack surface is defined, designers may be able to find ways to make the surface smaller, 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

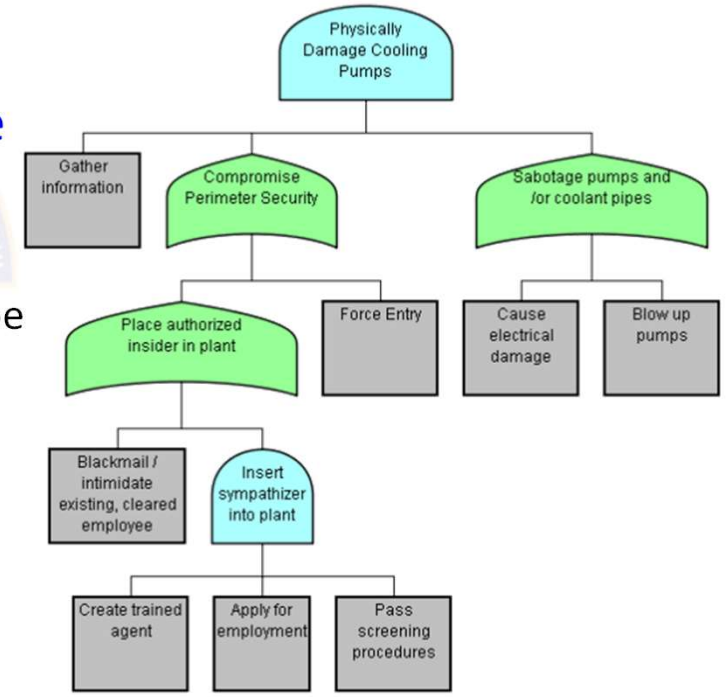
- An attack tree shows a set of potential techniques for exploiting security vulnerabilities
- The goal of the attack (the security incident) is represented as the root node
- Branches and subnodes represent the ways in which the goal can be reached
- Each subnode defines a subgoal
  - Each subgoal may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



Attack Surfaces and Attack Trees

Attack Trees

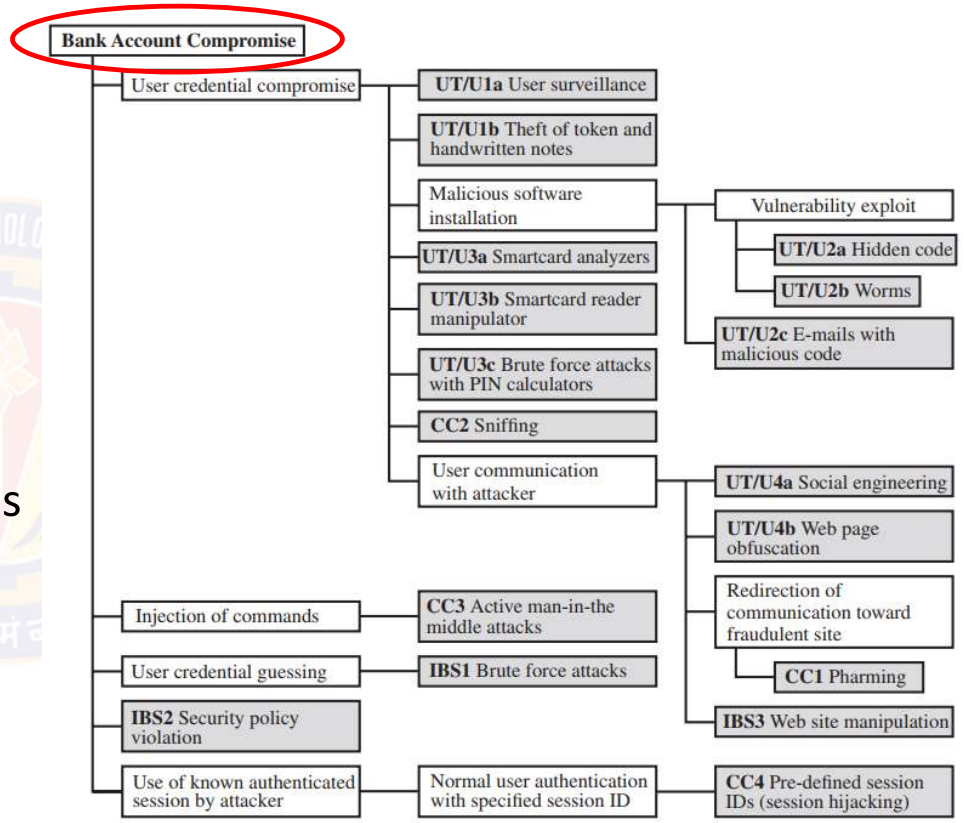
- The leaf nodes represent different ways to initiate an attack
- Each node other than a leaf is either an AND-node or an OR-node
- To achieve the goal represented by an AND-node,
  - all the subgoals represented by that node's subnodes must be achieved
- To achieve the goal represented by an OR-node,
  - 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)

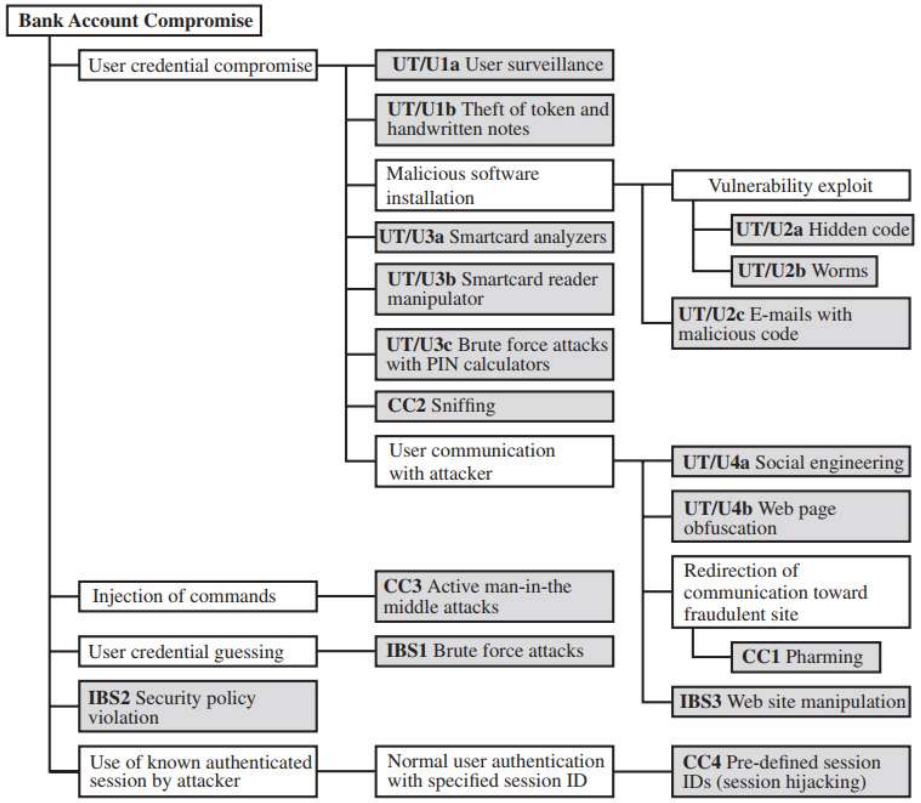


An Attack Tree for Internet Banking Authentication

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

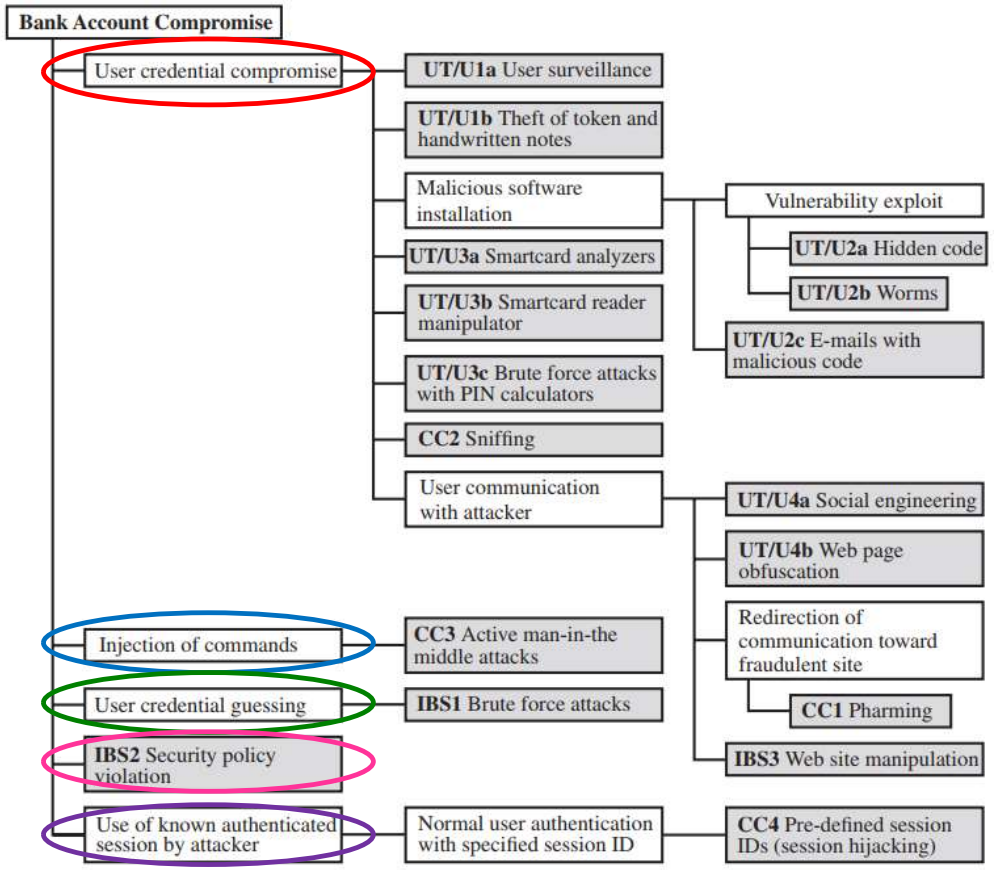


An Attack Tree for Internet Banking Authentication

Attack Surfaces and Attack Trees

Attack Trees – Example

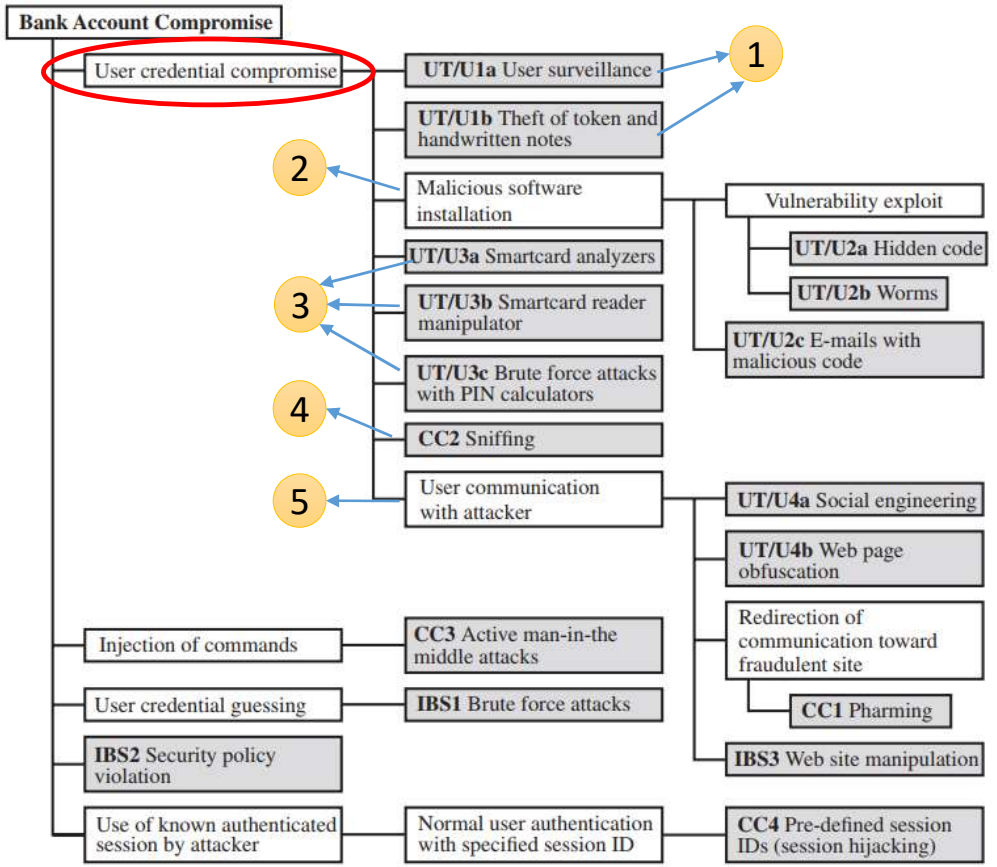
- Attack Strategies
  - Five attack strategies can be identified, each of which exploits one or more of the three components
    - User credential compromise
    - Injection of commands
    - User credential guessing
    - IBS Security policy violation
    - Use of known authenticated session



Attack Surfaces and Attack Trees

Attack Trees – Example

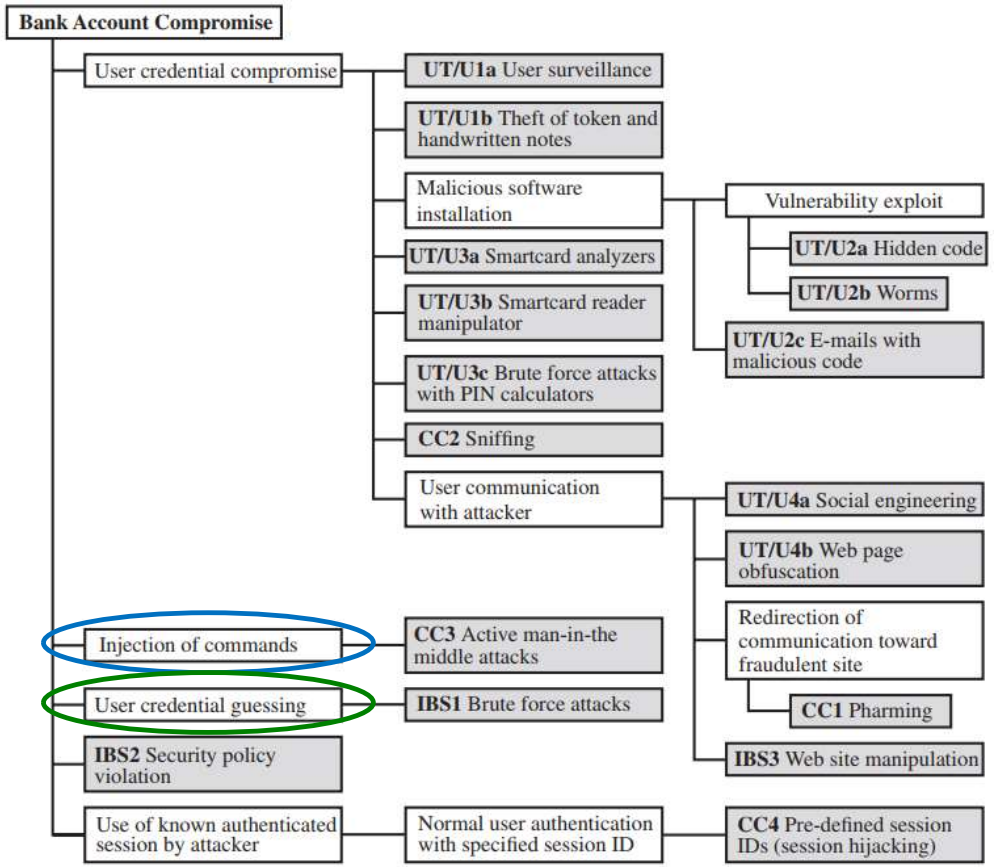
- User credential compromise
  - This strategy can be used against many elements of the attack surface
  - One by using procedural attacks
    - Monitoring a user's action to observe a PIN or other credential
    - Theft of the user's token or handwritten notes
  - Two
    - Embedding malicious software 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

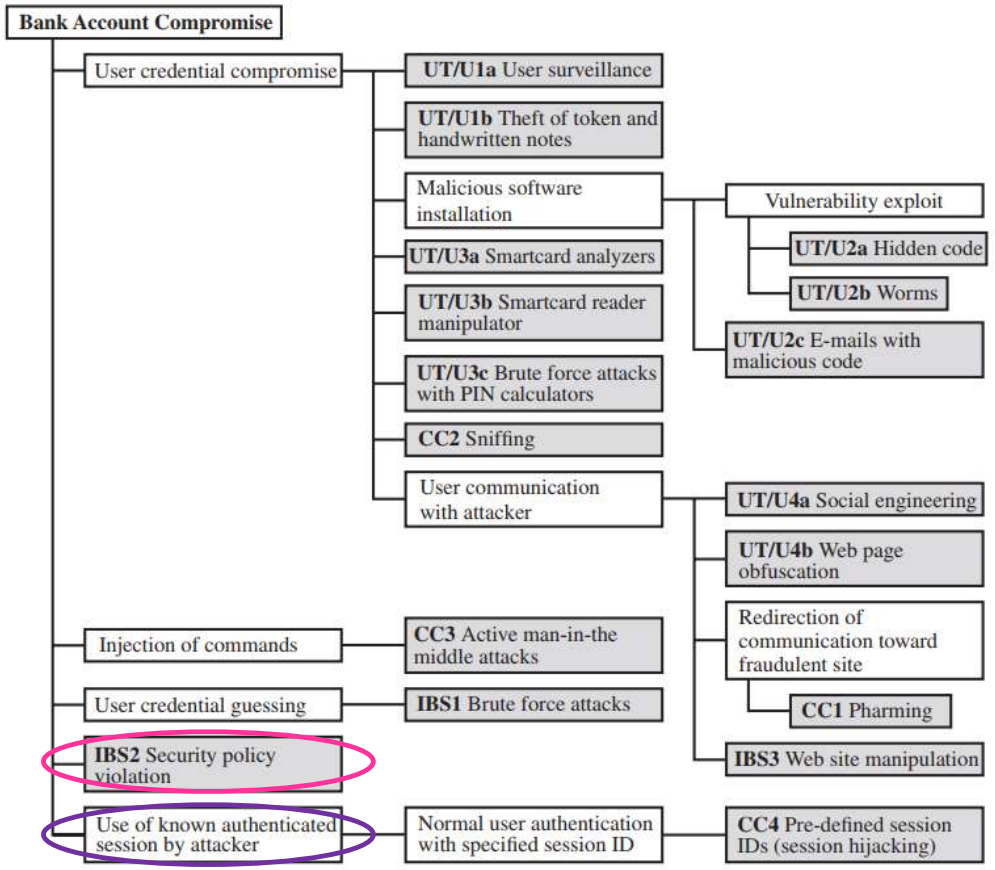
- Injection of commands
  - Involves intercepting communication between the UT and the IBS
  - Involves impersonating the valid user to gain access to the banking system.
- User credential guessing
  - Involves brute force attacks against banking authentication schemes by
    - sending random usernames 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.

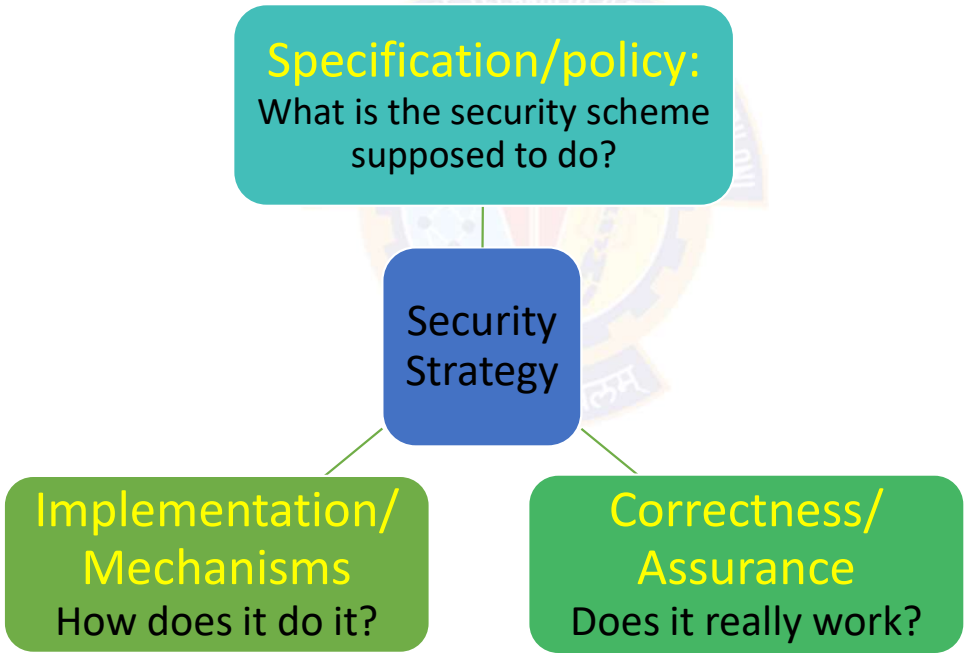
CERT = The Indian Computer Emergency Response Team

Computer Security Strategy

Computer Security Strategy

Comprehensive Security Strategy

- A comprehensive security strategy involves three aspects:



Computer Security Strategy

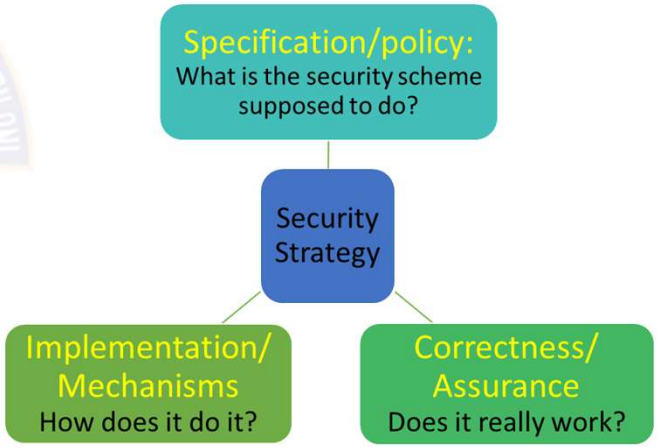
Security Policy

- 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

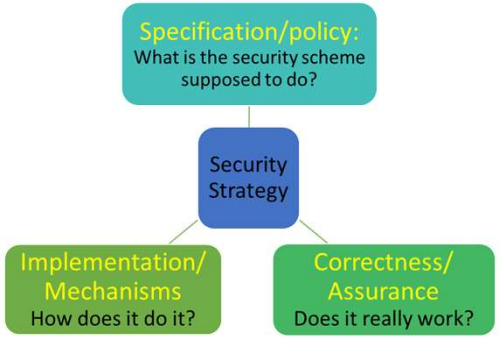
Computer Security Strategy

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



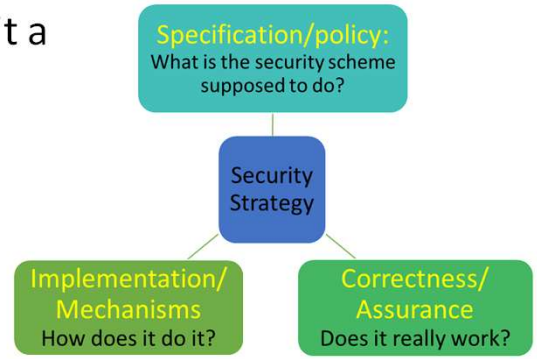
Security Policy – Trade-offs



- 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

Security Policy – Trade-offs

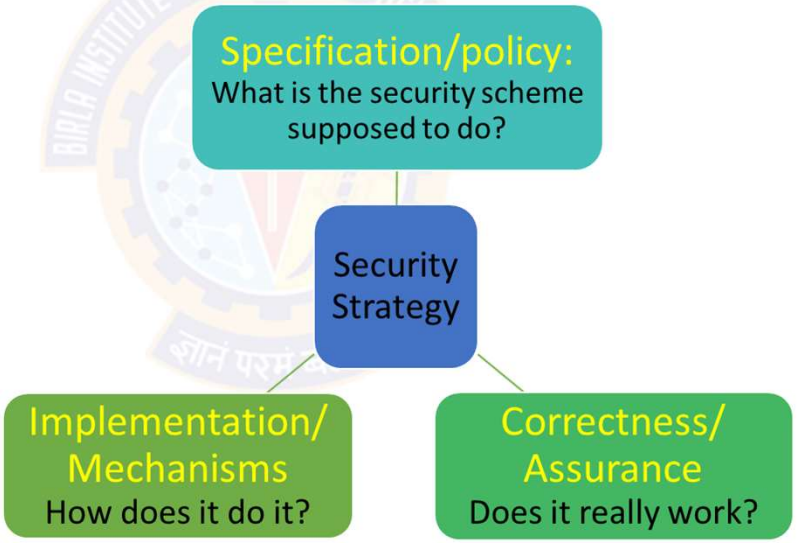
- Cost of security versus cost of failure and recovery
  - Costs of implementing and maintaining security measures must be balanced against the cost of security failure and recovery
  - The cost of security failure and recovery must take into account:
    - the value of the assets being protected and 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

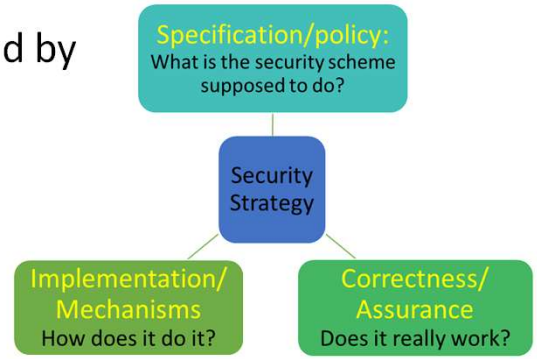
Security Implementation

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



Security Implementation

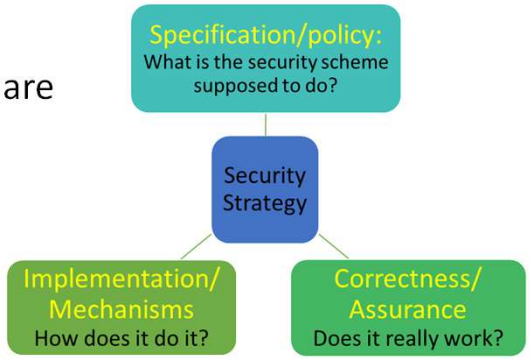
- Prevention
  - An ideal security scheme is one in which no attack is successful, which is impractical
  - There is a wide range of threats in which prevention is a reasonable goal
  - Example: Transmission of encrypted data
    - Attacks on confidentiality of the transmitted data can be prevented by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



Computer Security Strategy

Security Implementation

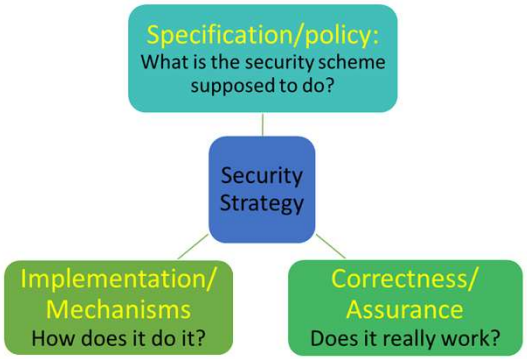
- Detection
  - Absolute prevention is not feasible, but it is practical to detect security attacks
  - For example:
    - Detecting the presence of unauthorized individuals logged into a system using intrusion detection systems
    - Detecting a denial of service attack
      - Communications or processing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



Computer Security Strategy

Security Implementation

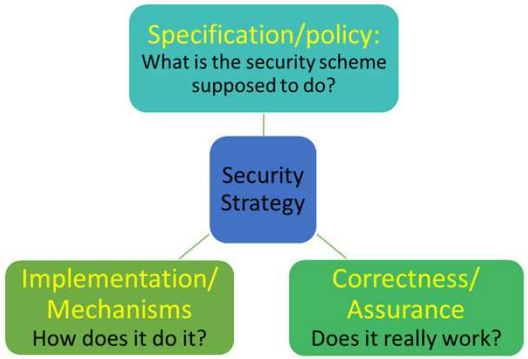
- Response:
  - Once an attack (E.g., denial of service) is detected, the system can respond by halting the attack and preventing further damage
- Recovery:
  - Assets (E.g., data) can be recovered using backup systems
  - If data integrity is compromised, a prior, correct copy of the data can be reloaded



Computer Security Strategy

Assurance and Evaluation

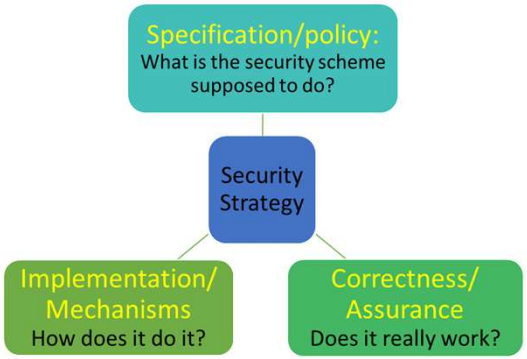
- The "consumers" of computer security services and mechanisms (e.g., system managers, vendors, customers, and end users) want to feel that the security measures work as intended
- This bring us to the concepts of:
  - Assurance and Evaluation.



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."*

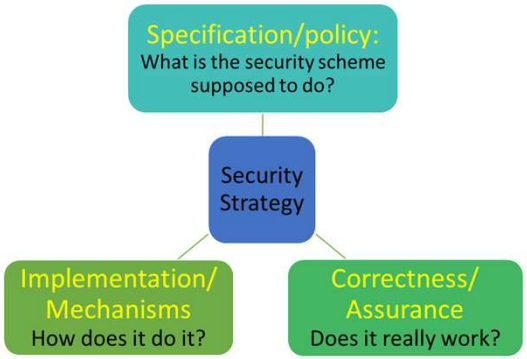
    -- NIST95
  - This encompasses both system design and system implementation
  - Assurance deals with the questions such as:
    - "Does the security system design meet its requirements?"
    - "Does the security system implementation meet its specifications?"
  - Note:
    - Assurance is expressed as 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

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

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

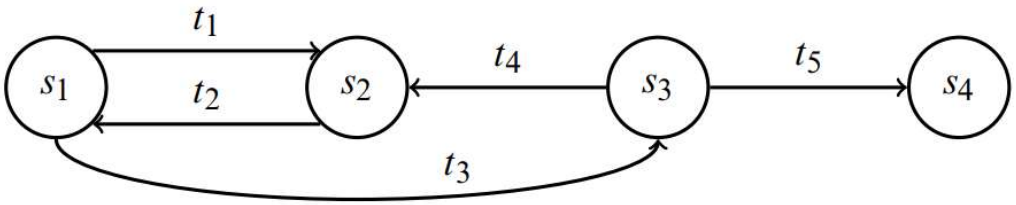
Overview

- 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

innovate achieve lead

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

Types of Security Policies

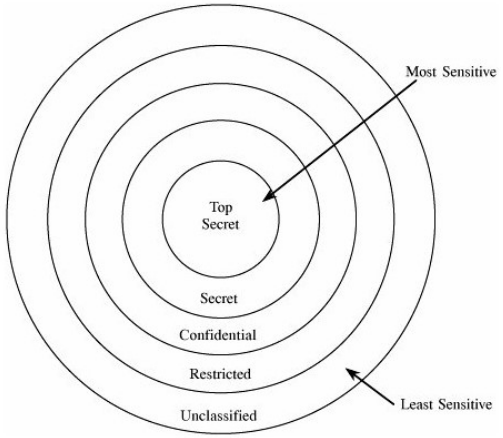
Policy Model

- 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



Hierarchy of Sensitivities.

- 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

Image Source: https://flylib.com

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

innovate achieve lead

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

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?

Maim = a short statement of a general truth, principle, or rule for behaviour

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 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

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

innovate achieve lead

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

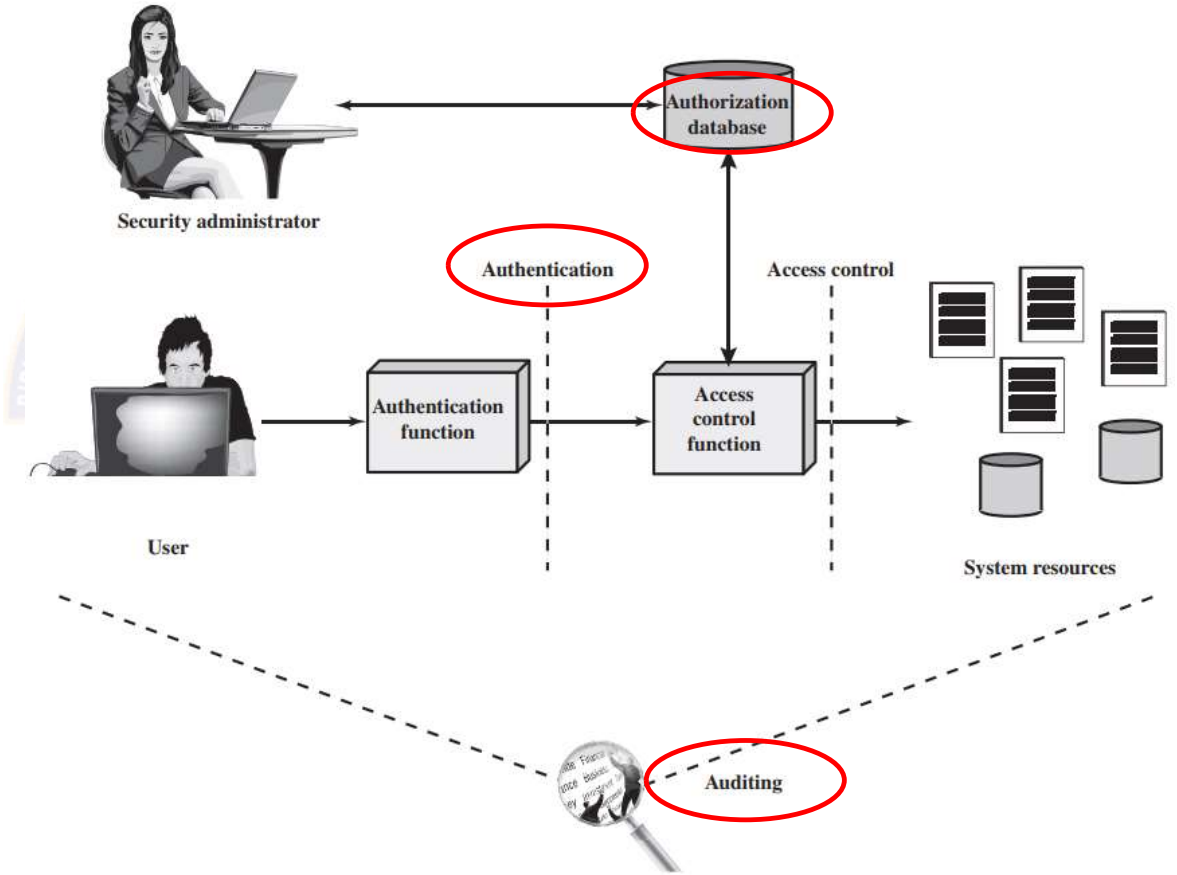
Overview

- 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

innovate achieve lead

Context

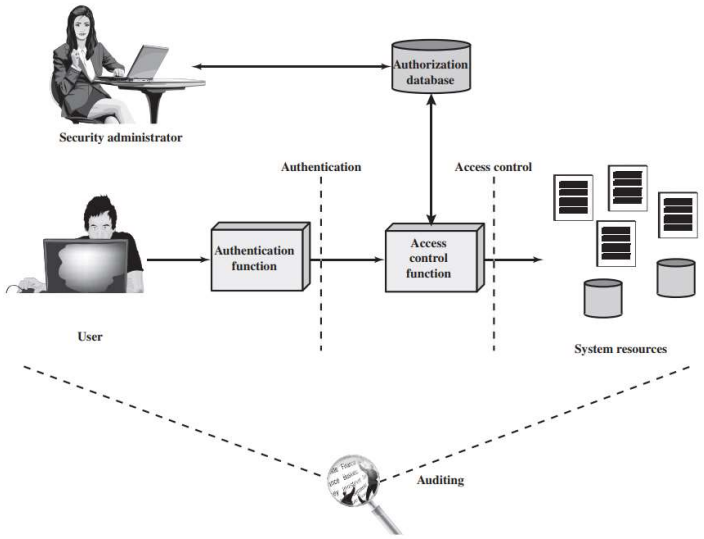
- Authentication:
  - Verification that the credentials of a user or other system entity are valid.
- Authorization:
  - The granting of a right or permission to a system entity to access a system resource
  - This function determines who is trusted for a given purpose.
- Audit:
  - An independent review 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

innovate achieve lead

Context

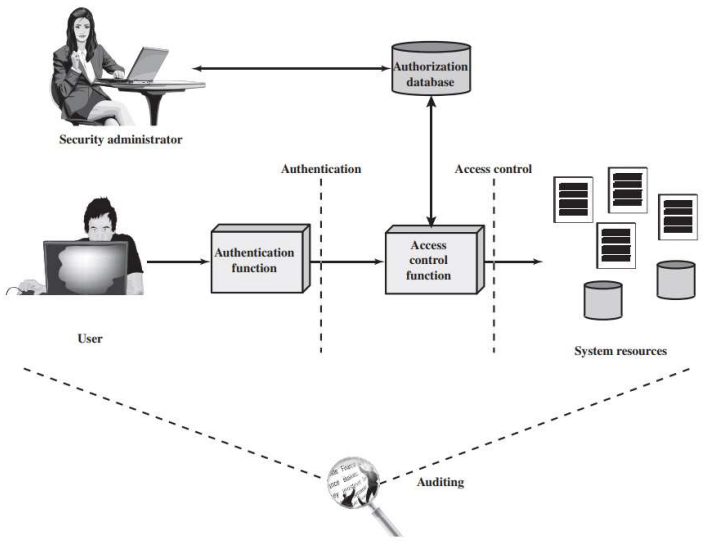
- An access control mechanism mediates between a subject and objects
- The system must first authenticate an entity seeking access
- Typically, the authentication function determines whether the user is permitted to access the system at all
- Then the access control function determines if the specific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

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

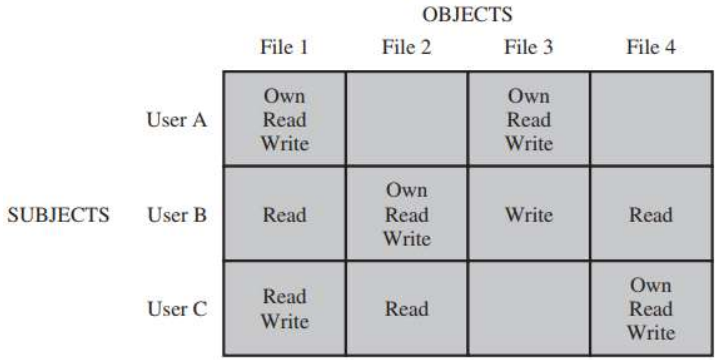
DAC/IBAC Model – Access Control Lists

- 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

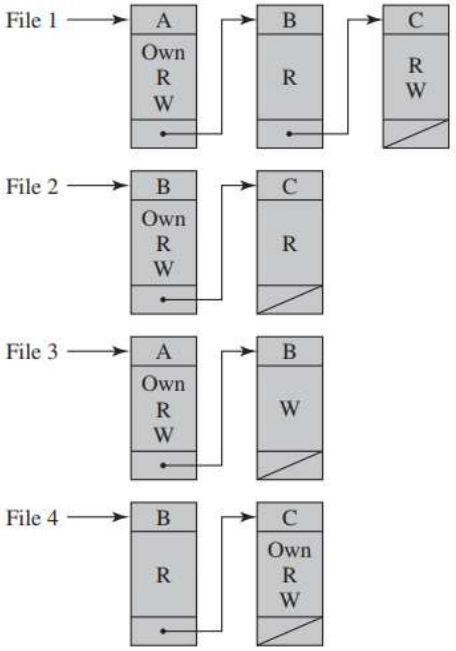
innovate achieve lead

DAC/IBAC Model – Access Control Lists

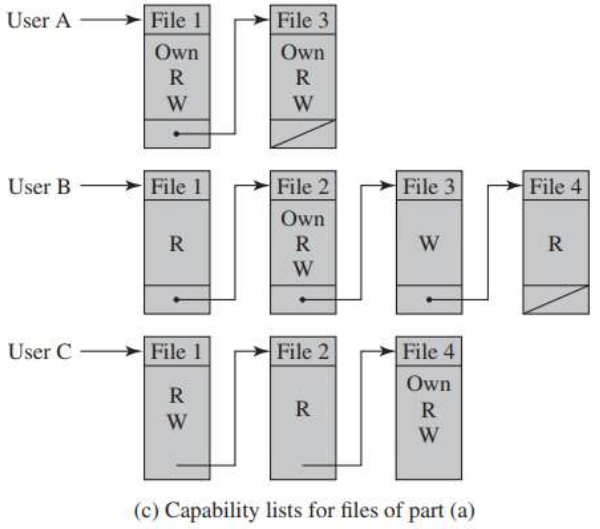


SUBJECTS	OBJECTS: File 1	File 2	File 3	File 4
User A	Own Read Write		Own Read Write	
User B	Read	Own Read Write	Write	Read
User C	Read Write	Read		Own Read Write

(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

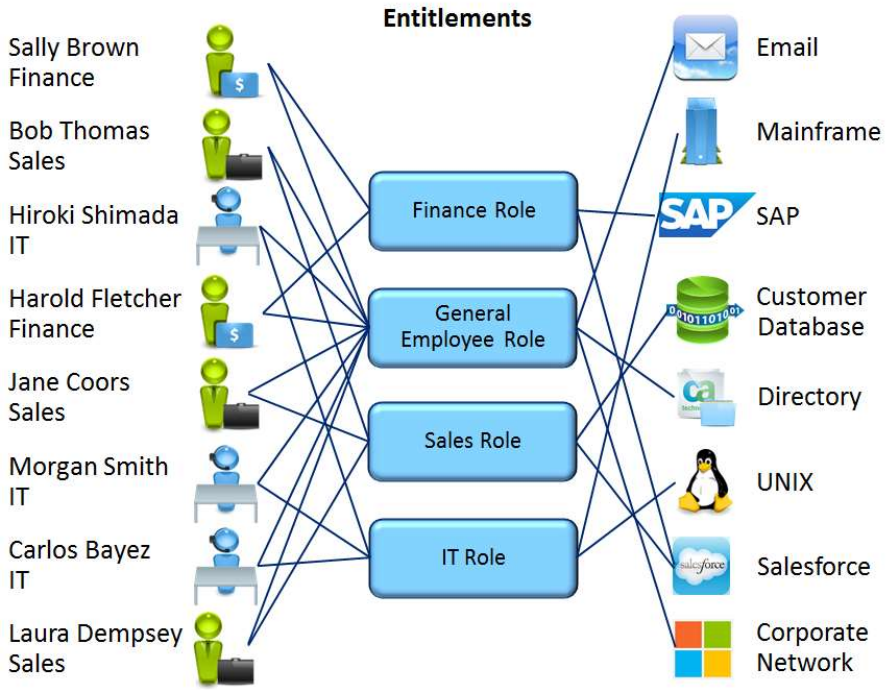
MAC/RAC

- 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

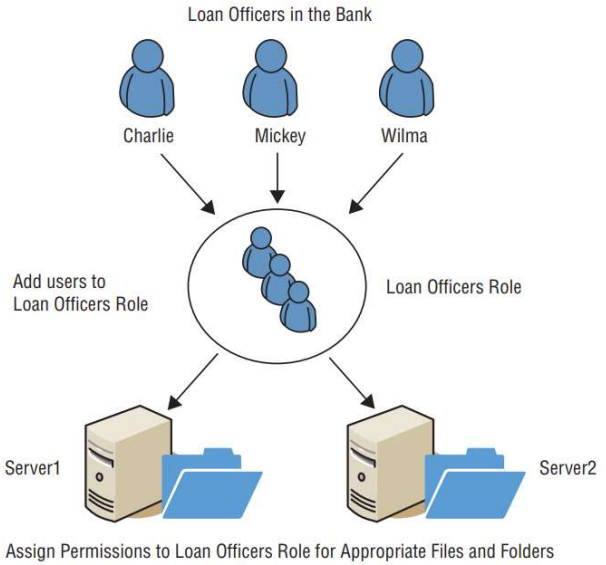


Types of Access Control

innovate achieve lead

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 to 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)

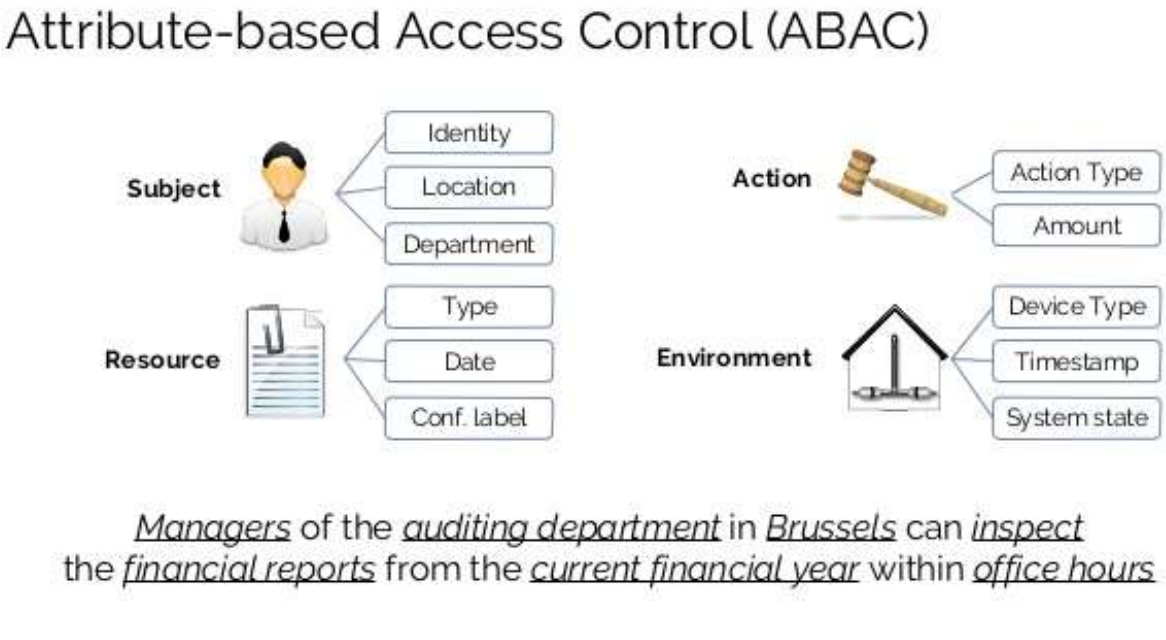
- Rule-based access control models include global rules that apply to all subjects equally
- An advanced implementation of a rule-based access control is an Attribute Based Access Control (ABAC) model
- ABAC models use policies that include multiple attributes for rules
  - Attributes are characteristics 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

innovate achieve lead

Attribute Based Access Controls (ABAC)



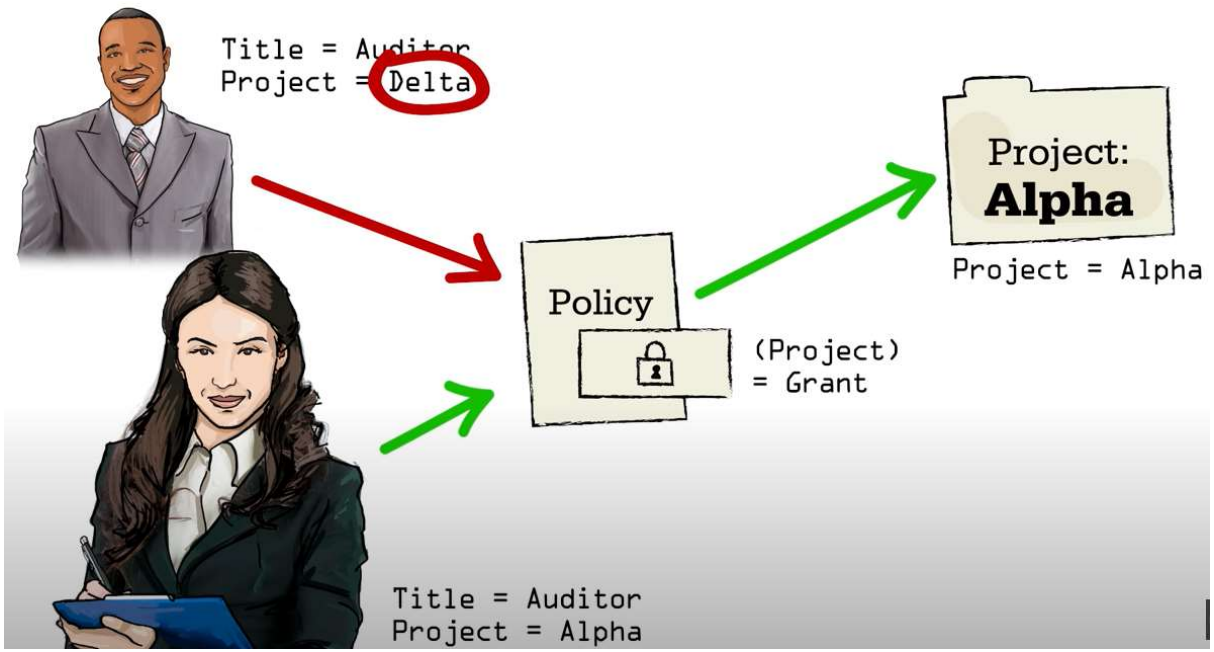
- Subject
  - Managers
  - Auditing Department
  - Brussels
- Action
  - Inspect
- Resource
  - Financial reports
  - Financial year
- Environment
  - Current
  - Office Hours

Image Source: https://www.slideshare.net/elimity/iam-innovation-tech-startup-experiences-a-deeper-look-into-rbac-vs-abac

Types of Access Control

innovate achieve lead

Attribute Based Access Controls (ABAC)



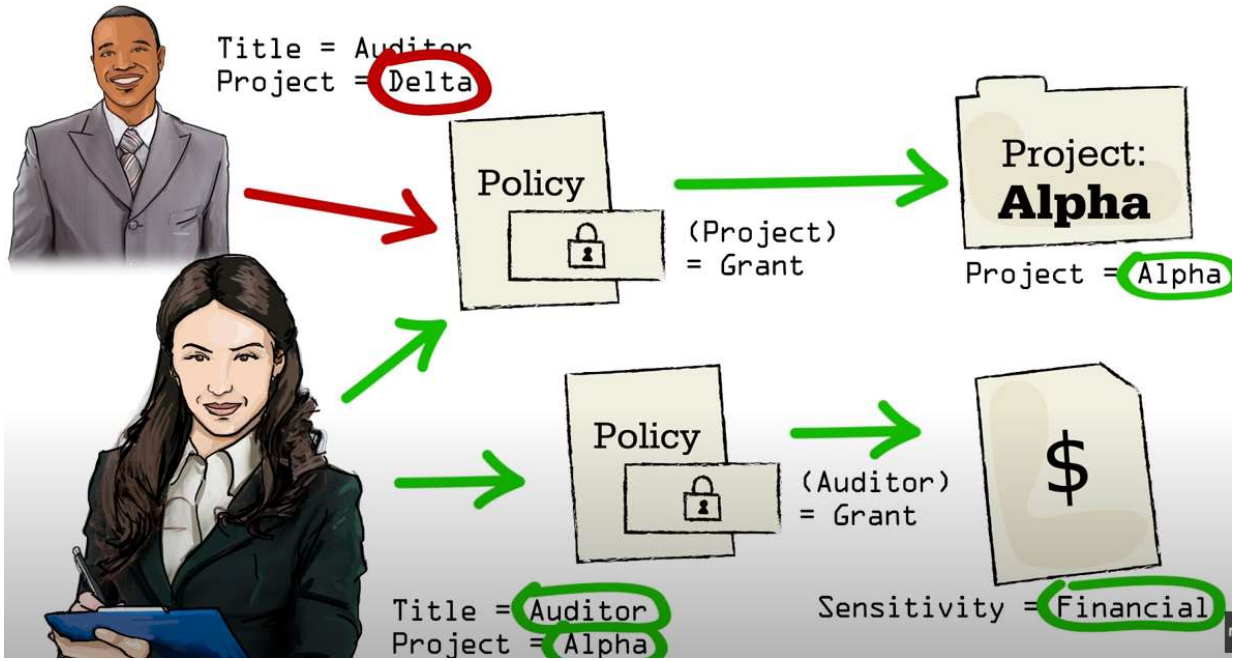
- Subject
  - Auditor
- Action
  - Read, Write
- Resource
  - Financial reports
  - Project = Alpha
  - Project = Delta
- Environment
  - Project Duration

Image Source: YouTube: https://youtu.be/cgTa7YnGfHA

Types of Access Control

innovate achieve lead

Attribute Based Access Controls (ABAC)



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

Image Source: YouTube: https://youtu.be/cgTa7YnGfHA

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!