

# CS472 - PRINCIPLES OF INFORMATION SECURITY

## Module – I

Introduction: Overview of computer security, Security concepts, Need of Security- Threats- Deliberate software attacks, Deviation in quality of service, Attacks- malicious code, brute force, Timing attack, sniffers. Access Control Mechanisms - Access Control, Access control matrix, Access control in OS-Discretionary and Mandatory access control, Role-based access control, case study SELinux

### Overview of computer security

#### What is Security?

- The protection of information and its critical elements, including systems and hardware that use, store, and transmit that information
- Necessary tools: policy, awareness, training, education, technology
- Information Security components are
  - **Confidentiality**
  - **Integrity**
  - **Availability**

#### *Confidentiality*

The principle of confidentiality specifies that only the sender and the intended recipient should be able to access the content of the message.



Confidentiality of information ensures that only those with sufficient privileges may access certain information. When unauthorized individuals or systems can access information, confidentiality is breached. To protect the confidentiality of information, a number of measures are used:

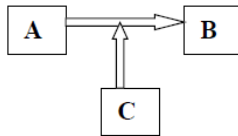
- Information classification
- Secure document storage
- Application of general security policies
- Education of information custodians and end users

Example, a credit card transaction on the Internet.

- The system attempts to enforce confidentiality by encrypting the card number during transmission, by limiting the places where it might appear (in data bases, log files, backups, printed receipts, and so on), and by restricting access to the places where it is stored.
- Giving out confidential information over the telephone is a breach of confidentiality if the caller is not authorized to have the information, it could result in a breach of confidentiality.

#### *Integrity*

Integrity means that data cannot be modified without authorization. The confidential information sent by A to B which is accessed by C without the permission or knowledge of A and B.



Integrity is the quality or state of being whole, complete, and uncorrupted. The integrity of information is threatened when it is exposed to corruption, damage, destruction, or other disruption of its authentic state. Corruption can occur while information is being compiled, stored, or transmitted.

- Integrity means that data cannot be modified without authorization.
- Eg: Integrity is violated when an employee deletes important data files, when a computer virus infects a computer, when an employee is able to modify his own salary in a payroll database, when an unauthorized user vandalizes a website, when someone is able to cast a very large number of votes in an online poll, and so on.

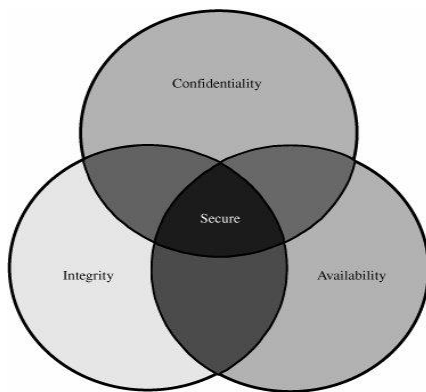
### **Availability**

Availability: It means that assets are accessible to authorized parties at appropriate times.

Availability is the characteristic of information that enables user access to information without interference or obstruction and in a required format. A user in this definition may be either a person or another computer system. Availability does not imply that the information is accessible to any user; rather, it means availability to authorized users.

- For any information system to serve its purpose, the information must be available when it is needed.
- Eg: High availability systems aim to remain available at all times, preventing service disruptions due to power outages, hardware failures, and system upgrades.

### **CIA Triangle**

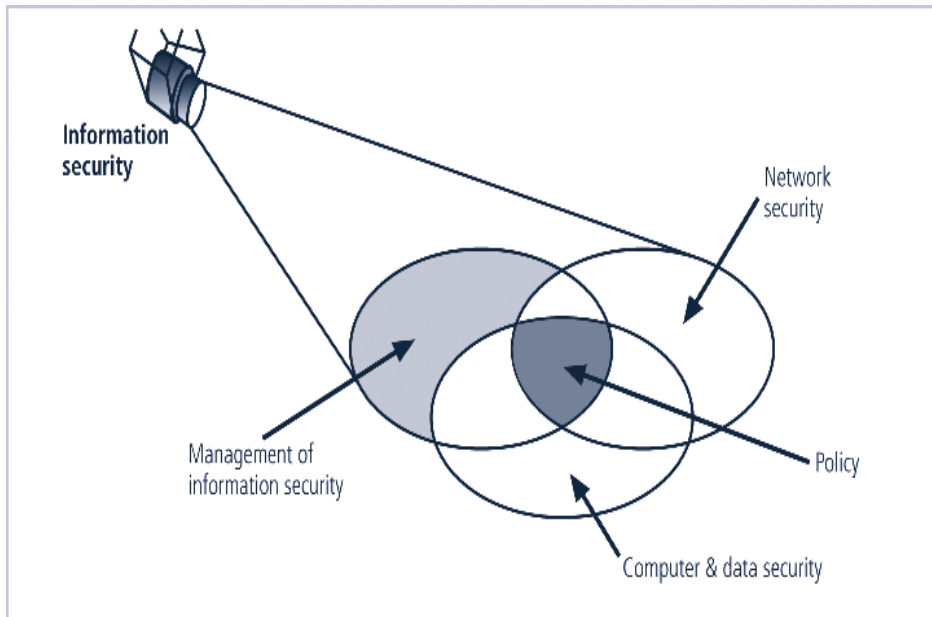


The C.I.A. triangle - confidentiality, integrity, and availability - has expanded into a more comprehensive list of critical characteristics of information.

At the heart of the study of information security is the concept of policy. Policy, awareness, training, education, and technology are vital concepts for the protection of information and for keeping information systems from danger.

- C.I.A. triangle now expanded into list of critical characteristics of information
- The value of information comes from the characteristics it possesses:
  - Availability
  - Accuracy

- Authenticity
- Confidentiality
- Integrity
- Utility
- Possession



**FIGURE 1-3** Components of Information Security

### ***Privacy***

The information that is collected, used, and stored by an organization is to be used only for the purposes stated to the data owner at the time it was collected.

This definition of privacy does focus on freedom from observation (the meaning usually associated with the word), but rather means that information will be used only in ways known to the person providing it.

### ***Identification***

An information system possesses the characteristic of identification when it is able to recognize individual users. Identification and authentication are essential to establishing the level of access or authorization that an individual is granted.

### ***Authentication***

Authentication occurs when a control provides proof that a user possesses the identity that he or she claims.

In computing, e-Business and information security it is necessary to ensure that the data, transactions, communications or documents (electronic or physical) are genuine (i.e. they have not been forged or fabricated)

### ***Authorization***

After the identity of a user is authenticated, a process called authorization provides assurance that the user (whether a person or a computer) has been specifically and explicitly authorized by the proper authority to access, update, or delete the contents of an information asset.

## ***Accountability***

The characteristic of accountability exists when a control provides assurance that every activity undertaken can be attributed to a named person or automated process. For example, audit logs that track user activity on an information system provide accountability.

## ***Accuracy***

Information should have accuracy. Information has accuracy when it is free from mistakes or errors and it has the value that the end users expects. If information contains a value different from the user's expectations, due to the intentional or unintentional modification of its content, it is no longer accurate.

## ***Utility***

Information has value when it serves a particular purpose. This means that if information is available, but not in a format meaningful to the end user, it is not useful. Thus, the value of information depends on its utility.

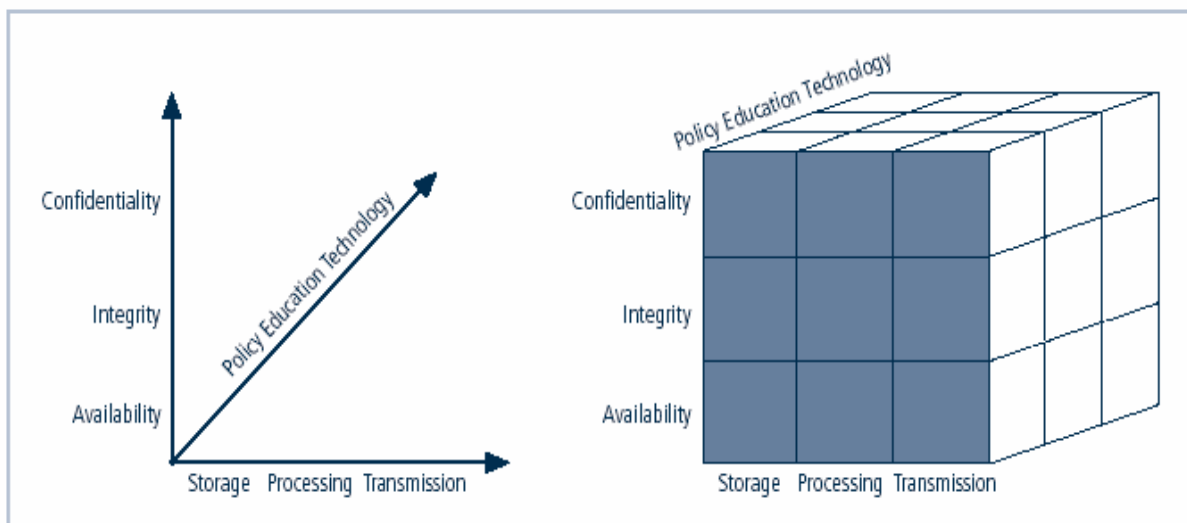
## ***Possession***

The possession of Information security is the quality or state of having ownership or control of some object or item.

## **Security Concepts**

### **NSTISSC Security Model**

**'National Security Telecommunications & Information systems security committee' document.**



**FIGURE 1-4** NSTISSC Security Model

- It is now called the National Training Standard for Information security professionals.
- The NSTISSC Security Model provides a more detailed perspective on security.
- While the NSTISSC model covers the three dimensions of information security, it omits discussion of detailed guidelines and policies that direct the implementation of controls.
- The 3 dimensions of each axis become a 3x3x3 cube with 27 cells representing areas that must be addressed to secure today's Information systems.

- To ensure system security, each of the 27 cells must be properly addressed during the security process.
- For ex, the intersection between technology, Integrity & storage areas requires a control or safeguard that addresses the need to use technology to protect the Integrity of information while in storage.
- weakness of using this model with too limited an approach is to view it from a single perspective.

### **Components of an Information System**

- |            |              |            |
|------------|--------------|------------|
| - Software | - Hardware   | - Data     |
| - People   | - Procedures | - Networks |

#### **Software**

The software components of IS comprises applications, operating systems, and assorted command utilities. Software programs are the vessels that carry the lifeblood of information through an organization. These are often created under the demanding constraints of project management, which limit time, cost, and manpower.

#### **Hardware**

Hardware is the physical technology that houses and executes the software, stores and carries the data, and provides interfaces for the entry and removal of information from the system.

Physical security policies deal with hardware as a physical asset and with the protection of these physical assets from harm or theft. Applying the traditional tools of physical security, such as locks and keys, restricts access to and interaction with the hardware components of an information system.

Securing the physical location of computers and the computers themselves is important because a breach of physical security can result in a loss of information. Unfortunately, most information systems are built on hardware platforms that cannot guarantee any level of information security if unrestricted access to the hardware is possible.

#### **Data**

- Data stored, processed, and transmitted through a computer system must be protected.
- Data is often the most valuable asset possessed by an organization and is the main target of intentional attacks.
- The raw, unorganized, discrete (separate, isolated) potentially-useful facts and figures that are later processed(manipulated) to produce information.

#### **People**

There are many roles for people in information systems. Common ones include

- Systems Analyst
- Programmer
- Technician
- Engineer
- Network Manager
- MIS (Manager of Information Systems)
- Data entry operator

## **Procedures**

A procedure is a series of documented actions taken to achieve something. A procedure is more than a single simple task. A procedure can be quite complex and involved, such as performing a backup, shutting down a system, patching software.

## **Networks**

- When information systems are connected to each other to form Local Area Network (LANs), and these LANs are connected to other networks such as the Internet, new security challenges rapidly emerge.
- Steps to provide network security are essential, as is the implementation of alarm and intrusion systems to make system owners aware of ongoing compromises.

## **Need of Security**

Information security is unlike any other aspect of information technology. It is an arena where the primary mission is to ensure things stay the way they are.

If there were no threats to information and systems, we could focus on improving systems that support the information, resulting in vast improvements in ease of use and usefulness.

The first phase, Investigation, provides an overview of the environment in which security must operate, and the problems that security must address.

## **BUSINESS NEEDS FIRST, TECHNOLOGY NEEDS LAST**

Information security performs four important functions for an organization:

1. Protects the organization's ability to function
2. Enables the safe operation of applications implemented on the organization's IT systems
3. Protects the data the organization collects and uses
4. Safeguards the technology assets in use at the organization

### **1. Protecting the Ability of the Organization to Function**

Both general management and IR management are responsible for implementing information security to protect the ability of the organization to function.

"information security is a management issue in addition to a technical issue, it is a people issue in addition to the technical issue."

To assist management in addressing the needs for information security, communities of interest must communicate in terms of business impact and the cost of business interruption and avoid arguments expressed only in technical terms.

### **2. Enabling the Safe Operation of Applications**

Today's organizations are under immense pressure to create and operate integrated, efficient, and capable applications.

The modern organization needs to create an environment that safeguards applications using the organization's IT systems, particularly the environment of the organization's infrastructure.

Once the infrastructure is in place, management must understand it has not abdicated to the IT department its responsibility to make choices and enforce decisions, but must continue to oversee the infrastructure.

### **3.Protecting Data Organizations Collect and Use**

Many organizations realize that one of their most valuable assets is their data, because without data, an organization loses its record of transactions and/or its ability to deliver value to its customers.

Protecting data in motion and data at rest are both critical aspects of information security.

An effective information security program is essential to the protection of the integrity and value of the organization's data.

### **4.Safeguarding the Technology Assets in Organizations**

To perform effectively, organizations must add secure infrastructure services based on the size and scope of the enterprise.

When an organization grows and more capabilities are needed, additional security services may have to be provided locally.

Likewise, as the organization's network grows to accommodate changing needs, more robust technology solutions may be needed to replace security programs the organization has outgrown.

## **Threats**

To protect an organization's information, you must

### **1. Know yourself**

(i.e) be familiar with the information to be protected, and the systems that store, transport and process it.

### **2. Know the threats you face**

To make sound decisions about information security, management must be informed about the various threats facing the organization, its application, data and information systems.

- A threat is an object, person, or other entity, that represents a constant danger to an asset.
- By examining each threat category in turn, management effectively protects its information through **policy, education and training, and technology controls**

### **Threats to Information Security**

<b>Categories of threat</b>		<b>Examples</b>
Acts of human error or failure	--	Accidents, employee mistakes
Compromises to intellectual property	--	Piracy, copyright infringement
Deliberate acts of espionage or trespass--		Unauthorized access and/or/data collection
Deliberate acts of information extortion--		Blackmail or information disclosure
Deliberate acts of sabotage or vandalism --		Destruction of systems or information
Deliberate acts of theft	--	Illegal confiscation of equipment or information

Deliberate software attacks	--	Viruses, worms, macros, denial-of-service
Forces of nature	--	Fire, flood, earthquake, lightning
Deviations in quality of service	--	ISP, power, or WAN service providers
Technical hardware failures or errors	--	Equipment failure
Technical software failures or errors	--	Bugs, code problems, unknown loopholes
Technological obsolescence	--	Antiquated or outdated technologies

### 1. Acts of Human Error or Failure:

- Acts performed without intent or malicious purpose by an authorized user.
  - because of in experience, improper training,
  - Making of incorrect assumptions.
- Employees are greatest threats to information security – They are closest to the organizational data
- Employee mistakes can easily lead to the following:
  - revelation of classified data
  - entry of erroneous data
  - accidental deletion or modification of data
  - storage of data in unprotected areas
  - failure to protect information
- Many of these threats can be prevented with
  - Training
  - Ongoing awareness activities
  - Verification by a second party
  - Many military applications have robust, dual- approval controls built in.

### 2. Compromises to Intellectual Property

- is defined as the ownership of ideas and control over the tangible or virtual representation of those ideas.
- Intellectual property includes **trade secrets, copyrights, trademarks, and patents**.
- Once intellectual property has been defined and properly identified, breaches to IP constitute a threat to the security of this information.
- Organization purchases or leases the IP of other organizations.
- Most Common IP breach is the unlawful use or duplication of software based intellectual property more commonly known as **software Piracy**.
- Software Piracy affects the world economy.
- U.S provides approximately 80% of world's software.

In addition to the laws surrounding software piracy, two watch dog organizations investigate allegations of software abuse.



1. Software and Information Industry Association (SIIA)  
(i.e) Software Publishers Association
2. Business Software Alliance (BSA)
  - Another effort to combat (take action against) piracy is the online registration process.

### 3. Deliberate Acts of Espionage or Trespass

- Electronic and human activities that can breach the confidentiality of information.
- When an unauthorized individual's gain access to the information an organization is trying to protect is categorized as act of espionage or trespass.
- Attackers can use many different methods to access the information stored in an information system.
  - Competitive Intelligence [use web browser to get information from market research]
  - Industrial espionage(spying)
  - Shoulder Surfing (ATM)
    - Hackers uses skill, guile, or fraud to steal the property of someone else

#### Trespass

- Can lead to unauthorized real or virtual actions that enable information gatherers to enter premises or systems they have not been authorized to enter.
- Sound principles of authentication & authorization can help organizations protect valuable information and systems.
- **Hackers**-> "People who use and create computer software to gain access to information illegally"
- Generally, two skill levels among hackers:
  - Expert hacker
    - develops software scripts and codes exploits
    - usually a master of many skills
    - will often create attack software and share with others
  - Script kiddies
    - hackers of limited skill
    - use expert-written software to exploit a system
    - do not usually fully understand the systems they hack
- Other terms for system rule breakers:
  - Cracker - an individual who "cracks" or removes protection designed to prevent unauthorized duplication
  - Phreaker - hacks the public telephone network

### 4. Deliberate Acts of information Extortion (obtain by force or threat)

Possibility of an attacker or trusted insider stealing information from a computer system and demanding compensation for its return or for an agreement not to disclose the information.

### 5. Deliberate Acts of sabotage or Vandalism

- Destroy an asset or
- Damage the image of organization

- Cyber terrorism-Cyber terrorists hack systems to conduct terrorist activities through network or internet pathways.

## **6. Deliberate Acts of Theft**

- Illegal taking of another's property-- is a constant problem.
- Within an organization, property can be physical, electronic, or intellectual.
- Physical theft can be controlled by installation of alarm systems.
- Trained security professionals.
- Electronic theft control is under research.

## **7. Deliberate Software Attacks**

- Deliberate software attacks occur when an individual or group designs software to attack an unsuspecting system. Most of this software is referred to as malicious code or malicious software, or sometimes malware.
- These software components or programs are designed to damage, destroy, or deny service to the target systems.
- Some of the more common instances of malicious code are viruses and worms, Trojan horses, logic-bombs, back doors, and denial-of-services attacks.
- "The British Internet Service Provider Cloud nine" be the first business "hacked out of existence"
- Includes:
  - macro virus
  - boot virus
  - worms
  - Trojan horses
  - logic bombs
  - back door or trap door
  - denial-of-service attacks
  - polymorphic
  - hoaxes

### **Virus**

- Virus is a computer program that attaches itself to an executable file or application. It can replicate itself, usually through an executable program attached to an e-mail.
- A program or piece of code that be loaded on to your computer, without your knowledge and run against your wishes.
- The keyword is "attaches". A virus cannot stand on its own. It cannot replicate itself or operate without the presence of a host program. A virus attaches itself to a host program, just as the flu attaches itself to a host organism.
- You must prevent viruses from being installed on computers in your organizations, otherwise, after the virus attaches itself to a program, such as Microsoft Word, it performs whatever its creator designed it to do.

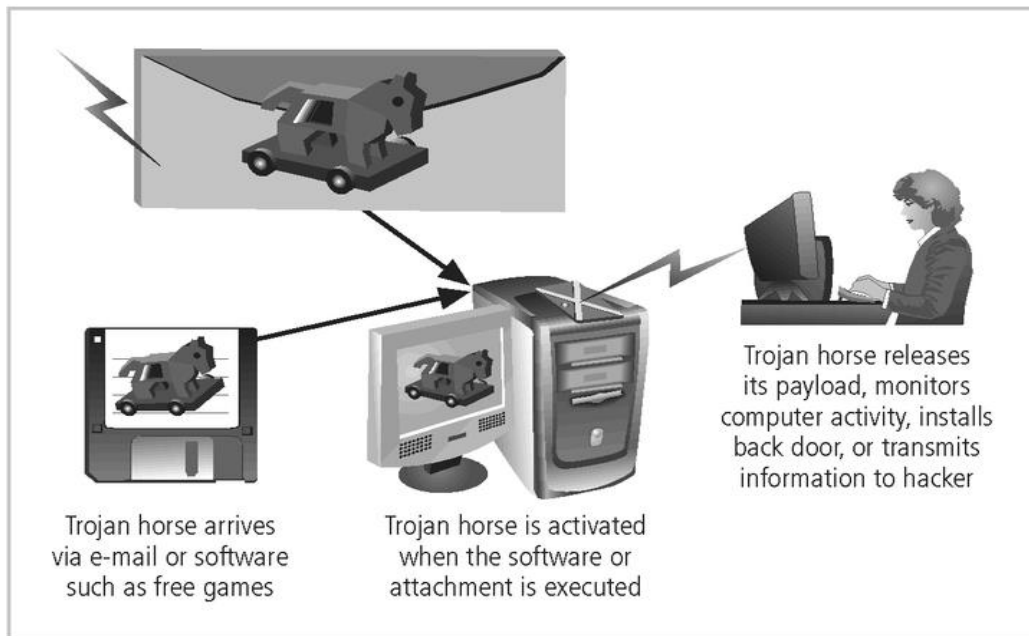
- There is no foolproof method of preventing them from attaching themselves to your computer. Antivirus software compares virus signature files against the programming code of known viruses. Regularly update virus signature files is crucial.
- Segments of code that performs malicious actions.
- Virus transmission is at the opening of Email attachment files.
- **Macro virus**-> Embedded in automatically executing macrocode common in word processors, spreadsheets and database applications.
- **Boot Virus**-> infects the key operating files located in the computer's boot sector.

## Worms

- A worm is a computer program that replicates and propagates itself without having to attach itself to a host.
- Most infamous worms are Code Red and Nimda.
- A worm is a malicious program that replicates itself constantly, without requiring another program to provide a safe environment for replication.
- Worms can continue replicating themselves until they completely fill available resources, such as memory, hard drive space, and network bandwidth.
- Eg: MS-Blaster, MyDoom, Netsky, are multifaceted attack worms.
- Once the worm has infected a computer, it can redistribute itself to all e-mail addresses found on the infected system.
- Furthermore, a worm can deposit copies of itself onto all Web servers that the infected systems can reach, so that users who subsequently visit those sites become infected.
- Cost businesses millions of dollars in damage as a result of lost productivity
- Computer downtime and the time spent recovering lost data, reinstalling programming's, operating systems, and hiring or contracting IT personnel.

## Trojan Horses

- Trojan Programs disguise themselves as useful computer programs or applications and can install a backdoor or rootkit on a computer.
- Backdoors or rootkits are computer programs that give attackers a means of regaining access to the attacked computer later. A rootkit is created after an attack and usually hides itself within the OS tools, so it's almost impossible to detect.
- Are software programs that hide their true nature and reveal their designed behavior only when activated.
- **Types of Trojans**
  - Data Sending Trojans
  - Proxy Trojans
  - FTP Trojans
  - Security software disabler Trojans
  - Denial of service attack Trojans (DOS)



**FIGURE 2-8** Trojan Horse Attack

■ Challenges:

- Trojan programs that use common ports, such as TCP 80, or UDP 53, are more difficult to detect.
- Many software firewalls can recognize port-scanning program or information leaving a questionable port.
- However, they prompt user to allow or disallow, and users are not aware.
- Educate your network users.
- Many Trojan programs use standard ports to conduct their exploits.
- Firewall would most likely identify traffic that's using unfamiliar ports, but Trojan programs that use common ports, such as TCP 80 (HTTP), or UDP 53 (DNS), are more difficult to detect.
- Many software firewalls (Zone Alarm, BlackIce, and McAfee Desktop) can recognize port-scanning program or information leaving a questionable port.

**Spyware**

- A Spyware program sends info from the infected computer to the person who initiated the spyware program on your computer
- Spyware program can register each keystroke entered.
- [www.spywareguide.com](http://www.spywareguide.com)

**Adware**

- Main purpose is to determine a user's purchasing habits so that Web browsers can display advertisements tailored to that user.
- Slow down the computer it's running on.
- Adware sometimes displays a banner that notifies the user of its presence

- Both programs can be installed without the user being aware of their presence

## Back Door or Trap Door

- A Virus or Worm has a payload that installs a backdoor or trapdoor component in a system, which allows the attacker to access the system at will with special privileges.  
Eg: Back Orifice

## Polymorphism

- A **Polymorphic threat** is one that changes its apparent shape over time, making it undetectable by techniques that look for preconfigured signatures.
- These viruses and Worms actually evolve, changing their size, and appearance to elude detection by antivirus software programs.

## Hoaxes

A more devious(tricky) approach to attacking computer systems is the transmission of a virus hoax, with a real virus attached

## Denial-of-service (DoS)

- attacker sends a large number of connection or information requests to a target
- so many requests are made that the target system cannot handle them successfully along with other, legitimate requests for service
- may result in a system crash, or merely an inability to perform ordinary functions

## Distributed Denial-of-service (DDoS)

- an attack in which a coordinated stream of requests is launched against a target from many locations at the same time

## Protecting against Deliberate Software Attacks

- Educating Your Users
  - Many U.S. government organizations make security awareness programs mandatory, and many private-sector companies are following their example.
  - Email monthly security updates to all employees.
  - Update virus signature files as soon as possible.
  - Protect a network by implementing a firewall.

## 8. Forces of Nature

- Forces of nature, *force majeure*, or acts of God are dangerous because they are unexpected and can occur with very little warning
- Can disrupt not only the lives of individuals, but also the storage, transmission, and use of information
- **Include fire, flood, earthquake, and lightning as well as volcanic eruption and insect infestation**

**Fire:** Structural fire that damages the building. Also encompasses smoke damage from a fire or water damage from sprinkles systems.

**Flood:** Can sometimes be mitigated with flood insurance and/or business interruption Insurance.

**Earthquake:** Can sometimes be mitigated with specific causality insurance and/or business interruption insurance, but is usually a separate policy.

**Lightning:** An Abrupt, discontinuous natural electric discharge in the atmosphere.

**Landslide/Mudslide:** The downward sliding of a mass of earth & rocks directly damaging all parts of the information systems.

Since it is not possible to avoid force of nature threats, organizations must implement controls to limit damage.

- They must also prepare contingency plans for continued operations, such as disaster recovery plans, business continuity plans, and incident response plans, to limit losses in the face of these threats.

### **9.Deviations in Quality of Service by Service Providers**

- A product or service is not delivered to the organization as expected.
- The Organization's information system depends on the successful operation of many interdependent support systems.
- It includes power grids, telecom networks, parts suppliers, service vendors, and even the janitorial staff & garbage haulers.
- This degradation of service is a form of **availability disruption**.
- Three sets of service issues that dramatically affect the availability of information and systems are
  - Internet service
  - Communications
  - Power irregularities

#### **Internet Service Issues**

- Internet service Provider (ISP) failures can considerably undermine the availability of information.
- The web hosting services are usually arranged with an agreement providing minimum service levels known as a **Service level Agreement (SLA)**.
- When a Service Provider fails to meet SLA, the provider may accrue fines to cover losses incurred by the client, but these payments seldom cover the losses generated by the outage.

#### **Communications & Other Service Provider Issues**

- Other utility services can affect the organizations are telephone, water, waste water, trash pickup, cable television, natural or propane gas, and custodial services.
- The loss of these services can impair the ability of an organization to function.
- For an example, if the waste water system fails, an organization might be prevented from allowing employees into the building.
- This would stop normal business operations.

### **Power Irregularities**

- The threat of irregularities from power utilities are common and can lead to fluctuations such as
  - Fluctuations due to power excesses.
  - Power shortages &
  - Power losses
- This can pose problems for organizations that provide inadequately conditioned power for their information systems equipment.
- In the U.S., buildings are “fed” 120-volt, 60-cycle power usually through 15- and 20-amp circuits.
- Voltage levels can:
  - **spike** - experience a momentary increase or
  - **surge** - experience prolonged increase,
  - **sag** – momentary low voltage,
  - **brownout** – prolonged drop,
  - **fault** – momentary loss of power,
  - **blackout** – prolonged loss
- the extra voltage can severely damage or destroy equipment.
- Since sensitive electronic equipment, especially networking equipment, computers, and computer-based systems are susceptible to fluctuations, controls can be applied to manage power quality.
- The more expensive uninterruptible power supply (UPS) can protect against spikes and surges.

### **10. Technical Hardware Failures or Errors**

- Technical hardware failures or errors occur when a manufacturer distributes to user’s equipment containing flaws
- These defects can cause the system to perform outside of expected parameters, resulting in unreliable service or lack of availability
- Some errors are terminal, in that they result in the unrecoverable loss of the equipment
- Some errors are intermittent, in that they only periodically manifest themselves, resulting in faults that are not easily repeated

### **11. Technical software failures or errors**

- This category involves threats that come from purchasing software with unknown, hidden faults.
- Large quantities of computer code are written, debugged, published, and sold only to determine that not all bugs were resolved
- Sometimes, unique combinations of certain software and hardware reveal new bugs
- Sometimes, these items aren’t errors, but are purposeful shortcuts left by programmers for honest or dishonest reasons
- These failures range from bugs to untested failure conditions.

### **12. Technological obsolescence**

- Outdated infrastructure can lead to unreliable and untrustworthy systems.

- Management must recognize that when technology becomes outdated, there is a risk of loss of data integrity from attacks.
- Ideally, proper planning by management should prevent the risks from technology obsolesce, but when obsolescence is identified, management must take action

## Attacks

- An attack is the deliberate act of or action that takes advantage of a vulnerability to compromise a controlled system.
- It is accomplished by a **threat agent** that damages or steals an organization's information or physical asset.
- **Vulnerability** is an identified weakness in a controlled system, where controls are not present or are no longer effective.
- Attacks exist when a specific act or action comes into play and may cause a potential loss.

### Malicious code

- The malicious code attack includes the execution of viruses, worms, Trojan horses, and active Web scripts with the intent to destroy or steal information.
- The goal is to destroy or corrupt data or to shut down a network or computer system.
- The state -of-the-art malicious code attack is the polymorphic or multivector, worm.
- These attack programs use up to six known attack vectors to exploit a variety of vulnerabilities in commonly found information system devices.

### Attack Replication Vectors

1. IP scan & attack
2. Web browsing
3. Virus
4. Unprotected shares
5. Mass mail
6. Simple Network Management Protocol (SNMP)

#### **1. IP scan & attack**

The infected system scans a random or local range of IP addresses and targets any of several vulnerabilities known to hackers.

#### **2. Web browsing**

If the infected system has written access to any Web pages, it makes all Web content files (.html,.asp,.cgi & others) infectious, so that users who browse to those pages become infected.

#### **3. Virus**

Each infected machine infects certain common executable or script files on all computers to which it can write with virus code that can cause infection.

#### **4. Unprotected shares**

Using vulnerabilities in file systems and the way many organizations configure them, the infected machine copies the viral component to all locations it can reach.



## 5. Mass Mail

By sending E-mail infections to addresses found in the address book, the infected machine infects many users, whose mail -reading programs also automatically run the program & infect other systems.

## 6. Simple Network Management Protocol (SNMP)

- By using the widely known and common passwords that were employed in early versions of this protocol, the attacking program can gain control of the device. Most vendors have closed these vulnerabilities with software upgrades.

### Man-in-the -Middle

- Otherwise called as **TCP hijacking attack**.
- An attacker monitors packets from the network, modifies them, and inserts them back into the network.
- This type of attack uses IP spoofing.
- It allows the attacker to change, delete, reroute, add, forge or divert data.
- TCP hijacking session, the spoofing involves the interception of an encryption key exchange.

### SPAM

- Spam is unsolicited commercial E-mail.
- It has been used to make malicious code attacks more effective.
- Spam is considered as a trivial nuisance rather than an attack.
- It is the waste of both computer and human resources it causes by the flow of unwanted E-mail.

### Mail Bombing

- Another form of E-mail attack that is also a DOS called a **mail bomb**.
- Attacker routes large quantities of e-mail to the target.
- The target of the attack receives unmanageably large volumes of unsolicited e-mail.
- By sending large e-mails, attackers can take advantage of poorly configured e-mail systems on the Internet and trick them into sending many e-mails to an address chosen by the attacker.
- The target e-mail address is buried under thousands or even millions of unwanted e-mails.

### Sniffers

- A **sniffer** is a program or device that can monitor data traveling over a network.
- Unauthorized sniffers can be extremely dangerous to a network's security, because they are virtually impossible to detect and can be inserted almost anywhere.
- Sniffers can be used both for legitimate network management functions and for stealing information from a network.
- Sniffer often works on TCP/IP networks, where they are sometimes called "**packet Sniffers**".

## Timing Attack

- Works by exploring the contents of a web browser's cache.
- These attacks allow a Web designer to create a malicious form of cookie, that is stored on the client's system.
- The cookie could allow the designer to collect information on how to access password-protected sites.
- another attack by the same name involves attempting to intercept cryptographic elements to determine keys and encryption algorithms

## Password Crack

- Attempting to reverse calculate a password is often called **cracking**.
- A password can be hashed using the same algorithm and compared to the hashed results, If they are same, the password has been cracked.
- The (SAM) Security Account Manager file contains the hashed representation of the user's password.

## Brute Force

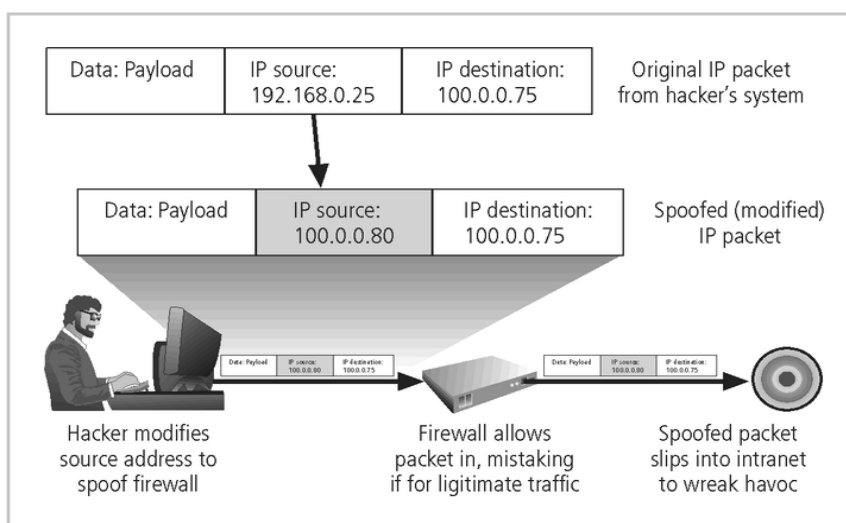
- The application of computing & network resources to try every possible combination of options of a password is called a **Brute force attack**.
- This is often an attempt to repeatedly guess passwords to commonly used accounts, it is sometimes called a **password attack**.

## Dictionary

- This is another form of the brute force attack noted above for guessing passwords.
- The **dictionary attack** narrows the field by selecting specific accounts to attack and uses a list of commonly used passwords instead of random combinations.

## Spoofing

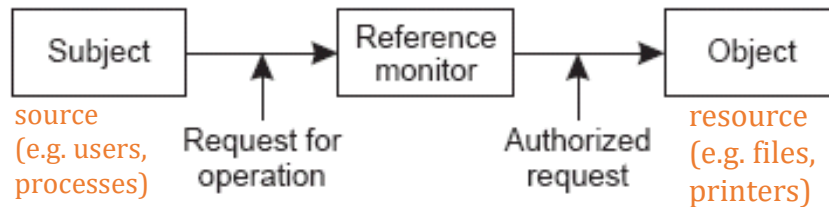
- It is a technique used to gain unauthorized access to computers, where in the intruder sends messages to a computer that has an IP address that indicates that the messages are coming from a trusted host.



**FIGURE 2-10** IP Spoofing

## Access Control Mechanisms - Access Control

- Once a client and a server have established a secure channel, the client can issue requests to the server
- Requests can only be carried out if the client has sufficient *access rights*
- The verification of access rights is *access control*, and the granting of access rights is *authorization*
  - These two terms are often used interchangeably



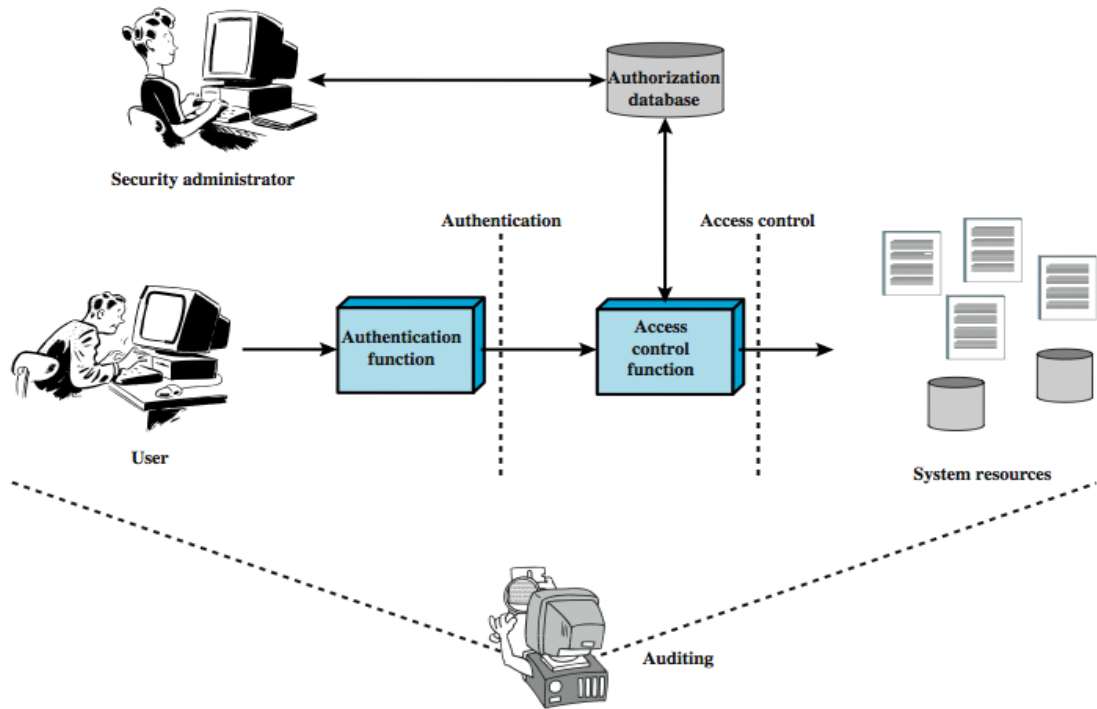
- This model is generally used to help understand the various issues involved in access control

### Access Control Elements

- Subject/Principal: active entity – user or process; often have 3 classes: **owner, group, world**
- Object: passive entity – files, directories, records, programs or resource
- Access operations: read, write, execute, delete, create, search
- Access operations vary from basic memory/file access to method calls in an object-oriented system.
- The *subject* issues request to access the *object*, and protection is enforced by a *reference monitor* that knows which subjects are allowed to issue which requests
- The ability to allow only authorized users, programs or processes system or resource access
- The granting or denying, according to a particular security model, of certain permissions to access a resource
- An entire set of procedures performed by hardware, software and administrators, to monitor access, identify users requesting access, record access attempts, and grant or deny access based on pre-established rules.
- Access control is the heart of security

### Examples of Access Control

- Social Networks: In most social networks, such as Facebook and MySpace, some of your personal information can only be accessed by yourself, some can be accessed by your friends, and some can be accessed by everybody. The part of system that implements such kind of control is doing access control.
- Web Browsers: When you browse a web site, and run JavaScript code from that web site, the browser has to control what such JavaScript code can access, and what it cannot access. For example, a code from one web site cannot access the cookies from another web site, and it cannot modify the contents from another web site either. These controls are conducted by the browser's access control.
- Firewalls: Firewalls inspect every incoming (sometimes outgoing) packet, if a packet matches with certain conditions, it will be dropped by the firewalls, preventing it from accessing the protected networks. This is also access control.



### What should we learn about access control?

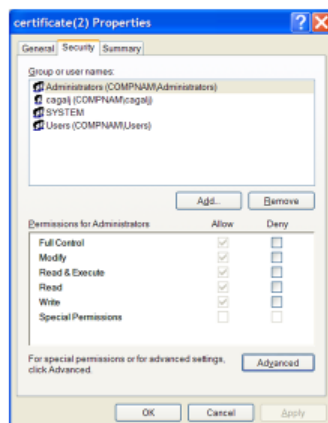
- Access Control Policy Models: how access control policies are configured and managed.
  - Discretionary Access Control (DAC)
  - Mandatory Access Control (MAC)
- Access Control Mechanism: how access control is implemented in systems.
  - Access Control Matrices
  - Access Control List
  - Capability
  - Role-Based Access Control

### Examples: UNIX and WINDOWS

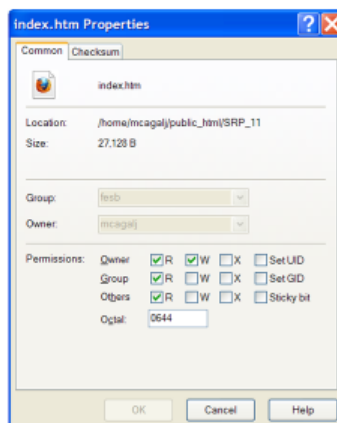
#### Examples: UNIX and WINDOWS

Access control is the part of security that constrains the actions that are performed in a system based on access control rules.

#### Windows File Access Control



#### Unix File Access Control



## Access control matrix

- Access rights can be defined individually for each combination of subject and object.
- The *access control matrix* is a matrix with each subject represented by a row, and each object represented by a column.
- Firstly, identify the objects, subjects and actions.
- Describes the protection state of a system.
- State of the system is defined by a triple (S, O, A)
  - S is the set of subjects,
  - O is the set of objects,
  - A is the access matrix
- Elements indicate the access rights that subjects have on objects
  - Subjects  $S = \{ s_1, \dots, s_n \}$
  - Objects  $O = \{ o_1, \dots, o_m \}$
  - Rights  $R = \{ r_1, \dots, r_k \}$
  - Entries  $A[s_i, o_j] \subseteq R$
  - $A[s_i, o_j] = \{ r_x, \dots, r_y \}$  means subject  $s_i$  has rights  $r_x, \dots, r_y$  over object  $o_j$
- Entry  $A[s, o]$  of access control matrix is the privilege of  $s$  on  $o$
- Access control matrix:  $M = (M_{so})_{s \in S, o \in O}, M_{so} \subseteq A$ ;  
 $M_{so}$  specifies the operations subject  $s$  may perform on object  $o$ .

objects (entities)

	$o_1$	...	$o_m$	$s_1$	...	$s_n$
subjects	$s_1$					
	$s_2$					
	.					
	.					
	.					
	$s_n$					

		Resources		
		File A	File B	File C
Subjects	Alice	read	read	read
	Bob	read, write		
	Charlie		read, write	read, write

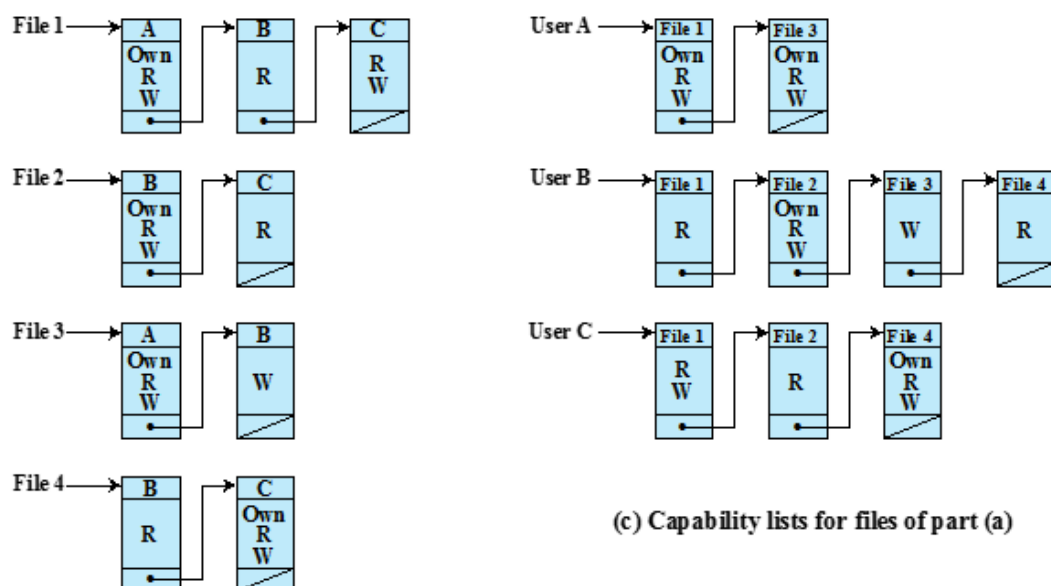
Permissions

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

(a) Access matrix

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

## Access matrix data structures



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

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

## An Access Control Model

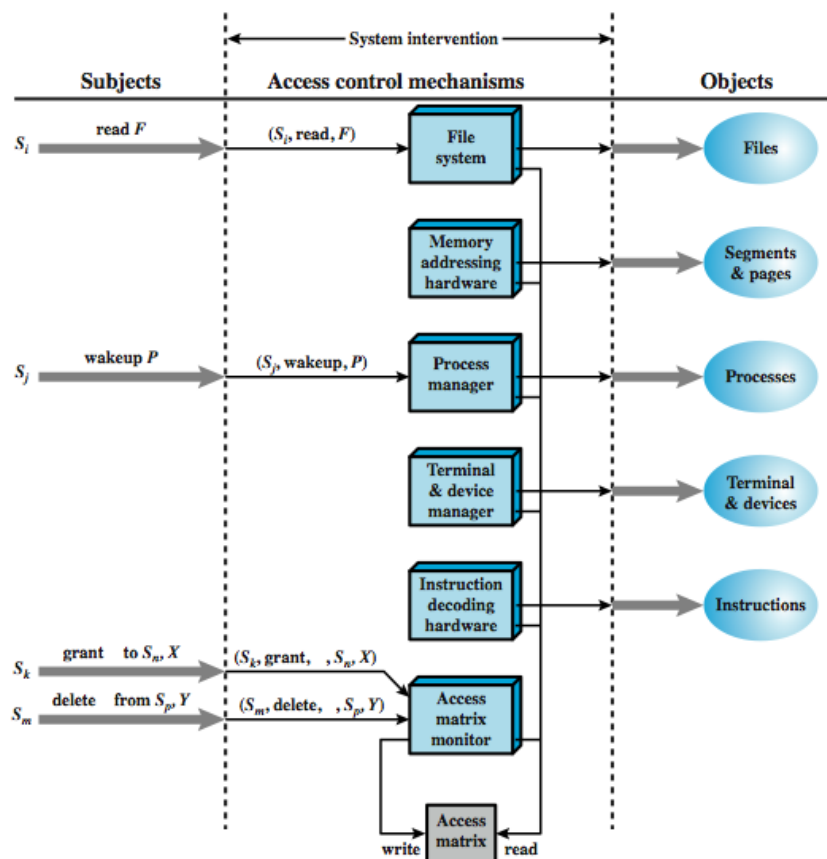
- Extend the universe of objects to include processes, devices, memory locations, subjects

		OBJECTS								
		subjects			files		processes		disk drives	
		S <sub>1</sub>	S <sub>2</sub>	S <sub>3</sub>	F <sub>1</sub>	F <sub>1</sub>	P <sub>1</sub>	P <sub>2</sub>	D <sub>1</sub>	D <sub>2</sub>
SUBJECTS	S <sub>1</sub>	control	owner	owner control	read *	read owner	wakeup	wakeup	seek	owner
	S <sub>2</sub>		control		write *	execute			owner	seek *
	S <sub>3</sub>			control		write	stop			

\* - copy flag set

- Set of objects together with access rights to those objects
- More flexibility when associating capabilities with protection domains
- In terms of the access matrix, a row defines a protection domain
- User can spawn processes with a subset of the access rights of the user
- Association between a process and a domain can be static or dynamic
- In user mode certain areas of memory are protected from use and certain instructions may not be executed
- In kernel mode privileged instructions may be executed and protected areas of memory may be accessed

## Access Control Function



## Access control system commands

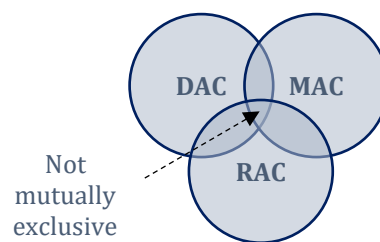
Rule	Command (by $S_o$ )	Authorization	Operation
R1	transfer $\left\{ \begin{matrix} \alpha^* \\ \alpha \end{matrix} \right\}$ to $S, X$	' $\alpha^*$ ' in $A[S_o, X]$	store $\left\{ \begin{matrix} \alpha^* \\ \alpha \end{matrix} \right\}$ in $A[S, X]$
R2	grant $\left\{ \begin{matrix} \alpha^* \\ \alpha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



Who can assign permissions?

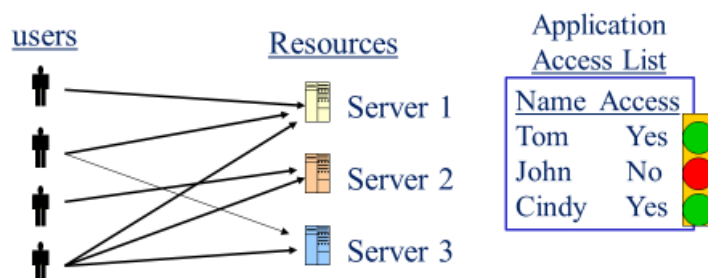
### Access Control Policies

- **Discretionary Access Control (DAC)**
  - User-oriented security policy (based on identity of requestor)
  - Permissions are set *at the discretion* of the resource owner
  - Entity has rights to enable another entity to access a resource
- **Mandatory Access Control (MAC)**
  - Access permissions are defined by a system itself
  - Permissions are assigned by a central authority according to a central policy
  - Based on comparing security labels of system resources (e.g. top security, low security) with security clearances of entities accessing the resources
  - Cleared entity cannot pass on access rights to another entity
- **Role-Based Access Control (RBAC)**
  - Based on roles that users have within system and on rules stating what accesses are allowed to users in given roles



### Discretionary Access Control (DAC)

- In discretionary access control (DAC), the owner of the object specifies which subjects can access the object. This model is called discretionary because the control of access is based on the discretion of the owner.
- Most operating systems such as all Windows, Linux, and Macintosh and most flavours of Unix are based on DAC models.
- In these operating systems, when you create a file, you decide what access privileges you want to give to other users; when they access your file, the operating system will make the access control decision based on the access privileges you created.



- Permissions are set *at the discretion* of the resource owner
  - Highly flexible policy, where permissions can be transferred
  - Lack of central control makes revocation or changes difficult
- Discretionary access control in use
  - Controlling access to files
    - E.g., Windows Access Control Lists (ACL), UNIX file handles
  - Controlling the sharing of personal information

- E.g., Social networks
- Restricts access to objects based solely on the identity of users who are trying to access them.

### DAC - Boolean Expression Evaluation

- Access controls access to database fields
  - Subjects have attributes
  - Action/Operation/Verb define type of access
  - Rules associated with objects, action pair
- Subject attempts to access object
  - Rule for object, action evaluated, grants or denies access

### Example

- Subject Annie
  - Attributes role (artist), groups (creative)
- Verb paint
  - Default 0 (deny unless explicitly granted)
- Object picture
  - Rule:  
Annie paint picture if:  
'artist' in subject.role and  
'creative' in subject.groups and  
time.hour  $\geq 0$  and time.hour  $< 5$

### ACM at 3AM and 10AM

At 3AM, time condition  
met; ACM is:

... picture ...

... annie ...			
	paint		

At 10AM, time condition  
not met; ACM is:

... picture ...

... annie ...			

### Access Controlled by History

- Statistical databases need to
  - **answer queries on groups**
  - **prevent revelation of individual records**
- Query-set-overlap control
  - Prevent an attacker to obtain individual piece of information using a set of queries.
  - A parameter  $r$  ( $=2$ ) is used to determine if a query should be answered

Name	Position	Age	Salary
Alice	Teacher	45	40K
Bob	Aide	20	20K
Cathy	Principal	37	60K
Dilbert	Teacher	50	50K
Eve	Teacher	33	50K

- Query 1:
  - sum\_salary(position = teacher)
  - Answer: 140K

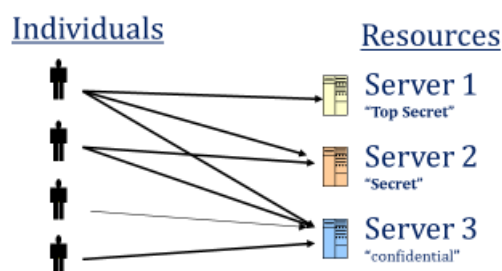
Name	Position	Age	Salary
Celia	Teacher	45	40K
Leonard	Teacher	50	50K
Matt	Teacher	33	50K

- Query 2:
  - sum\_salary(age > 40 & position = teacher)
  - Should not be answered as Matt's salary can be deduced
  - Can be represented as an ACM

Name	Position	Age	Salary
Celia	Teacher	45	40K
Leonard	Teacher	50	50K

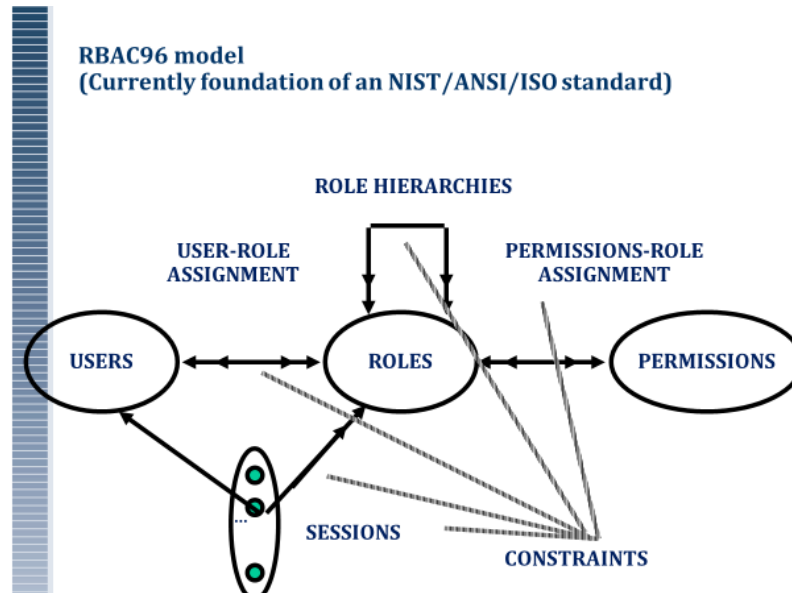
## Mandatory Access Control (MAC)

- In mandatory access control (MAC), **the system (and not the users) specifies which subjects can access specific data objects.**
- The MAC model is based on **security labels**. Subjects are given a **security clearance (secret, top secret, confidential, etc.)** and data objects are given a **security classification (secret, top secret, confidential, etc.)**. The clearance and classification data are stored in the security labels, which are bound to the specific subjects and objects.
- When the system is making an access control decision, **it tries to match the clearance of the subject with the classification of the object.** For example, if a user has a security clearance of secret, and he requests a data object with a security classification of top secret, then the user will be denied access because his clearance is lower than the classification of the object.
- The MAC model is usually used in environments where confidentiality is of utmost importance, such as a military institution.**
- Examples of the **MAC-based commercial systems are SELinux and Trusted Solaris.**
- Better security than DAC
- Permissions are assigned by a central authority according to a central policy
  - Good fit within organizations with a strong need for central controls
  - Low flexibility and high management overhead
  - Mandatory Access Control in use
  - Often linked to multi-level security systems
    - E.g. Government-regulated secrecy systems, military applications
  - Modern operating systems, to separate applications and processes
    - E.g. Windows' *Mandatory Integrity Control*, SELinux, TrustedBSD

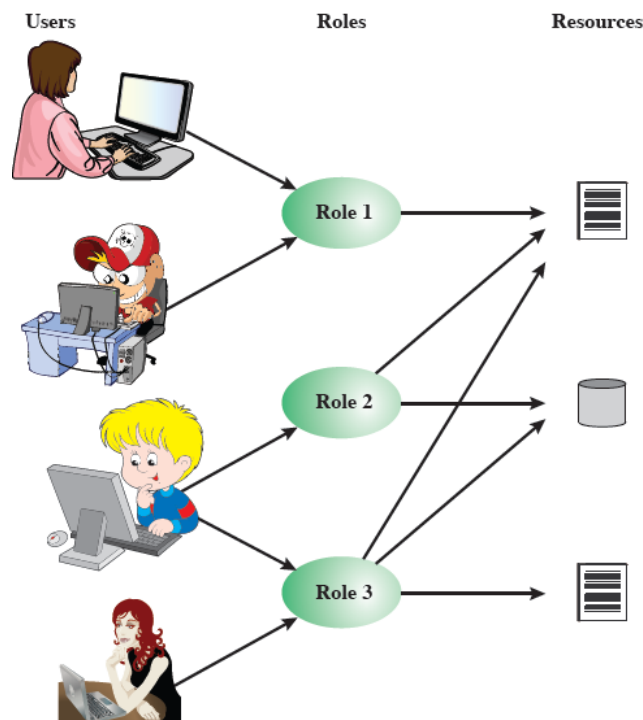


## Role-Based AC

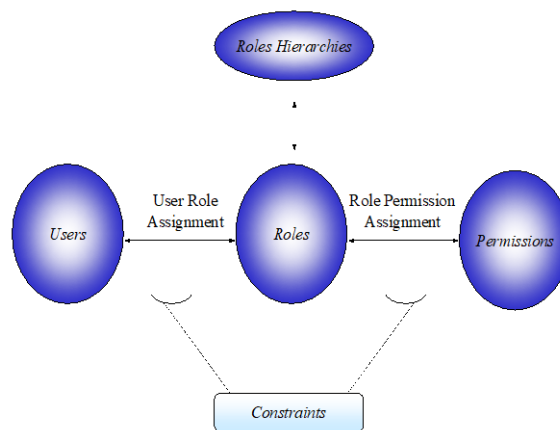
- A user has access to an object based on the assigned role.
- Roles are defined based on job functions.
- Permissions are defined based on job authority and responsibilities within a job function.
- Operations on an object are invoked based on the permissions.
- The object is concerned with the user's role and not the user.



- Access based on 'role', not identity
- Many-to-many relationship between users and roles
- Roles often static



- Many organizations base access control decisions on “**the roles that individual users take on as part of the organization**”.
- They prefer to centrally control and maintain access rights that reflect the organization’s protection guidelines.
- With RBAC, role-permission relationships can be predefined, which makes it simple to assign users to the predefined roles.
- The combination of users and permissions tend to change over time, the permissions associated with a role are more stable.
- RBAC concept supports three well-known security principles:
  - **Least privilege**
  - **Separation of duties**
  - **Data abstraction**
- Access control in organizations is based on “**roles that individual users take on as part of the organization**”
- A role is “**is a collection of permissions**”



- **Access depends on role/function, not identity**
  - Example: Allison is **bookkeeper** for Math Dept. She has access to financial records. If she leaves and Betty is hired as the new **bookkeeper**, Betty now has access to those records. The role of “bookkeeper” dictates access, not the identity of the individual.

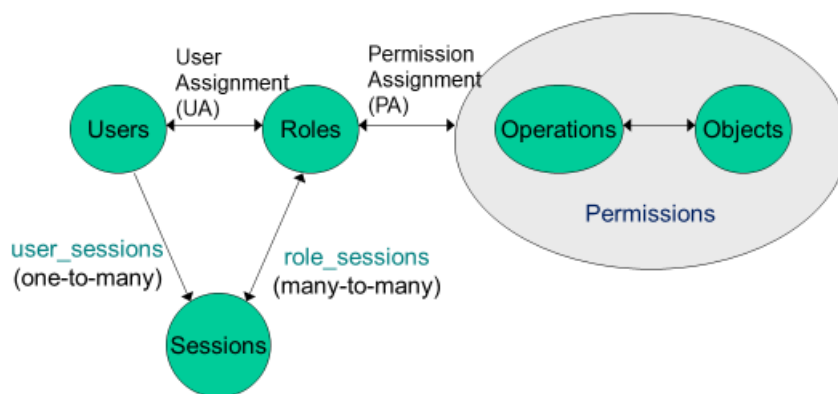
### Advantages of RBAC

- Allows Efficient Security Management
  - Administrative roles, Role hierarchy
- Principle of least privilege allows minimizing damage
- **Separation of Duties** constraints to prevent fraud
- Allows grouping of objects
- **Policy-neutral - Provides generality**
- Encompasses DAC and MAC policies

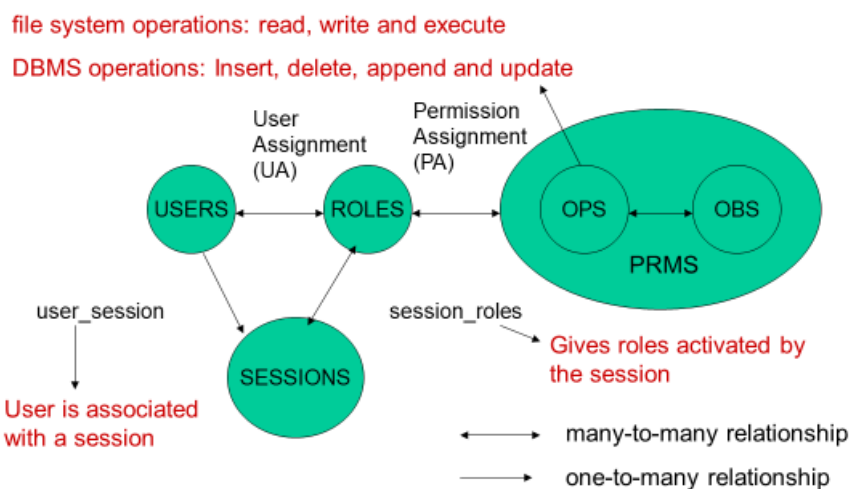
Is RBAC a discretionary or mandatory access control?

- RBAC is **policy neutral**; however individual RBAC configurations can support a mandatory policy, while others can support a discretionary policy.
- **Role Hierarcies**
- Role Administration

## RBAC (NIST Standard)

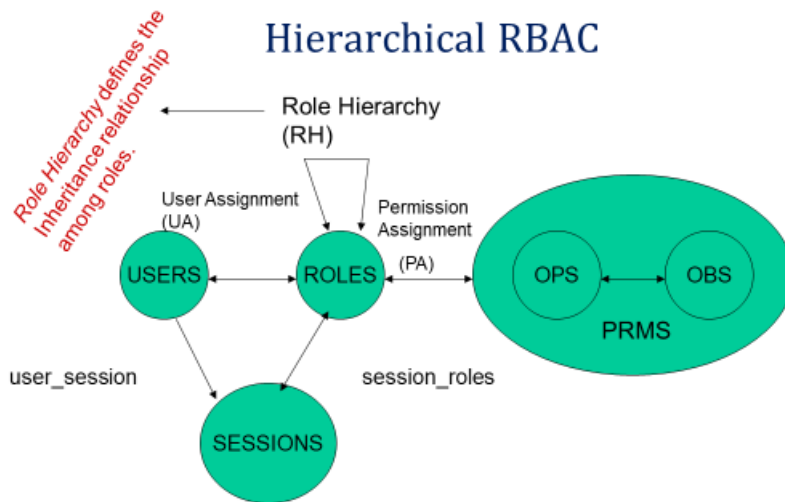


An important difference from classical models is that **Subject** in other models corresponds to a **Session** in RBAC



## Hierarchical RBAC

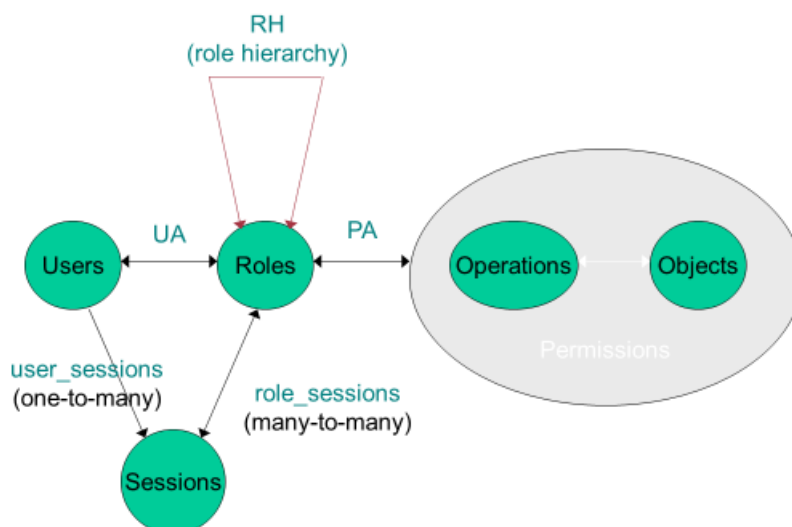
Role of Specialist could contain the roles of Doctor and Intern. members of the role Specialist are implicitly associated with the operations associated with the roles Doctor and Intern without the administrator having to explicitly list the Doctor and Intern operations. Moreover, the roles Cardiologist and Rheumatologist could each contain the Specialist role.



### Core RBAC (relations)

- $\text{Permissions} = 2^{\text{Operations} \times \text{Objects}}$
- $\text{UA} \subseteq \text{Users} \times \text{Roles}$
- $\text{PA} \subseteq \text{Permissions} \times \text{Roles}$
- $\text{assigned\_users: Roles} \rightarrow 2^{\text{Users}}$
- $\text{assigned\_permissions: Roles} \rightarrow 2^{\text{Permissions}}$
- $\text{Op}(p)$ : set of operations associated with permission  $p$
- $\text{Ob}(p)$ : set of objects associated with permission  $p$
- $\text{user\_sessions: Users} \rightarrow 2^{\text{Sessions}}$
- $\text{session\_user: Sessions} \rightarrow \text{Users}$
- $\text{session\_roles: Sessions} \rightarrow 2^{\text{Roles}}$ 
  - $\text{session\_roles}(s) = \{r \mid (\text{session\_user}(s), r) \in \text{UA}\}$
- $\text{avail\_session\_perms: Sessions} \rightarrow 2^{\text{Permissions}}$

### RBAC with General Role Hierarchy



## RBAC with General Role Hierarchy

- *authorized\_users*: Roles  $\rightarrow 2^{\text{Users}}$   
 $authorized\_users(r) = \{u \mid r' \geq r \ \& \ (r', u) \in UA\}$
  - *authorized\_permissions*: Roles  $\rightarrow 2^{\text{Permissions}}$   
 $authorized\_users(r) = \{p \mid r' \geq r \ \& \ (p, r') \in PA\}$
  - RH Roles x Roles is a partial order
    - called the inheritance relation
    - written as  $\geq$ .
- $(r_1 \geq r_2) \rightarrow authorized\_users(r_1) \subseteq authorized\_users(r_2) \ \& \ authorized\_permissions(r_2) \subseteq authorized\_permissions(r_1)$

## case study SELinux

Why Linux?

Linux is an open source project with many developers; therefore:

- Provides an opportunity for more research.
- Allows application/testing in a mainstream OS.
- Improves security in an existing OS.

Why SELinux?

- Security-Enhanced Linux
- Uses the Linux Security Modules (LSM) framework to implement flexible Mandatory Access Control (MAC) in the Linux kernel.
- Restricts privileges of user programs and system servers using security labels and an administratively-defined policy.

What is SELinux?

- Developed by NSA(National Security Agency)
  - Released in 2000
- Adds additional security capabilities to Linux
- Maintains compatibility with existing software
- “Designed to enforce separation of information based on confidentiality and integrity requirements.”
- Open source
  - GPL

### SELinux Security Models

- Type Enforcement (TE)
  - Confine processes (subjects) to domains by using security contexts.
- Role-based Access Control (RBAC)
  - Recognizes that users often need to move from 1 domain to another. RBAC rules explicitly allow roles to move from one domain to another
  - Users are assigned to one or more roles
  - Roles indicate which type domains a user may access
  - Similar to traditional Unix uid
  - Used to separate privileges
  - Each daemon may have its own role
  - Example roles include system\_r, sysadm\_r, user\_r
  - Role transitions must be defined



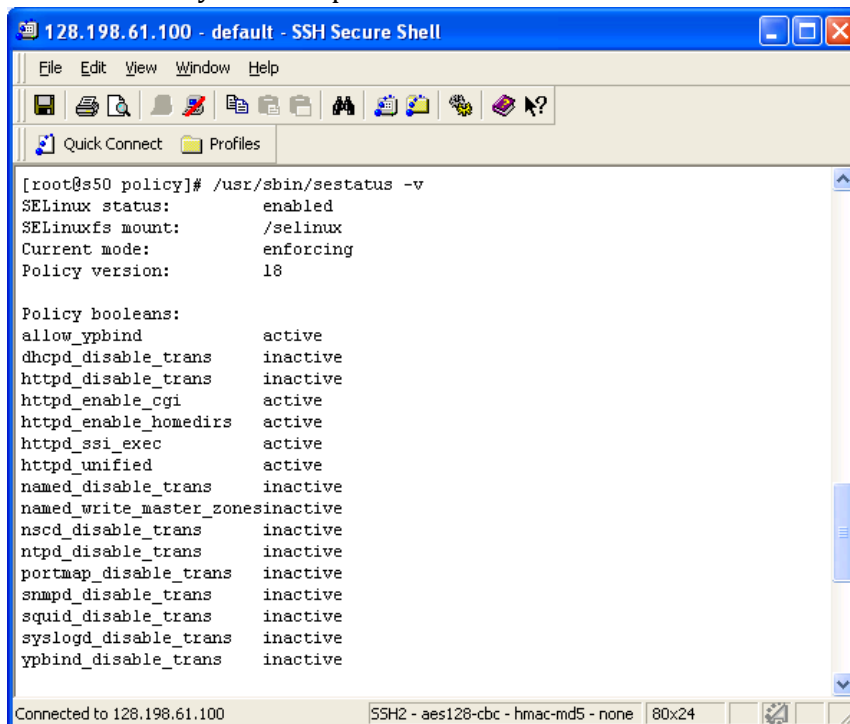
- Multi-Level Security
  - Enforce Bell-LaPadula security model.
  - Users allowed to read at one level cannot read at higher levels. Also users allowed to write at 1 level are not allowed to write at a lower level. (Ensures that secure information does not propagate to lower levels.

## Policies

- A policy is a set of rules which specifies allowable behavior
- Strict versus targeted
  - Enumerating good versus bad behavior
  - No “default permit”
- Defines
  - Types for file objects
  - Domains for processes
  - Roles
  - User identities
- Highly configurable with booleans

Configuration consists of:

- Flask definitions
- TE and RBAC declarations and rules
- User declarations
- Constraint definitions
- Security context specifications.



```

128.198.61.100 - default - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles

[root@ss50 policy]# /usr/sbin/sestatus -v
SELinux status:                enabled
SELinuxfs mount:              /selinux
Current mode:                  enforcing
Policy version:                18

Policy booleans:
allow_yppbind                  active
dhcpd_disable_trans           inactive
httpd_disable_trans           inactive
httpd_enable_cgi              active
httpd_enable_homedirs         active
httpd_ssi_exec                active
httpd_unified                 active
named_disable_trans           inactive
named_write_master_zones      inactive
nscd_disable_trans            inactive
ntpd_disable_trans            inactive
portmap_disable_trans         inactive
snmpd_disable_trans           inactive
squid_disable_trans           inactive
syslogd_disable_trans         inactive
ypbind_disable_trans          inactive

Connected to 128.198.61.100    SSH2 - aes128-cbc - hmac-md5 - none    80x24
  
```

- Based on a strong, flexible mandatory access control architecture based on Type Enforcement, a mechanism first developed for the LOCK system
- Theoretically, can be configured to provide high security.
- In practice, mostly used to confine daemons like web servers
  - They have more clearly defined data access and activity rights.
  - They are often targets of attacks
  - A confined daemon that becomes compromised is thus limited in the harm it can do.
- Ordinary user processes often run in the unconfined domain
  - not restricted by SELinux, but still restricted by the classic Linux access rights.

### SELinux Terminology

- Subject: A domain or process.
- Object: A resource (file, directory, socket, etc.).
- Types: A security attribute for files and other objects.
- Roles: A way to define what “types” a user can use.
- Identities: Like a username, but specific to SELinux.
- Contexts: Using a type, role and identity is a “Context.”

Context is -> Identities : role : type

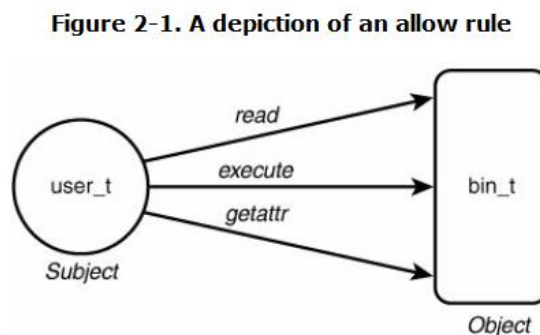
### SELinux Allow Rule

- Source type(s) Usually the domain type of a process attempting access
- Target type(s) The type of an object being accessed by the process
- Object class(es) The class of object that the specified access is permitted
- Permission(s) The kind of access that the source type is allowed to the target type for the indicated object classes

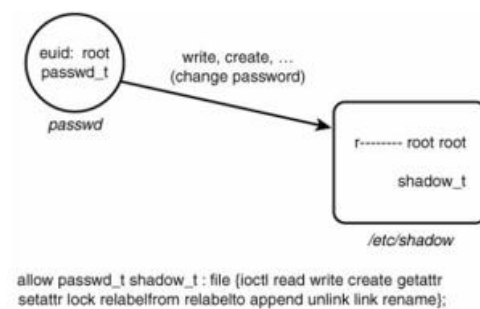
### Rule Example1:

allow user\_t bin\_t : file {read execute getattr};

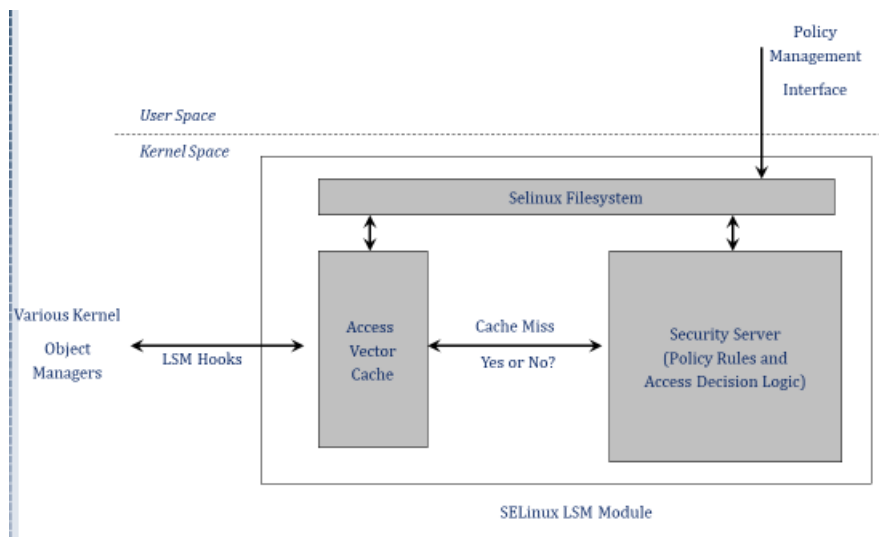
->A process with a domain type of user\_t can read, execute, or get attributes for a file object with a type of bin\_t



## Rule Example2 : passwd program in linux



## SELinux LSM Module



```

~]$ ls -Z /usr/bin/passwd
-rwsr-xr-x.    root  root  system_u:object_r:passwd_exec_t:s0/usr/bin/passwd

~]$ ls -Z /etc/shadow
-----,      root  root  system_u:object_r:shadow_t:s0/etc/shadow
  
```

**SELinux policies:**

- applications running in the passwd\_t domain can access files labeled with the shadow\_t type
- the passwd\_t domain can be entered from the passwd\_exec\_t type

## SELinux Commands

- Policy Control
  - checkpolicy (check and create a new policy)
  - load\_policy
  - setfiles
  - restorecon
  - chcon

- semanage
- Process related context information (in man)
  - ftpd\_selinux
  - named\_selinux
  - rsync\_selinux
  - httpd\_selinux
  - nfs\_selinux
  - samba\_selinux
  - kerberos\_selinux
  - nis\_selinux
  - ypbind\_selinux

#### Example

- Execute the command “ls -Z /usr/bin/passwd”
  - This will produce the output:
 

```
-r-s—x—x root root
system_u:object_r:passwd_exec_t /usr/bin/passwd
```
  - Using this provided information, we can then create TE rules to have a domain transition.
- Three rules are required to give the user the ability to do a domain transition to the password file:
  - **allow user\_t passwd\_exec\_t : file {getattr execute};**
    - Lets user\_t execute an execve() system call on passwd\_exec\_t
  - **allow passwd\_t passwd\_exec\_t : file entry point;**
    - This rule provides entry point access to the passwd\_t domain, entry point defines which executable files can “enter” a domain.
  - **allow user\_t passwd\_t : process transition;**
    - The original type (user\_t) must have transition permission to the new type (passwd\_t) for the domain transition to be allowed.
- This isn't very useful by itself since the user would have to specifically say that they want a domain transition. This is where type transition rules are used.
- To create a domain transition by default the following rule is created:
  - type\_transition user\_t passwd\_exec\_t : process passwd\_t;
  - The type\_transition rule indicates that by default on an execve() system call, if the calling process' domain type is user\_t and the executable file's type is passwd\_exec\_t a domain transition to a new domain type (passwd\_t) will be attempted
- A type\_transition rule causes a domain transition to be attempted by default, but it does not allow it, that's why the other 3 rules had to be created

### Model Questions:

1. Define Information Security.
2. What are the Information Security components? Explain in detail.
3. What are the measures to protect the confidentiality of information?
4. What are the characteristics of CIA triangle?  
What are the characteristics of Information Security?
5. Explain NSTISSC Security Model with neat diagram.
6. What are the components of information system?
7. Explain in detail about need of security functions.
8. What is a threat?
9. Explain the categories of Threat in detail.
10. What are the four types of Intellectual property?
11. What is software piracy?
12. What is Shoulder Surfing?
13. What are Hackers?
14. What are the levels of hackers?
15. What is a Phreaker?
16. What is Malicious code?
17. What are the types of virus?
18. What is worm?
19. What are trojan horses?
20. What is a polymorphic threat?
21. What are Hoaxes?
22. What are the attack replication vectors?
23. What is a brute force attack?
24. What are sniffers?
25. What is technological obsolescence?
26. What is an attack?
27. Explain the types of Attacks in detail
28. What do you mean by access control?
29. What are the types of access control policies?
30. Explain in detail about the Access control matrix with neat diagram.
31. Explain Discretionary Access Control in detail.
32. Explain Mandatory Access Control in detail.
33. Explain Role-Based Access Control in detail.
34. What are the security models in SELinux.
35. Explain in detail about SELinux with example.
36. Give the SELinux commands.