

IE2022 – Introduction to Cyber Security

Lecture - 01

Introduction

Mr. Amila Senarathne

Lecture 1: Introduction to Cyber Security

Objective:

- * Describe the formal definition of Computer Security
- * Describe Confidentiality, Integrity, and Availability as the key security requirements
- * Computer Security Model and Strategy
- * Describe the security threats and attacks types

Recommended Texts

W. Stallings and L. Brown, “Computer Security, Principles and Practice, 2nd edition, Pearson, 2012, Chapter 1.

Supplementary text

Charles P. Pfleeger and Shari L. Pfleeger, Security in Computing (3rd edition). Prentice-Hall. 2003. ISBN: 0-13-035548-8.

Computer Security

Definition (NIST Computer Security Handbook)

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).

Key objectives of Computer Security:

- * Confidentiality
- * Integrity
- * Availability

Information Security (InfoSec)

The protection of information and information systems from unauthorized access, use, disclosure, disruption, modification, or destruction in order to provide confidentiality, integrity, and availability.

(Source : NIST Glossary of Key Information Security Terms)

Computer Security Objectives

1) Confidentiality (C).

This term covers two related concepts.

- Data confidentiality. Assures that confidential information is not made available or disclosed to unauthorized individuals.
- Privacy. Assures that the owners have control on:
 - * What information related to them may be collected and stored,
 - * By whom and to whom that information may be disclosed.

NIST's Requirement: Preserve authorized restrictions on information access and disclosure, including means for protecting personal privacy and proprietary information

Loss of confidentiality means unauthorized disclosure of information.

Objectives (cont.)

2) Integrity (I).

This term covers two related concepts.

- Data integrity: Information and programs are changed only in a specified and authorized manner.
- System integrity: A system performs its intended function
 - * in an unimpaired manner, and
 - * free from deliberate or inadvertent unauthorized manipulation of the system.

Requirement: Guard against improper information modification or destruction, including ensuring information nonrepudiation authenticity.

Loss of Integrity means unauthorized modification or destruction of information.

Objectives (cont.)

3) Availability (A).

Systems work promptly and service is not denied to authorized users.

NIST's requirement: Ensuring timely and reliable access and use of information.

Loss of Availability means disruption to the authorized users in accessing or use of information.

Figure from Stallings
& Brown textbook

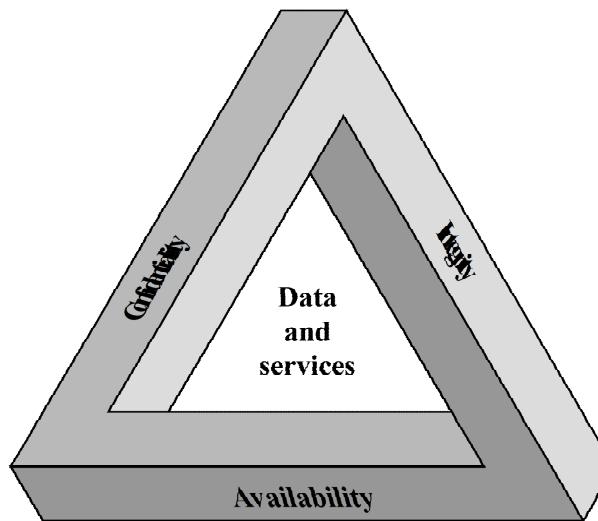


Figure 1.1 The Security Requirements Triad

Additional Objectives

4) **Authenticity:** Able to verify that

- the users are who they claim they are, and
- the system receives data from a trusted source.

NIST includes authenticity as part of Integrity

5) **Accountability:** Able to trace back the actions performed by an entity to that entity.

Accountability supports: nonrepudiation, deterrence, fault isolation, intrusion detection and prevention, after-action recovery and legal action.

Read the examples of C-I-A in the textbook (Stallings & Brown)

Computer Security Model (RFC 2828)

1) System Resource or asset that needs to be protected

Hardware: e.g., Computer System, data storage, communication devices.

Software: e.g., operating systems, program utilities and applications.

Data: e.g., data and password files, databases.

Communication facilities and networks: e.g., LAN, WAN, routers, etc.

2) Vulnerabilities of system resources

Definition: A flaw or weaknesses in a system's design, implementation, or operation and management that could be exploited to violate the system's security policy.

When the resource is corrupted → violate Integrity

When the resource is leaky → violate Confidentiality

When the resource is unavailable → violate Availability

Computer Security Model

3) **Threat** is a possible danger that might exploit a vulnerability.

It represents a potential harm to the system resource.

4) **Attack** is a threat that is carried out (threat action)

Two attack types:

- * Active attack: An act that has negative effects on system resources
- * Passive attack: An act to make use of system information but it does not affect the system

The origin of an attack:

- * Inside attack is carried out by an entity inside the security perimeter.
- * Outside attack is performed by an unauthorized users.

Computer Security Model (cont.)

5) **Adversary** is an entity that carried out an attack

- A threat agent or an attacker.

6) **Countermeasure** is any means taken

- to address an attack,
- to prevent an attack from being successful,
- to detect the attack if the attack is successful, and
- to recover from the damage due to the attack.

7) **Risk** is the expected loss due to a particular attack.

- Examples?

Exploits

An exploit is a piece of software, a chunk of data, or a sequence of commands that takes advantage of a bug or vulnerability to cause unintended or unanticipated behavior to occur on computer software, hardware, or something electronic (usually computerized).

Such behavior frequently includes things like gaining control of a computer system, allowing privilege escalation, or a denial-of-service (DoS or related DDoS) attack.

Used as a verb, exploit refers to the act of successfully making such an attack (make use of a vulnerability).

Vulnerability Assessment

A vulnerability assessment is the process of defining, identifying, classifying and prioritizing vulnerabilities in Information systems, applications and network infrastructures and providing the organization doing the assessment with the necessary knowledge, awareness and risk background to understand the threats to its environment and react appropriately.

Penetration Testing

Penetration testing (also called pen testing or ethical hacking) is the practice of testing a Information system, network or web application to find security vulnerabilities that an attacker could exploit. The process involves gathering information about the target before the test, identifying possible entry points, attempting to break in either virtually or for real and reporting back the findings.

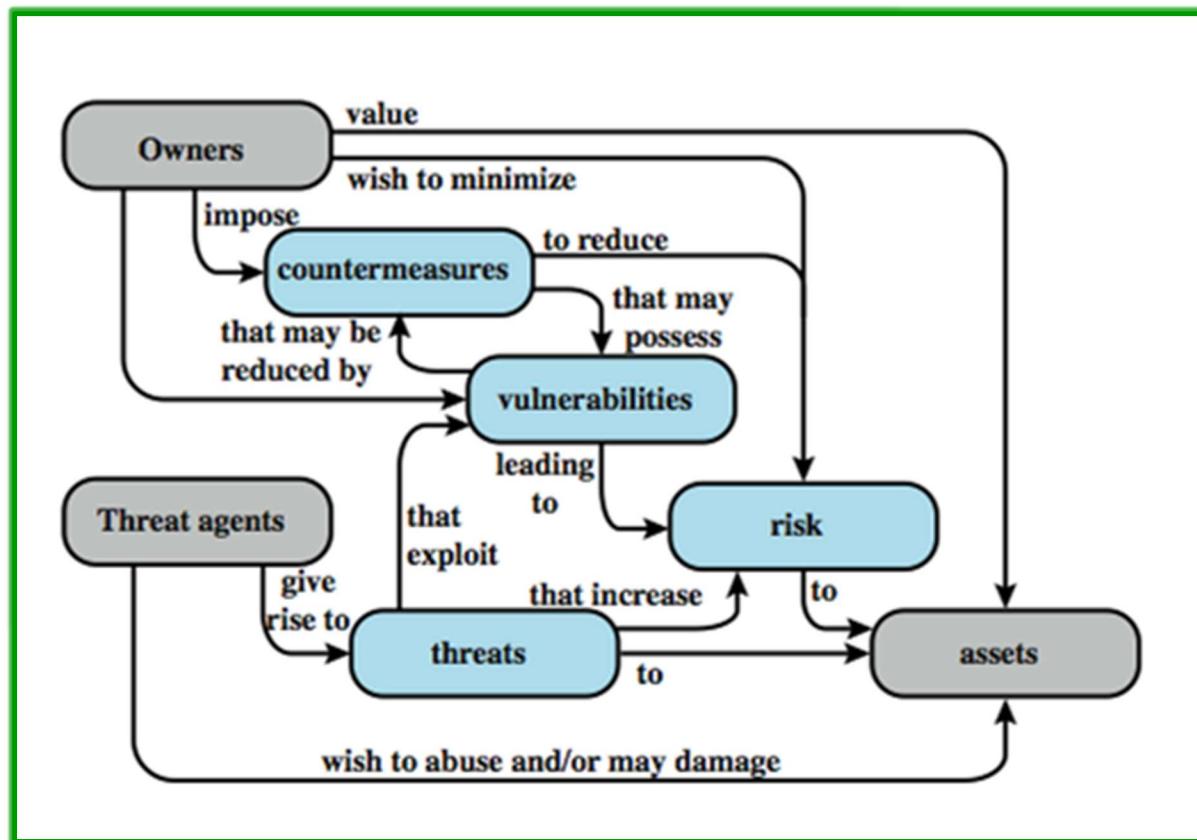
Penetration testing can be automated with software applications or performed manually.

Goal of Penetration Testing

- Identify weak spots in an organization's security posture
- Measure the compliance of its security policy
- Test the staff's awareness of security issues
- Determine whether and how the organization would be subject to security disasters.

Computer Security Model

Figure from Stallings
& Brown textbook



Threats and Attacks

Four kinds of threats and their types of attacks (RFC 2828)

1) **Unauthorized disclosure:** a threat to system confidentiality

Types of Attacks:

Exposure. The attacker obtains unauthorized knowledge of sensitive data.

Interception. The attacker gain access to data being transmitted

- A common attack in communication network

Inference. The attacker gains information from analyzing the pattern of traffic in a network

Intrusion. The attacker gains unauthorized access to data

- Probably after breaking the system's access control protection

Threats and Attacks (cont.)

2) **Deception:** a threat to system or data integrity

Types of Attacks:

Masquerade. The attacker accesses to the system acting as an authorized user

- the attacker may have the login name and password.

Falsification. The attacker modifies or replaces valid data or produces false data

Repudiation. The attacker denies

- sending the data,
- denies receiving the data, or
- Possessing the data

Threats and Attacks

3) **Disruption:** a threat to system availability and integrity

Types of Attacks:

Incapacitation. An attack on system availability by destructing or damaging system resources (e.g., hardware) and their services.

Corruption. An attack to system integrity such that the system resources or services operate in an unintended manner.

- This can be done by a malware or an attacker that modifies system function

Obstruction. An attack to system availability by interfering, altering, or overloading communication functions

Threats and Attacks

4) **Usurpation:** a threat to system integrity

Types of Attacks:

Misappropriation. An unauthorized software uses the OS and hardware resources

- E.g., DoS attack that steals system services

Misuse. Disabling security functions, can be by the following means:

- malicious logic, or
- an attacker that gains access to the system

Threats and Assets

Four categories of assets and their attacks.

- 1) **Threats on hardware:** attack on system availability
e.g., damaging or stealing the hardware
- 2) **Threats on software:** attack of system availability and integrity/authenticity
e.g., deleting and damaging (availability), and modifying (integrity/authenticity) the software
- 3) **Threats on data:** attack on availability, integrity and confidentiality
e.g., destroying data (availability), accessing and analyzing unauthorized data (confidentiality), and modifying data (integrity)

Threats and Assets

- 4) **Threats on communication lines and networks:** can be passive or active attacks

Passive attack is performed by eavesdropping or monitoring data transmission

- * The attacker only learns or makes use of information without affecting system resources
- * Passive attack is hard to detect because data is not altered
 - Use attack prevention (not detection) to handle it

Two types of passive attacks.

- * Release of message contents (confidentiality)
- * Traffic analysis, if the data is encrypted.

Threats and Assets (cont.)

4) Threats on communication lines and networks (cont.)

Active attacks alters system resources or affecting their operations

- * Active attack is difficult to prevent but easy to detect

Four categories of active attack:

Replay. Capture and retransmit data unit to produce an unauthorized effect

Masquerade. One entity pretends to be another entity

- It usually includes other form of attack, e.g., replay

Data modification. Alter some portion of legitimate data, delay the data, or reorder the data to produce an unauthorized effect

Denial of Service. Prevent or disallow the legitimate use of facilities

Security Functional Requirement

FIPS PUB 200 (NIST) lists 17 security related areas to protect confidentiality, integrity, and availability of systems and information stored, processed and transmitted in the system.

Countermeasures to security vulnerabilities and threats are divided into two categories:

- 1) Those that require computer security technical measures: access control, identification and authentication, system and communication protection, system information integrity.
- 2) Those that are fundamentally management issues: awareness and training, audit and accountability, certification, accreditation, and security assessments, etc.

Access control: Limit information system access to authorized users, processes acting on behalf of authorized users, or devices (including other information systems) and to the types of transactions and functions that authorized users are permitted to exercise.

Awareness and training: (i) Ensure that managers and users of organizational information systems are made aware of the security risks associated with their activities and of the applicable laws, regulation, and policies related to the security of organizational information systems; and (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 information system audit records to the extent needed to enable the monitoring, analysis, investigation, and reporting of unlawful, unauthorized, or inappropriate information system activity; and (ii) ensure that the actions of individual information system 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 organizational information systems 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 organizational information systems; (iii) authorize the operation of organizational information systems and any associated information system connections; and (iv) monitor information system security controls on an ongoing basis to ensure the continued effectiveness of the controls.

Configuration management: (i) Establish and maintain baseline configurations and inventories of organizational information systems (including hardware, software, firmware, and documentation) throughout the respective system development life cycles; and (ii) establish and enforce security configuration settings for information technology products employed in organizational information systems.

Contingency planning: Establish, maintain, and implement plans for emergency response, backup operations, and postdisaster recovery for organizational information systems to ensure the availability of critical information resources and continuity of operations in emergency situations.

Identification and authentication: Identify information system users, processes acting on behalf of users, or devices and authenticate (or verify) the identities of those users, processes, or devices, as a prerequisite to allowing access to organizational information systems.

Incident response: (i) Establish an operational incident handling capability for organizational information systems that includes adequate preparation, detection, analysis, containment, recovery, and user response activities; and (ii) track, document, and report incidents to appropriate organizational officials and/ or authorities.

Maintenance: (i) Perform periodic and timely maintenance on organizational information systems; and (ii) provide effective controls on the tools, techniques, mechanisms, and personnel used to conduct information system maintenance.

**Table 1.4
(FIPS PUB 200)**

Security Requirements

From Stallings &
Brown textbook



Media protection: (i) Protect information system media, both paper and digital; (ii) limit access to information on information system media to authorized users; and (iii) sanitize or destroy information system media before disposal or release for reuse.

Physical and environmental protection: (i) Limit physical access to information systems, equipment, and the respective operating environments to authorized individuals; (ii) protect the physical plant and support infrastructure for information systems; (iii) provide supporting utilities for information systems; (iv) protect information systems against environmental hazards; and (v) provide appropriate environmental controls in facilities containing information systems.

Planning: Develop, document, periodically update, and implement security plans for organizational information systems that describe the security controls in place or planned for the information systems and the rules of behavior for individuals accessing the information systems.

Personnel security: (i) Ensure that individuals occupying positions of responsibility within organizations (including third-party service providers) are trustworthy and meet established security criteria for those positions; (ii) ensure that organizational information and information systems are protected during and after personnel actions such as terminations and transfers; and (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 organizational information systems and the associated processing, storage, or transmission of organizational information.

Systems and services acquisition: (i) Allocate sufficient resources to adequately protect organizational information systems; (ii) employ system development life cycle processes that incorporate information security considerations; (iii) employ software usage and installation restrictions; and (iv) ensure that third-party providers employ adequate security measures to protect information, applications, and/or services outsourced from the organization.

System and communications protection: (i) Monitor, control, and protect organizational communications (i.e., information transmitted or received by organizational information systems) at the external boundaries and key internal boundaries of the information systems; and (ii) employ architectural designs, software development techniques, and systems engineering principles that promote effective information security within organizational information systems.

System and information integrity: (i) Identify, report, and correct information and information system flaws in a timely manner; (ii) provide protection from malicious code at appropriate locations within organizational information systems; and (iii) monitor information system security alerts and advisories and take appropriate actions in response.

OSI Security Architecture

- * The International Telecommunication Union (ITU) Recommendation X.800 defines the Security Architecture for Open System Interconnection (OSI) Architecture
 - To assess the security needs of an organization
 - To evaluate and choose various security products and policies
 - To define security requirements and approaches to satisfy the requirements
- * OSI Security Architecture focuses on
 - Security Attack. Any action that compromises the security information owned by an organization.
 - Security Mechanism. A process to detect, prevent, or recover from a security attack.
 - Security Service. A service that enhances the security of the data processing systems and the information transfers of an organization to counter security attacks by making use of one or more security mechanisms

OSI Security Services

X 800 divides security services into six categories and 14 specific services.

- * X800 focuses on distributed and networked systems
 - It stresses on network security than single computer security

Six categories of security services:

- 1) **Authentication.** Make sure that a communication is authentic
- 2) **Access control.** Limit and control accesses to host systems through communication channels
- 3) **Data confidentiality.** Protect data from passive attacks
- 4) **Data Integrity.** Make sure that data received is that sent by authorized entity
- 5) **Nonrepudiation.** Prevent sender or receiver from denying a transmitted data.
- 6) **Availability.** Prevent denial of authorized access to system resources

Table 1.5
Security Services

Figure from Stallings & Brown textbook

AUTHENTICATION	DATA INTEGRITY
The assurance that the communicating entity is the one that it claims to be.	The assurance that data received are exactly as sent by an authorized entity (i.e., contain no modification, insertion, deletion, or replay).
Peer Entity Authentication Used in association with a logical connection to provide confidence in the identity of the entities connected.	Connection Integrity with Recovery Provides for the integrity of all user data on a connection and detects any modification, insertion, deletion, or replay of any data within an entire data sequence, with recovery attempted.
Data-Origin Authentication In a connectionless transfer, provides assurance that the source of received data is as claimed.	Connection Integrity without Recovery As above, but provides only detection without recovery.
ACCESS CONTROL	SELECTIVE-FIELD CONNECTION INTEGRITY
The prevention of unauthorized use of a resource (i.e., this service controls who can have access to a resource, under what conditions access can occur, and what those accessing the resource are allowed to do).	Provides for the integrity of selected fields within the user data of a data block transferred over a connection and takes the form of determination of whether the selected fields have been modified, inserted, deleted, or replayed.
DATA CONFIDENTIALITY	CONNECTIONLESS INTEGRITY
The protection of data from unauthorized disclosure.	Provides for the integrity of a single connectionless data block and may take the form of detection of data modification. Additionally, a limited form of replay detection may be provided.
Connection Confidentiality The protection of all user data on a connection.	Selective-Field Connectionless Integrity Provides for the integrity of selected fields within a single connectionless data block; takes the form of determination of whether the selected fields have been modified.
Connectionless Confidentiality The protection of all user data in a single data block.	NONREPUDIATION
Selective-Field Confidentiality The confidentiality of selected fields within the user data on a connection or in a single data block.	Provides protection against denial by one of the entities involved in a communication of having participated in all or part of the communication.
Traffic-Flow Confidentiality The protection of the information that might be derived from observation of traffic flows.	Nonrepudiation, Origin Proof that the message was sent by the specified party.
AVAILABILITY	Nonrepudiation, Destination Proof that the message was received by the specified party.
Ensures that there is no denial of authorized access to network elements, stored information, information flows, services and applications due to events impacting the network. Disaster recovery solutions are included in this category.	

OSI Security Mechanism

X800 divides security mechanism into

- * Those specific to specific protocol layers and protocol applications, e.g., TCP
- * Others.

SPECIFIC SECURITY MECHANISMS	PERVERSIVE SECURITY MECHANISMS
May be incorporated into the appropriate protocol layer in order to provide some of the OSI security services.	Mechanisms that are not specific to any particular OSI security service or protocol layer.
Encipherment The use of mathematical algorithms to transform data into a form that is not readily intelligible. The transformation and subsequent recovery of the data depend on an algorithm and zero or more encryption keys.	Trusted Functionality That which is perceived to be correct with respect to some criteria (e.g., as established by a security policy).
Digital Signature Data appended to, or a cryptographic transformation of, a data unit that allows a recipient of the data unit to prove the source and integrity of the data unit and protect against forgery (e.g., by the recipient).	Security Label The marking bound to a resource (which may be a data unit) that names or designates the security attributes of that resource.
Access Control A variety of mechanisms that enforce access rights to resources.	Event Detection Detection of security-relevant events.
Data Integrity A variety of mechanisms used to assure the integrity of a data unit or stream of data units.	Security Audit Trail Data collected and potentially used to facilitate a security audit, which is an independent review and examination of system records and activities.
Authentication Exchange A mechanism intended to ensure the identity of an entity by means of information exchange.	Security Recovery Deals with requests from mechanisms, such as event handling and management functions, and takes recovery actions.
Traffic Padding The insertion of bits into gaps in a data stream to frustrate traffic analysis attempts.	
Routing Control Enables selection of particular physically secure routes for certain data and allows routing changes, especially when a breach of security is suspected.	
Notarization The use of a trusted third party to assure certain properties of a data exchange.	

TABLE 1.6

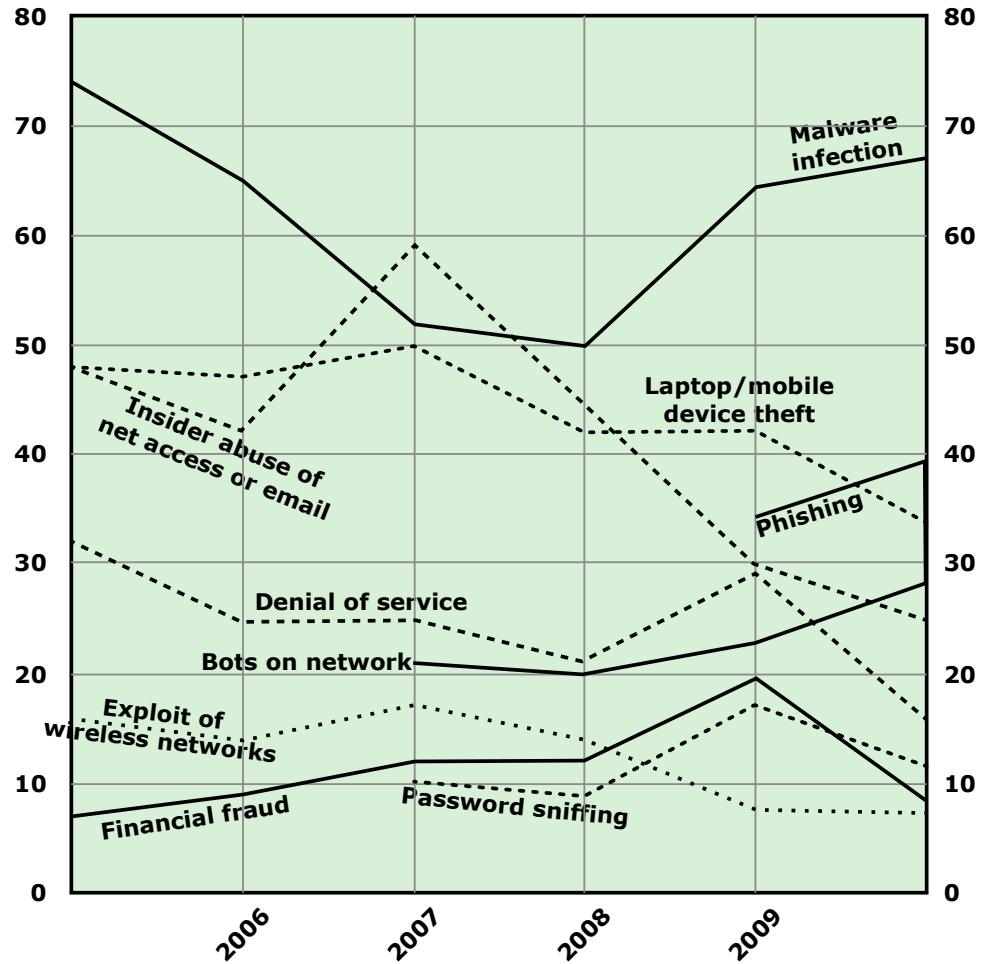
X.800 Security Mechanisms

Figure from Stallings & Brown textbook

Computer Security Trends

Survey (2010/2011) conducted by Computer Security Institute with respondents from 350 organizations in US based on

- * Types of Attacks (see Fig. 1.4)
 - There is growing incidents on malware infection
- * Security Technology used (See Fig. 1.5)
 - Most organizations use anti-virus software and firewalls



Source: Computer Security Institute 2010/2011 Computer Crime and Security Survey

**Figure 1.4
Security Trends**

Figure from Stallings & Brown textbook

Figure 1.4 Types of Attacks Experienced

(Percent of respondents)

IE2022 | Introduction to Cyber Security | Lecture 01 | Amila Senarathne

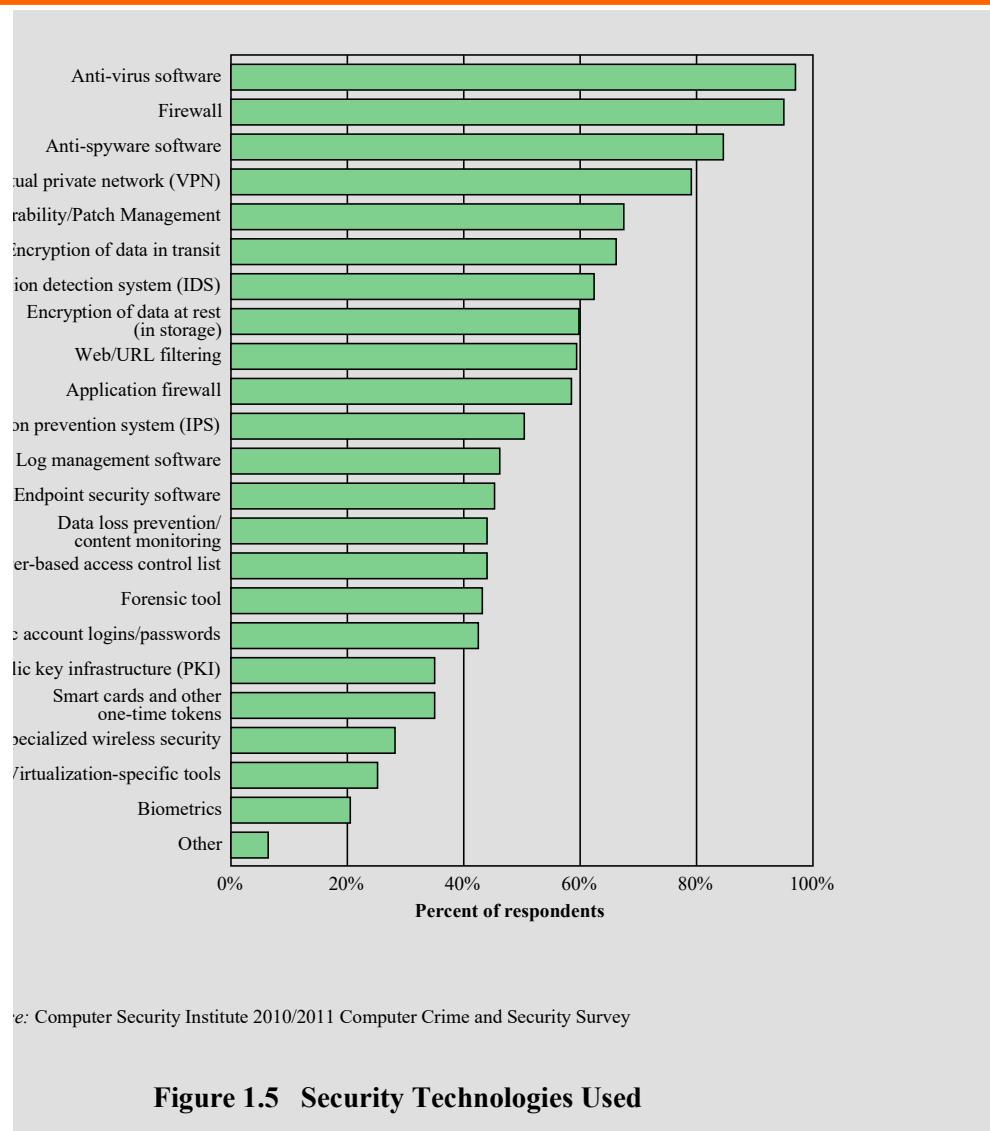


Figure 1.5 Security Technologies Used

Figure from Stallings & Brown textbook

Computer Security Strategy

Lampson suggests a security strategy to include three aspects

- ***Specification/policy:** What to do
- ***Implementation/mechanisms:** How to do it
- ***Correctness/assurance:** Does it work

Factors to consider for Security Policy:

- * The value of the assets to be protected
- * The system's vulnerabilities
- * Potential threats and their possible attacks
- * Ease of use versus security
- * Cost of security versus cost of security failure and recovery

Computer Security Strategy (cont.)

Security implementation includes these four complementary actions:

- * **Prevention**

- This is an ideal case; but not always feasible

- * **Detection**

- When prevention is not possible, detect security attacks
 - Can use intrusion detection

- * **Response**

- When an attack is detected, respond to halt the attack or prevent further damage

- * **Recovery**

- Recover from the attack by using, for example, a backup copy

Computer Security Strategy (cont.)

NIST defines **assurance** as:

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

- * Does the security system design meet its requirements?
- * Does the security system implementations meet its specifications?

Evaluation is the process of examining a computer product or system with respect to certain criteria

- * Involves testing and analysis

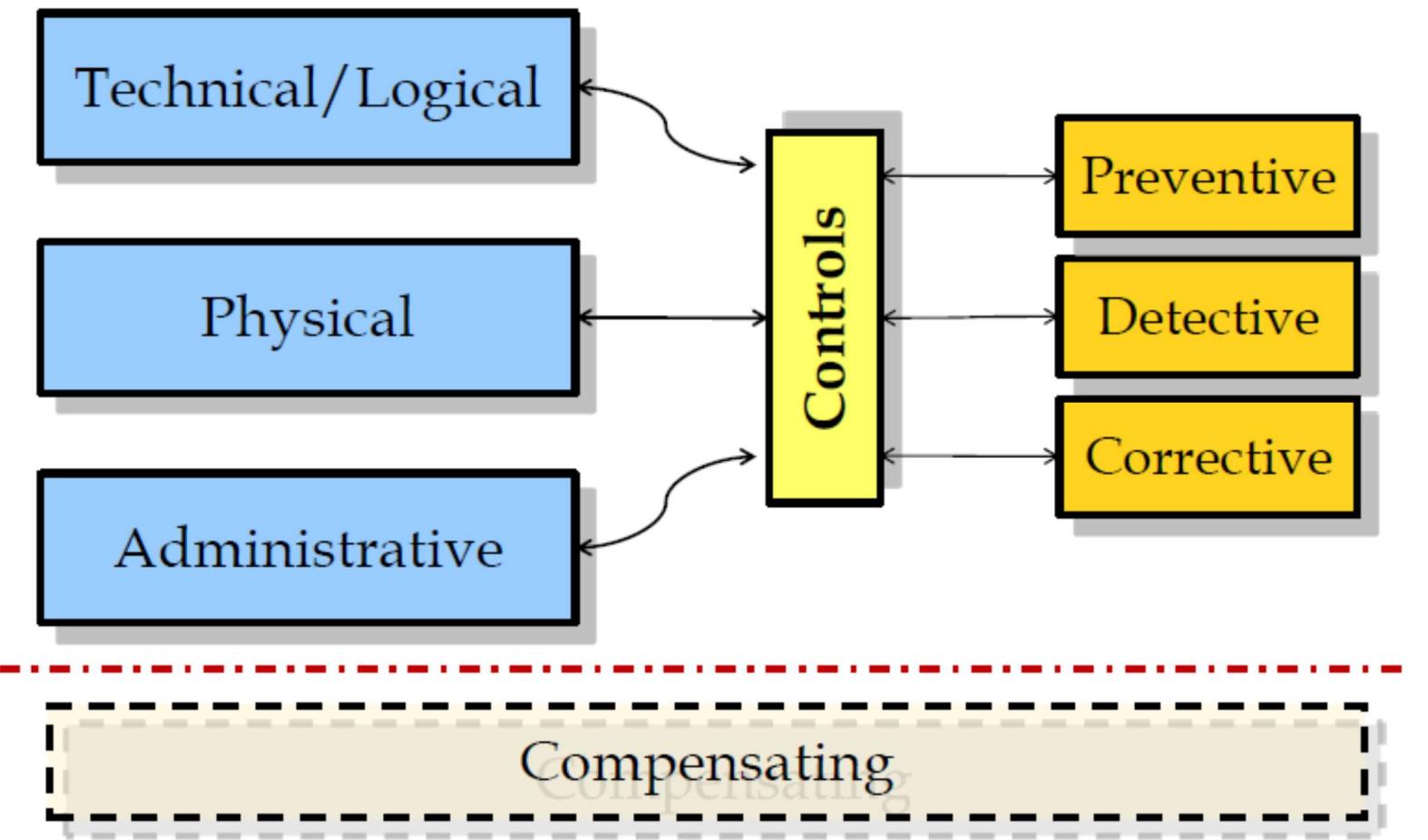
IE2022 – Introduction to Cyber Security

Lecture - 03

Security Controls and Risk Management

Mr. Amila Senarathne

Security Controls



Security Controls

Computer/information security controls are often divided into three distinct categories

- Physical controls
- Technical/Logical controls
- Administrative controls

Physical Controls

The Physical control is the implementation of security measures in a defined structure used to deter or prevent unauthorized access to sensitive material.

- Surveillance cameras
- Motion or thermal alarm systems
- Security guards
- Picture IDs
- Locked and dead-bolted steel doors
- Network segregation
- Work area separation

Technical Controls

The Technical control uses technology as a basis for controlling the access and usage of sensitive data throughout a physical structure and over a network.

- Encryption
- Smart cards
- Network authentication
- Access control lists (ACLs)
- File integrity auditing software
-

Administrative Controls

Administrative controls define the human factors of security. It involves all levels of personnel within an organization and determines which users have access to what resources and information by such means as:

- Training and awareness
- Disaster preparedness and recovery plans
- Personnel recruitment and separation strategies
- Personnel registration and accounting
- Policy and procedures

Controls categorized : By functionality

- Preventive Controls
- Detective Controls
- Deterrent Controls
- Corrective Controls
- Recovery Controls
- Compensating Controls

Preventive Controls

Designed to discourage errors or irregularities from occurring. They are proactive controls that help to ensure departmental objectives are being met.

- Separation of duties
- Security of Assets (Preventive and Detective)
- Planning/testing
- Proper hiring practices
- Proper processing of terminations
- Approvals, Authorizations, and Verifications

Detective Controls

Designed to find errors or irregularities after they have occurred.

- Monitoring Systems
- Log reviews
- Bugler Alarm
- File Integrity checkers
- Security reviews and audits
- Performance evaluations

Deterrent Controls

Intended to discourage potential attackers and send the message that it is better not to attack, but even if you decide to attack we are able to defend ourselves.

- Notices of monitoring logging
- Visible practice of sound information security management.

Corrective Controls

Designed to correct the situation after a security violation has occurred. Although a violation occurred, not all is lost, so it makes sense to try and fix the situation.

- Procedure to clean a virus from an infected system
- A guard checking and locking a door left unlocked by a careless employee
- Updating firewall rules to block an attacking IP address

Recovery Controls

Somewhat like corrective controls, but they are applied in more serious situations to recover from security violations and restore information and information processing resources.

- Disaster recovery and business continuity mechanisms
- Backup systems and data
- Emergency key management arrangements and similar controls.

Compensating Controls

- * Intended to be alternative arrangements for other controls when the original controls have failed or cannot be used.
- * When a second set of controls addresses the same threats that are addressed by another set of controls, the second set of controls are referred to as compensating controls.

Risk Management

What is risk?

- Life is full of risk. We all manage risk consciously or automatically in life.
- Risk is the possibility of damage happening, and the ramifications of such damage should it occur.

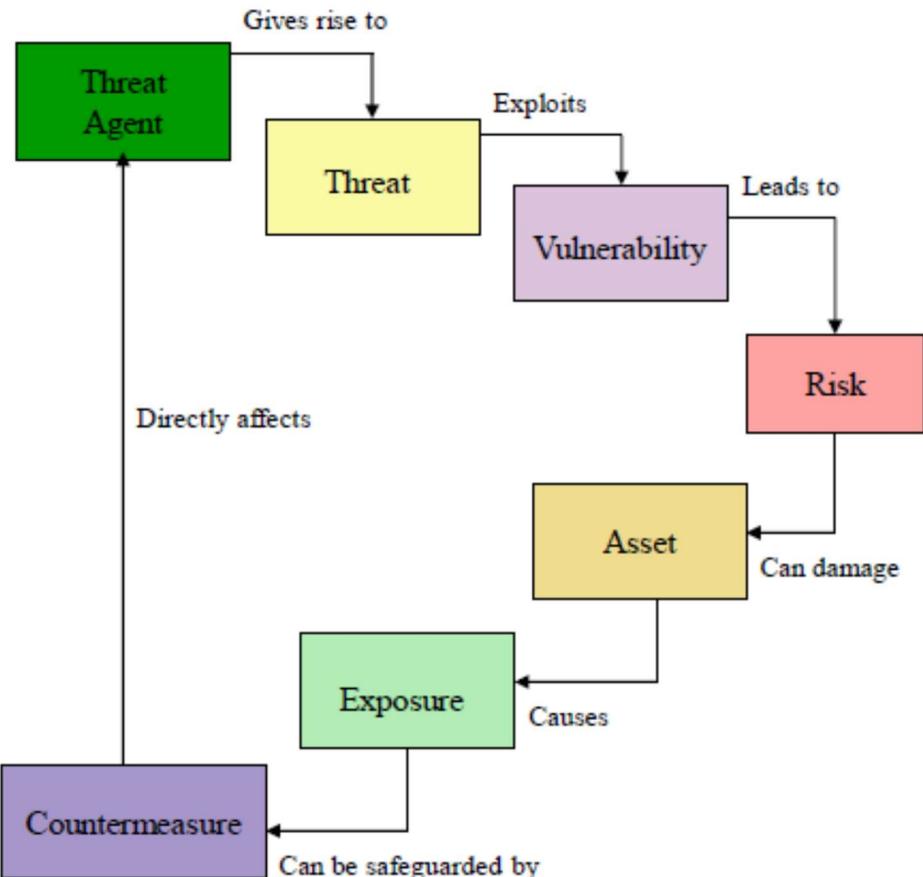
Information Risk Management (IRM) is the process of identifying and assessing risk, reducing it to an acceptable level, and implementing the right mechanisms to maintain that level.

- Risk can be mitigated, but cannot be eliminated (which is usually not an option in the commercial world, where controlled (managed) risk enables profits)

Risk Management Terms

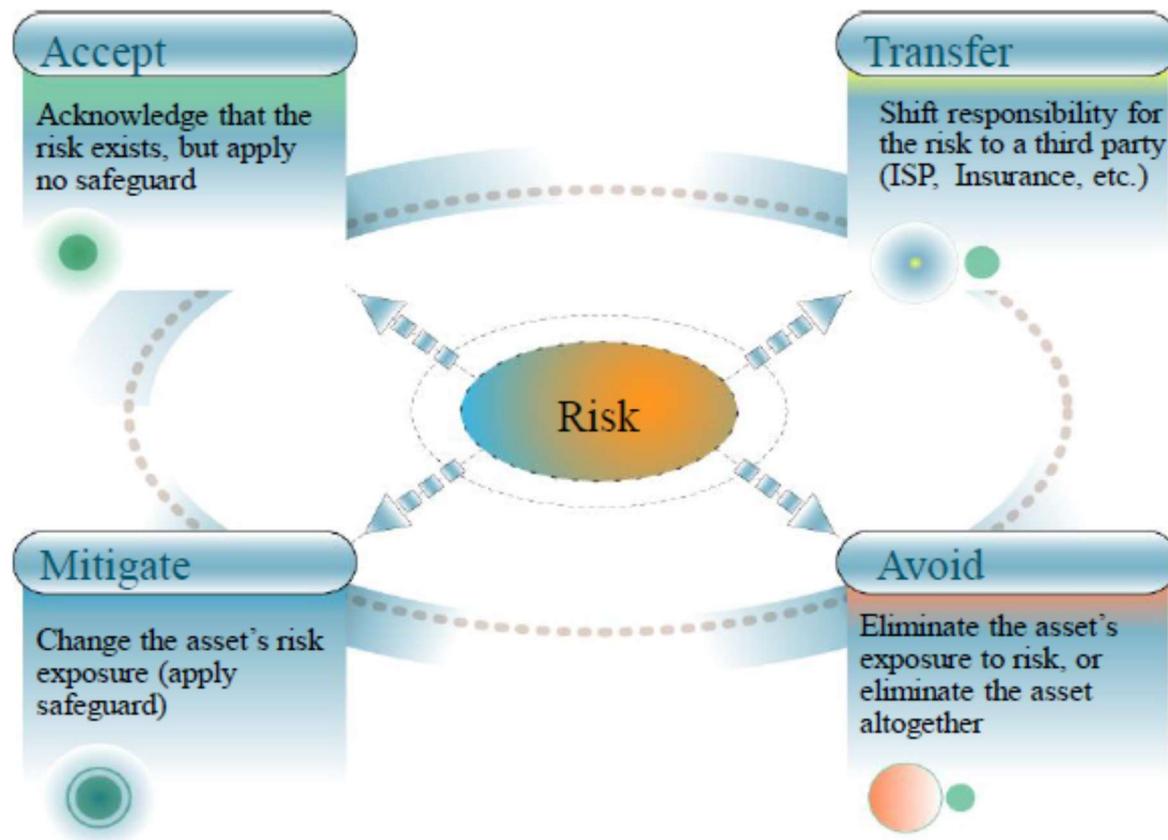
- Vulnerability – a system, network or device weakness
- Threat – potential danger posed by a vulnerability
- Threat agent – the entity that identifies a vulnerability and uses it to attack the victim
- Risk – likelihood of a threat agent taking advantage of a vulnerability and the corresponding business impact
- Exposure – potential to experience losses from a threat agent
- Countermeasure – put into place to mitigate the potential risk

Understanding Risk

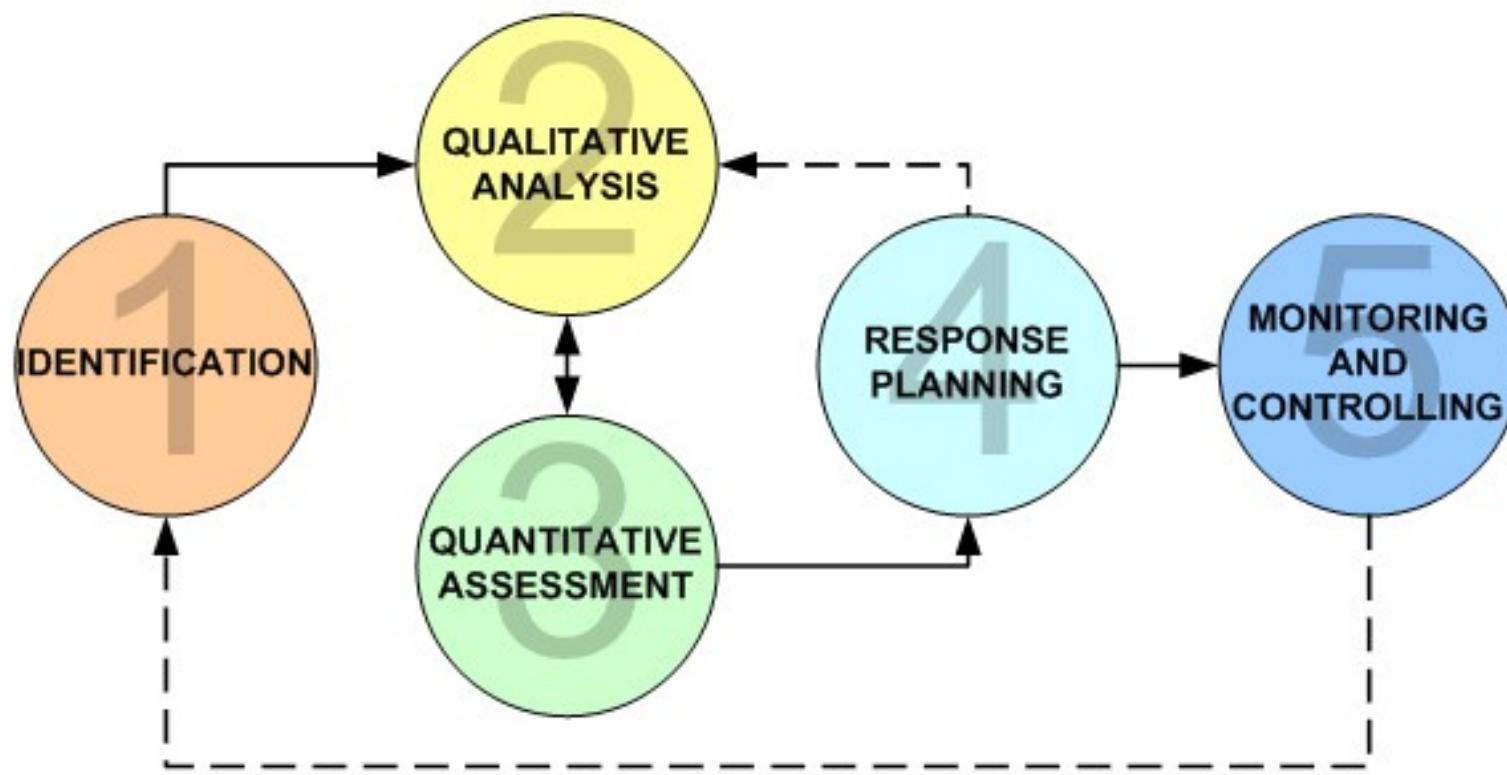


A **threat agent** gives rise to a **threat** that exploits a **vulnerability** and can lead to a **security risk** that can damage your **assets** and cause an **exposure**. This can be counter-measured by a safeguard that directly affects the threat agent.

Managing Risks



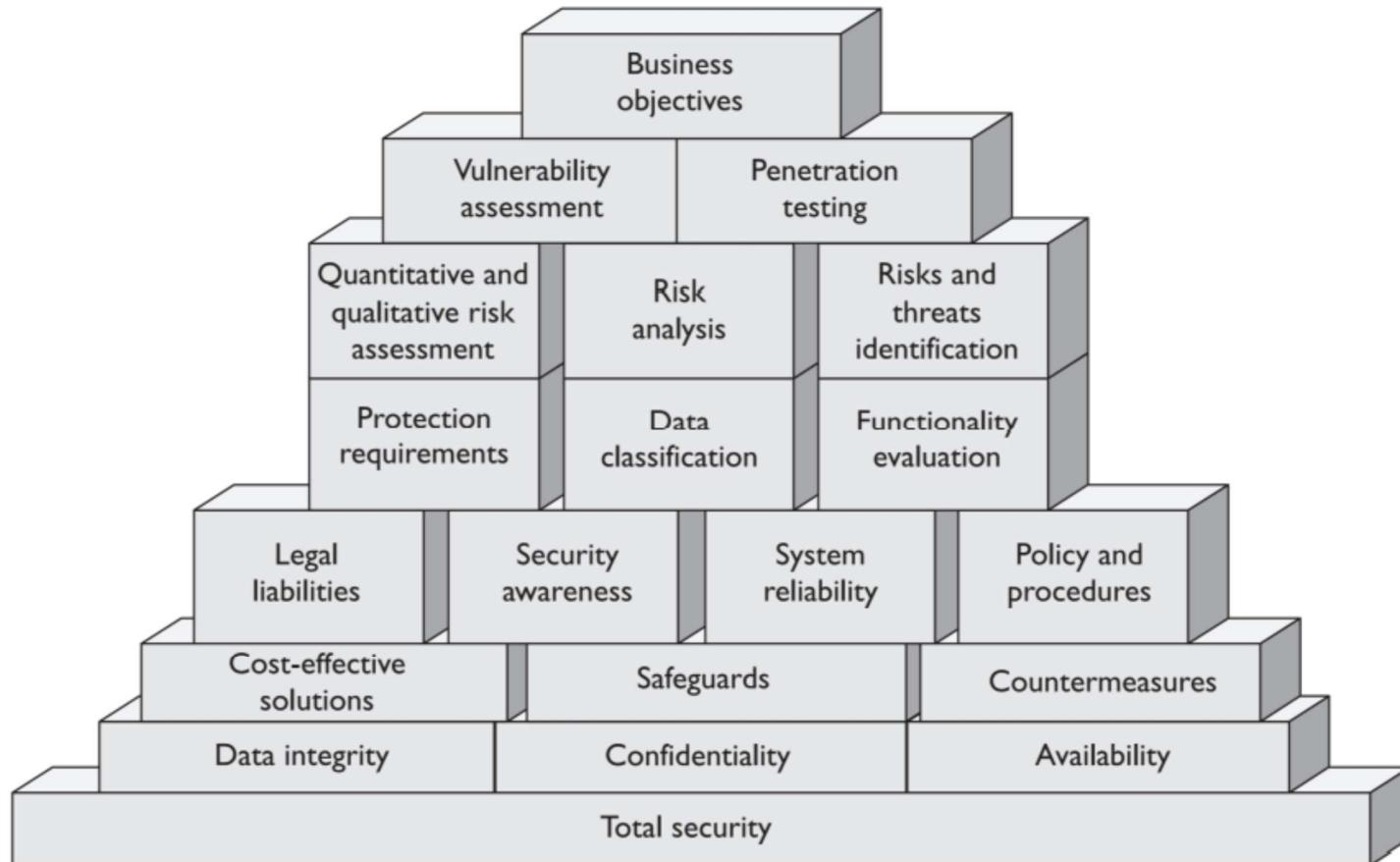
Risk Management Process



Quantitative Risk Analysis

- * **Exposure Factor(EF)** = Percentage of asset loss caused by identified threat (0-100%)
- * **Single Loss Expectancy (SLE)** = Asset Value x EF
e.g. Rs. 50,000 x 20% = Rs. 10,000
- * **Annualized Rate of Occurrence (ARO)** = Frequency a threat will occur within a year
- * **Annualized Loss Expectancy (ALE)** = SLE x ARO
- * **Safeguard Cost/Benefit** = ALE before Safeguard - ALE After Safeguard - Annual Cost of Safeguard

Comprehensive Security Model



IE2022 – Introduction to Cyber Security

Lecture - 04

Data Loss and Hackers

Mr. Amila Senarathne

Data Loss

Data loss or data exfiltration is when data is intentionally or unintentionally lost, stolen, or leaked to the outside world.

Data is likely to be an organization's most valuable asset. Organizational data can include:

- Research and development data
- Sales data
- Financial data
- Human resource and legal data
- Employee data
- Contractor data
- Customer data.

Data Loss can result in:

- ★ Brand damage and loss of reputation
- ★ Loss of competitive advantage
- ★ Loss of customers
- ★ Loss of revenue
- ★ Litigation/legal action resulting in fines and civil penalties
- ★ Significant cost and effort to notify affected parties and recover from the breach

Vectors of Data Loss

- ★ Unencrypted Devices
- ★ Cloud Storage Devices
- ★ Removable Media
- ★ Hard Copy
- ★ Improper Access Control
- ★ Email
- ★ Social Networking

BYOD (Bring Your Own Device)

- * BYOD is the emerging trend of employees using their personal devices, like smartphones, tablets, laptops etc, to remotely access any organizational network to carry out office work.
- * Employees can thus access official mail on their smartphone, connect to office and work using their laptop even while they are traveling and use tablets to be part of conferences that happen at their office when they are away.
- * BYOD is important today since employees would want to deliver their best in today's competitive world and companies too would want to make the most of the manpower they have at hand.

BYOD Benefits

- * Boosts productivity: Employees can always work by accessing work using their personal devices and they can even check emails and update presentations while on vacation or while traveling back home.
- * Employees work with devices that they are more comfortable with and are hence happier when they work in places where BYOD is encouraged.
- * The money that needs to be invested on buying hardware, software etc. can be utilized for other things even as employees use their own personal devices for work. Thus SMBs can benefit out of BYOD in a very direct manner.
- * BYOD helps companies stay abreast of changing technology as employees using personal devices for work would stay up-to-date as regards technology and would use the same for the company as well.

BYOD Drawbacks

- ★ The security threats arise due to the increased number of people who would be accessing a company's data using other devices and also due to the fact that malware could get in through any BYOD device that isn't properly secured.
- ★ Company files and data, which are free to be accessed by employees using their personal devices, could also end up in wrong hands. Such data can be easily seen or stolen by outsiders with malicious intentions.
- ★ BYOD devices might also get stolen or they may get lost, which would also cause data breaches.
- ★ The IT departments in companies where BYOD is practiced would have to undergo tremendous pressure support, managing and securing all BYOD devices.

COPE (Corporate-Owned, Personally Enabled)

- * COPE is a business model in which an organization provides its employees with mobile computing devices and allows the employees to use them as if they were personally-owned notebook computers, tablets or smartphones.
- * The COPE model provide the organization with greater power to protect the organization's data both technically and legally.
- * Corporate-owned device policies provide several benefits, such as:
- * The ability to actively manage and control if and when a device can access particular apps, sites, services, networks and solutions.
- * The opportunity to wipe a device of any corporate data when an employee loses his or her device or parts ways with the organization.
- * The chance to incorporate controls on the device that determine how applications, networks and IT systems can be utilized remotely, and whether specific information can be retrieved in certain scenarios.

Security measures for COPE/BYOD

Mobile Device Management (MDM) features secure, monitor, and manage mobile devices, including corporate-owned devices and employee-owned devices.

- * Data Encryption
- * PIN enforcement / Strong Authentication Mechanisms
- * Remote Date Wipe of stolen/misplaced devices
- * Data Loss Prevention (DLP) options
- * Jailbreak/Root detection
- * Remotely locating devices
- * Security assessments (Vulnerability assessments/ Pen testing/ Audits)

The Hacker

Hacker is a common term used to describe a network attacker.

However, the term “hacker” has a variety of meanings:

- ★ A clever programmer capable of developing new programs and coding changes to existing programs to make them more efficient.
- ★ A network professional that uses sophisticated programming skills to ensure that networks are not vulnerable to attack.
- ★ A person who tries to gain unauthorized access to devices on the Internet.
- ★ Individuals who run programs to prevent or slow network access to a large number of users, or corrupt or wipe out data on servers.

White Hat Hackers

- * Ethical Hackers Who use their hacking skills for good, ethical and legal purposes
- * May perform Security assessments such as vulnerability assessment penetration tests to discover vulnerabilities.
- * Security vulnerabilities are reported to developers for them to fix before the vulnerabilities can be exploited.
- * Some organizations award prizes or bounties to white hat hackers when they report vulnerabilities

Gray Hat Hackers

These are the individuals who commit crimes and do arguably unethical things, but not for personal gain or cause serious damage.

Example:

- * Someone who compromise a system without permission and then disclose the vulnerabilities publically.
- * However, by publicizing a vulnerability, the gray hat hacker may give other hackers the opportunity to exploit it.

Black Hat Hackers

- * These are unethical criminals who violate computer and network security for personal gain or for malicious reasons.
- * Black hat hackers exploit vulnerabilities to compromise computer and network systems.

Modern Hacking Titles

- * Script Kiddies
- * Vulnerability Brokers
- * Cyber Criminals
- * Hacktivists
- * State-Sponsored Hackers

Script Kiddies

- * Inexperienced hackers running existing scripts, tools and exploits developed by skillful hackers to cause harm but typically not for profit.
- * It is generally assumed that most script kiddies are juveniles who lack the ability to write sophisticated programs or exploits on their own
- * Their objective is to try to impress their friends or gain credit in computer-enthusiast communities.
- * However, the term does not relate to the actual age of the participant.

Vulnerability Brokers

- * They are usually gray hat hackers who attempt to discover exploits and report them to vendors, sometime for prize or rewards.

Cyber Criminals

- * Cyber criminals are black hat hackers with the motive to make money using any means necessary.
- * Self employed (working independently) or working for criminal organizations.
- * It is estimated that globally, cyber criminals steal billions of dollars from consumers and businesses.
- * Cyber criminals operate in an underground economy where they buy, sell, and trade attack toolkits, zero day exploit code, botnet services, banking Trojans, keyloggers, and much more.
- * They also buy and sell the private information and intellectual property they steal from victims.
- * Cyber criminals target small businesses and consumers, as well as large enterprises and industry verticals.

Hacktivists

- * Grey hat hackers who rally and protest against different social and political ideas.
- * Hacktivists do not hack for profit, they hack for attention.
- * Hacktivists publically protest against organization or governments by posting articles, videos. Leaking sensitive information and performing distributed denial of service attacks.

Examples of hacktivist groups

- Anonymous Hackers
- Syrian Electronic Army.

State-Sponsored Hackers

- * These are government-funded and guided attackers.
- * State-sponsored hackers create advanced and customized attack code, often using previously undiscovered software vulnerabilities, Steal government secrets , gather intelligence and sabotage networks and systems.
- * Their targets are foreign governments, terrorist groups and corporations.
- * Most countries in the world participate to some degree in state-sponsored hacking.
- * Nations hire the best talent to create the most advanced and stealthy threats.
- * **An example :** Stuxnet malware that was created to damage Iran's nuclear enrichment capabilities.

IE2022 – Introduction to Cyber Security

Lecture - 05

Cryptography I

Mr. Amila Senarathne

Cryptography I

- * Reading Assignment
 - CCNA Security Curriculum, Chapter 7: Cryptographic Systems
- * Supplementary text
 - W. Stallings and L. Brown, “Computer Security, Principles and Practice, 2nd edition, Pearson, 2012, Chapter 2 :Cryptographic Tools.

Topics to be discussed

- * Cryptographic Services
- * History of cryptography
- * Substitution and Transposition Ciphers
- * Introduction to Symmetric and Asymmetric Encryption Algorithms
- * One-time pad
- * Cryptanalysis
- * Cryptology

Cryptographic Services

Authentication, Integrity, and Confidentiality Cont.

- * Secure communications necessitates three primary objectives:
- * **Authentication** - Guarantees that the message is not a forgery and does actually come from whom it states.
- * **Integrity** - Guarantees that no one intercepted the message and altered it; similar to a checksum function in a frame.
- * **Confidentiality** - Guarantees that if the message is captured, it cannot be deciphered.



Authentication



Integrity



Confidentiality

Authentication

- * Authentication guarantees that the message:
 - Is not a forgery.
 - Does actually come from who it states it comes from.
- * Authentication is similar to a secure PIN for banking at an ATM.
 - The PIN should only be known to the user and the financial institution.
 - The PIN is a shared secret that helps protect against forgeries.

Entering an ATM Authentication PIN



Authentication Cont.

- * Data nonrepudiation is a similar service that allows the sender of a message to be uniquely identified.
- * This means that a sender/device cannot deny having been the source of that message. It cannot repudiate, or refute, the validity of a message sent.

Data Integrity

- * Data integrity ensures that messages are not altered in transit. The receiver can verify that the received message is identical to the sent message and that no manipulation occurred.
- * European nobility ensured the data integrity by creating a wax seal to close an envelope.
 - The seal was often created using a signet ring.
 - An unbroken seal on an envelope guaranteed the integrity of its contents.
 - It also guaranteed authenticity based on the unique signet ring impression.

An Unbroken Wax Seal Ensures Integrity



Data Confidentiality Cont.

- * Data confidentiality ensures privacy so that only the receiver can read the message.
- * Encryption is the process of scrambling data so that it cannot be read by unauthorized parties.
 - Readable data is called plaintext, or cleartext.
 - Encrypted data is called ciphertext.
 - A key is required to encrypt and decrypt a message. The key is the link between the plaintext and ciphertext.

Encoded Caesar Cipher Message



Creating Ciphertext

- * Authentication, integrity, and confidentiality are components of cryptography.
- * Cryptography is both the practice and the study of hiding information.
- * It has been used for centuries to protect secret documents. Today, modern day cryptographic methods are used in multiple ways to ensure secure communications.



Authentication



Integrity



Confidentiality

Creating Ciphertext Cont.

- * Encryption methods uses a specific algorithm, called a cipher, to encrypt and decrypt messages.
- * A cipher is a series of well-defined steps that can be followed as a procedure when encrypting and decrypting messages.
- * There are several methods of creating cipher text:
 - Transposition
 - Substitution
 - One-time pad

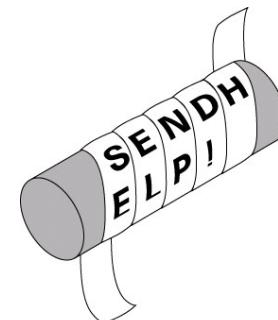
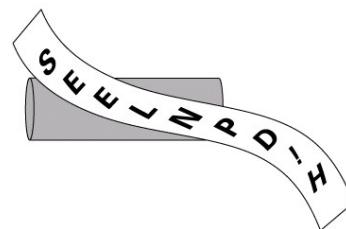
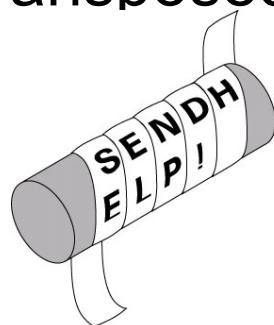
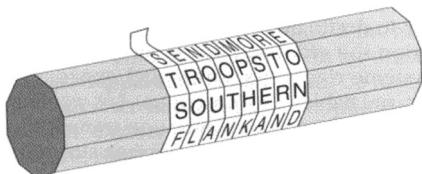
Creating Ciphertext Cont.

- * Cryptography is both the practice and the study of hiding information.
- * Cryptography is used to ensure the protection of data when that data might be exposed to untrusted parties.
- * Cryptographic services are the foundation for many security implementations
- * Over the centuries, various cipher methods, physical devices, and aids have been used to encrypt and decrypt text:
 - Scytale
 - Caesar cipher
 - Vigenère Cipher
 - Jefferson's encryption device
 - German Enigma machine

Creating Ciphertext Cont.

* Scytale

- Earliest cryptography method was used by the Spartans in ancient Greece.
- It is a rod used as an aid for a transposition cipher. The sender and receiver had identical rods (scytale) on which to wrap a transposed messaged.



Creating Ciphertext Cont.

- * Caesar Cipher
- * When Julius Caesar sent messages to his generals, he did not trust his messengers.
- * Caesar encrypted his messages by replacing every letter:
 - A with a D
 - B with an E
 - and so on
- * His generals knew the “shift by 3” rule and could decipher his messages.



Vigenère Cipher

- * Vigenère Cipher
- * In 1586, Frenchman Blaise de Vigenère described a polyalphabetic system of encryption. It became known as the Vigenère Cipher.
- * Based on the Caesar cipher, it encrypted plaintext using a multi-letter key. It is also referred to as an autokey cipher.

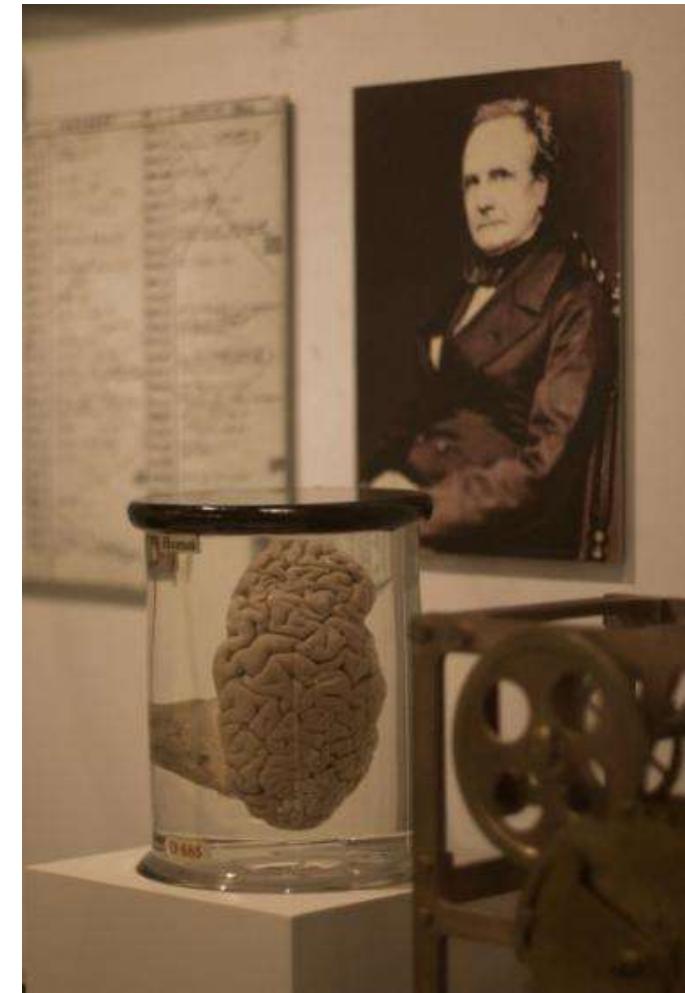


Vigenère Cipher Cont.

	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
A	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
B	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a
C	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b
D	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c
E	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d
F	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e
G	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f
H	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g
I	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h
J	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i
K	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j
L	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k
M	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l
N	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m
O	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n
P	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
Q	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p
R	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q
S	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r
T	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s
U	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t
V	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u
W	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v
X	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w
Y	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x
Z	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y

Note of Interest ...

- * It took 300 years for the Vigenère Cipher to be broken by Englishman Charles Babbage who is known as the father of modern computers.
- * Babbage created the first mechanical computer called the difference engine to calculate numerical tables.
- * He then designed a more complex version called the analytical engine that could use punch cards.
- * He also invented the pilot (cow-catcher).



Creating Ciphertext Cont.

- * Jefferson's Encryption Device

- Thomas Jefferson, the third president of the United States, invented an encryption system that was believed to have been used when he served as secretary of state from 1790 to 1793.



Creating Ciphertext Cont.

- * German Enigma Machine

- Arthur Scherbius invented the Enigma in 1918 and sold it to Germany. It served as a template for the machines that all the major participants in World War II used.
- It was estimated that if 1,000 cryptanalysts tested four keys per minute, all day, everyday, it would take 1.8 billion years to try them all.
- Germany knew their ciphered messages could be intercepted by the allies, but never thought they could be deciphered.



<http://users.telenet.be/d.rijmenants/en/enigma.htm>

Code Talkers

- * During World War II, Japan deciphered every code that the Americans created. A more elaborate coding system was needed. The answer came in the form of the Navajo code talkers.
- * Code talkers were bilingual Navajo speakers specially recruited by the Marines during World War II.
- * Other Native American code talkers were Cherokee, Choctaw, and Comanche soldiers.
- * Not only were there no words in the Navajo language for military terms, the language was unwritten and less than 30 people outside of the Navajo reservations could speak it, and not one of them was Japanese. By the end of the war, more than 400 Navajo Indians were working as code talkers.



Transposition Ciphers

- * In transposition ciphers, no letters are replaced; they are simply rearranged.
- * For example: Spell it backwards.
- * Modern encryption algorithms, such as the Data Encryption Standard (DES) and 3DES, still use transposition as part of the algorithm.

Transposition Ciphers - Rail Fence Cipher

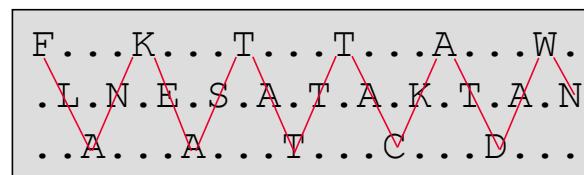
1

Solve the ciphertext.



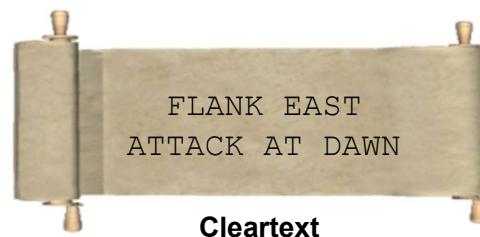
2

Use a rail fence cipher and a key of 3.



3

The cleartext message.

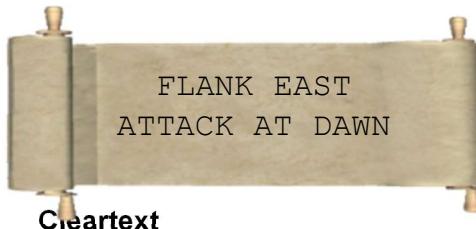


Substitution Ciphers

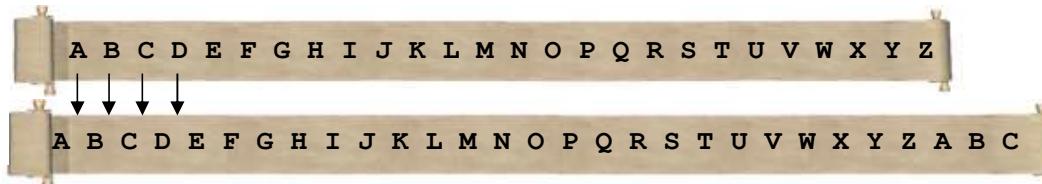
- * Substitution ciphers substitute one letter for another. In their simplest form, substitution ciphers retain the letter frequency of the original message.
- * Examples include:
 - Caesar Cipher
 - Vigenère Cipher

Substitution Ciphers - Encoding using the Caesar Cipher

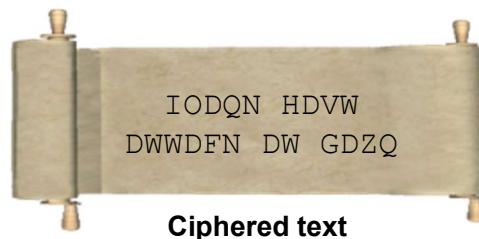
- 1 The cleartext message.



- 2 Encode using a key of 3. Therefore, A becomes a D, B an E, ...

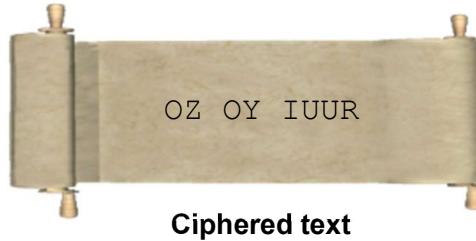


- 3 The encrypted message becomes ...



Decoding

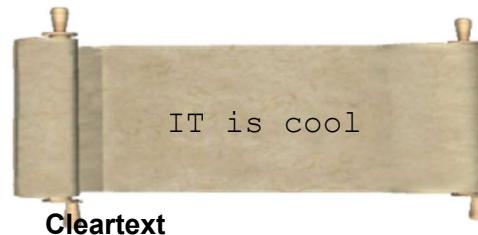
- 1 Solve the ciphertext.



- 2 Use a shift of 6 (ROT6).



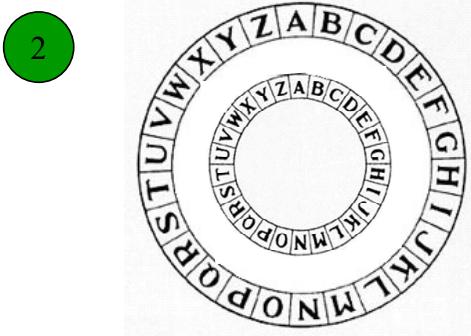
- 3 The clear text message.



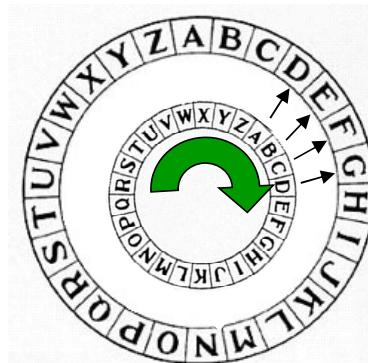
Substitution Ciphers - Caesar Cipher Disk



The cleartext message would be encoded using a key of 3.



Shifting the inner wheel by 3, the A becomes D, B becomes E, and so on.



The ciphertext message appears as follows using a key of 3.

Substitution Ciphers - Vigenère Cipher

- * The Vigenère cipher is based on the Caesar cipher, except that it encrypts text by using a different polyalphabetic key shift for every plaintext letter.
 - The different key shift is identified using a shared key between sender and receiver.
 - The plaintext message can be encrypted and decrypted using the Vigenère Cipher Table.
- * For example:
 - A sender and receiver have a shared secret key: SECRETKEY.
 - The sender then uses the key to encode: FLANK EAST ATTACK AT DAWN.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
A	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
B	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a
C	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	
D	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	
E	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	
F	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	
G	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	
H	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	g	
I	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	h	
J	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i
K	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j
L	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k
M	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l
N	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m
O	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n
P	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
Q	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p
R	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q
S	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r
T	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s
U	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t
V	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u
W	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v
X	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w
Y	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x
Z	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y

F	L	A	N	K	E	A	S	T	A	T	T	A	C	K	A	T	D	A	W	N
S	E	C	R	E	T	K	E	Y	S	E	C	R	E	T	K	E	Y	S	E	C
X	P	C	E	O	X	K	U	R	S	X	V	R	G	D	K	X	B	S	A	P

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
A	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
B	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a
C	c	d	e	f	g	h	i	j	k								s	t	u	v	w	x	y	z	a	b
D	d	e	f	g	h	i	j	k	l								t	u	w	x	y	z	a	b	c	
E	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d
F	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e
G	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f
H	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g
I	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h
J	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i
K	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j
L	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k
M	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l
N	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m
O	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n
P	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
Q	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p
R	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q
S	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r
T	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s
U	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t
V	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u
W	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v
X	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w
Y	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x
Z	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y

To Decrypt

S	E	C	R	E	T	K	E	Y	S	E	C	R	E	T	K	E	Y	S	E	C
X	P	C	E	O	X	K	U	R	S	X	V	R	G	D	K	X	B	S	A	P
F	L	A	N	K	E	A	S	T	A	T	T	A	C	K	A	T	D	A	W	N

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
A	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
B	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a
C	c	d	e	f	g	h	i	j											v	w	x	y	z	a	b	
D	d	e	f	g	h	i	j	k										w	x	y	z	a	b	c		
E	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d
F	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e
G	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f
H	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g
I	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h
J	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i
K	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j
L	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k
M	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l
N	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m
O	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n
P	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
Q	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p
R	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q
S	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r
T	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s
U	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t
V	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u
W	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v
X	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w
Y	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x
Z	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y

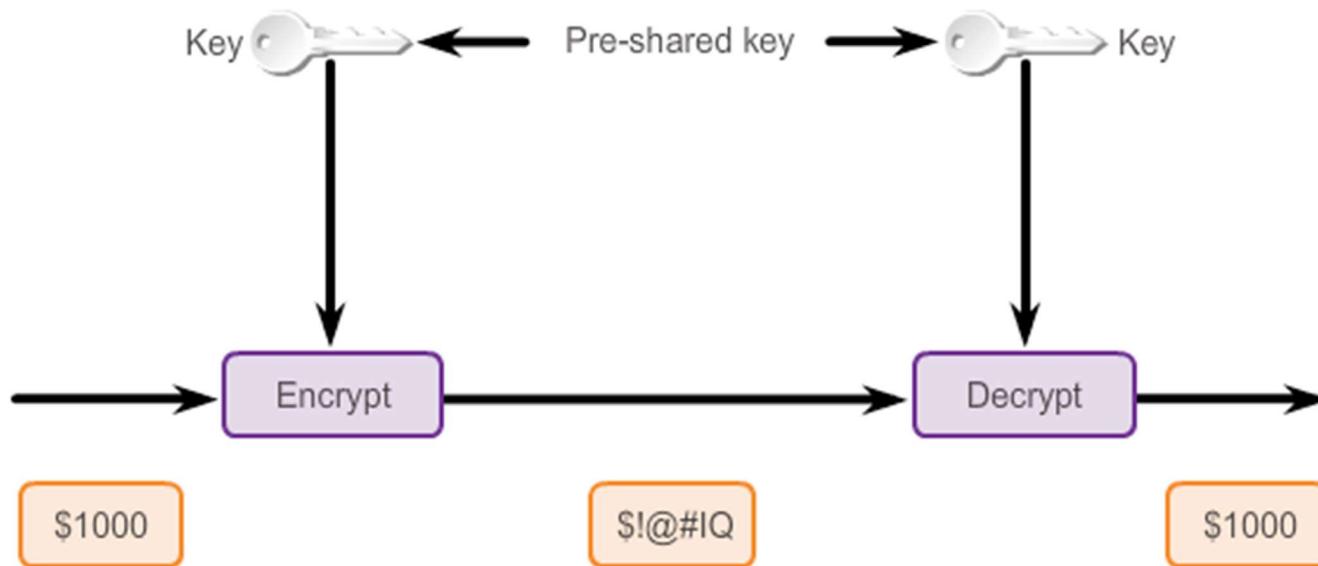
Decrypt the following

T	C	P	I	P	T	C	P	I	P	T	C	P	I	P	T	C	P	I	P	T						
V	E	C	I	H	X	E	J	Z	X	M	A															
C	C	N	A	S	E	C	U	R	I	T	Y															



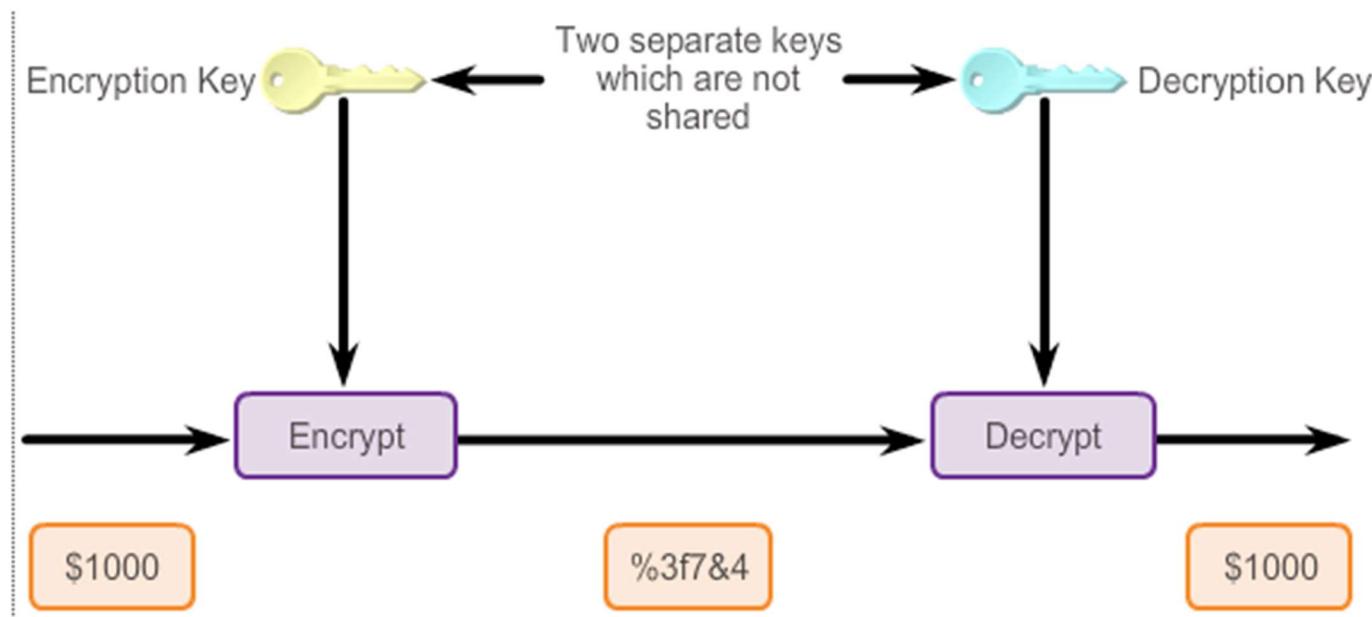
Symmetric Encryption Algorithms

- * Symmetric encryption algorithms characteristics include:
 - Symmetric encryption algorithms are best known as shared-secret key algorithms.
 - A sender and receiver must share a secret key.



Asymmetric Encryption Algorithms Cont.

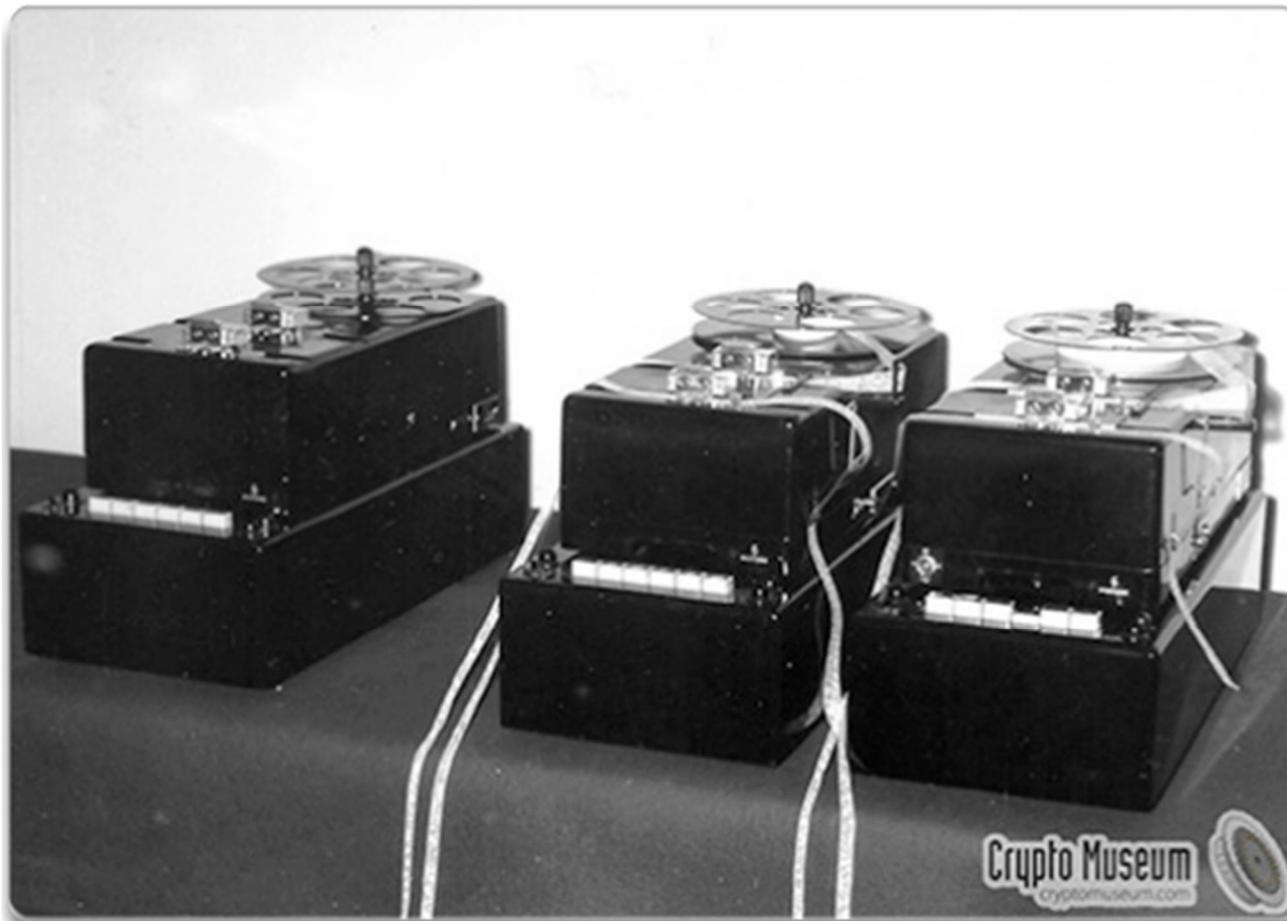
- * Asymmetric encryption algorithms characteristics include:
 - Asymmetric encryption algorithms are best known as public key algorithms.
 - A sender and receiver do not share a secret key.



One-Time Pad Ciphers

- * In 1917, Gilbert Vernam, an AT&T Bell Labs engineer, invented and patented the stream cipher and later co-invented the one-time pad cipher.
 - Vernam proposed a teletype cipher in which a prepared key consisting of an arbitrarily long, non-repeating sequence of numbers was kept on paper tape.
 - It was then combined character by character with the plaintext message to produce the ciphertext.
 - To decipher the ciphertext, the same paper tape key was again combined character by character, producing the plaintext.
- * Each tape was used only once,; hence the name one-time pad. As long as the key tape does not repeat or is not reused, this type of cipher is immune to cryptanalytic attack, because the available ciphertext does not display the pattern of the key.

One-Time Pad Ciphers



Crypto Museum
cryptomuseum.com

One-Time Pad Ciphers Cont.

- * Several difficulties are inherent in using one-time pads in the real world.
 - Key distribution is challenging.
 - Creating random data is challenging and if a key is used more than once, it becomes easier to break.
- * Computers, because they have a mathematical foundation, are incapable of creating true random data.
- * RC4 is a one-time pad cipher that is widely used on the Internet. However, because the key is generated by a computer, it is not truly random.

Cracking Code

Cryptanalysis

- * The practice and study of determining the meaning of encrypted information (cracking the code), without access key/s.
- * Been around since cryptography.



Methods for Cracking Code

- * Brute-Force Method
- * Ciphertext-Only Method
- * Known-Plaintext Method
- * Chosen-Plaintext Method
- * Chosen-Ciphertext Method
- * Meet-in-the-Middle Method

Methods for Cracking Code - Brute-Force Attack

- * An attacker tries every possible key with the decryption algorithm knowing that eventually one of them will work. All encryption algorithms are vulnerable to this attack.
- * The objective of modern cryptographers is to have a keyspace large enough that it takes too much time (money) to accomplish a brute-force attack.
- * For example: The best way to crack Caesar cipher-encrypted code is to use brute force.
 - There are only 25 possible rotations.
 - Therefore, it is not a big effort to try all possible rotations and see which one returns something that makes sense.

Methods for Cracking Code - Brute-Force Attack

- * On average, a brute-force attack succeeds about 50 percent of the way through the keyspace, which is the set of all possible keys.
- * A DES cracking machine recovered a 56-bit DES key in 22 hours using brute force.
- * It is estimated it would take 149 trillion years to crack an AES key using the same method.



Methods for Cracking Code - Ciphertext-Only Attack

- * An attacker has:
- * The ciphertext of several messages, all of which have been encrypted using the same encryption algorithm, but the attacker has no knowledge of the underlying plaintext.
- * The attacker could use statistical analysis to deduce the key.
- * These kinds of attacks are no longer practical, because modern algorithms produce pseudorandom output that is resistant to statistical analysis.

Methods for Cracking Code - Known-Plaintext Attack

- * An attacker has:
 - Access to the ciphertext of several messages.
 - Knowledge (underlying protocol, file type, or some characteristic strings) about the plaintext underlying that ciphertext.
- * The attacker uses a brute-force attack to try keys until decryption with the correct key produces a meaningful result.
- * Modern algorithms with enormous keyspaces make it unlikely for this attack to succeed, because, on average, an attacker must search through at least half of the keyspace to be successful.

Methods for Cracking Code - Chosen-Plaintext Attack

- * An attacker chooses which data the encryption device encrypts and observes the ciphertext output. A chosen-plaintext attack is more powerful than a known-plaintext attack, because the chosen plaintext might yield more information about the key.
- * This attack is not very practical, because it is often difficult or impossible to capture both the ciphertext and plaintext.

Methods for Cracking Code - Chosen-Ciphertext Attack

- * An attacker chooses different ciphertext to be decrypted and has access to the decrypted plaintext. With the pair, the attacker can search through the keyspace and determine which key decrypts the chosen ciphertext in the captured plaintext.
- * This attack is analogous to the chosen-plaintext attack.
 - Like the chosen-plaintext attack, this attack is not very practical.
 - Again, it is difficult or impossible for the attacker to capture both the ciphertext and plaintext.

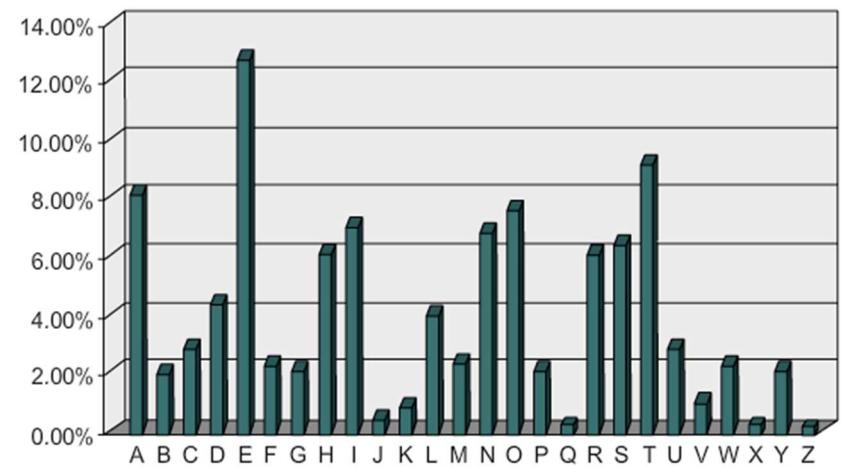
Methods for Cracking Code - Meet-in-the-Middle

- * The meet-in-the-middle attack is a known plaintext attack.
- * The attacker knows that a portion of the plaintext and the corresponding ciphertext.
- * The plaintext is encrypted with every possible key, and the results are stored. The ciphertext is then decrypted using every key, until one of the results matches one of the stored values.

Cracking Code Example

- * The best way to crack the code is to use brute force.
- * Because there are only 25 possible rotations, the effort is relatively small to try all possible rotations and see which one returns something that makes sense.
- * A more scientific approach is to use the fact that some characters in the English alphabet are used more often than others.
- * This method is called frequency analysis.

Deciphering Using Frequency Analysis



The graph outlines the frequency of letters in the English language.

For example, the letters E, T and A are the most popular.

Cracking Code Example- Frequency Analysis Method

- * The English alphabet is used more often than others.
 - E, T, and A are the most popular letters.
 - J, Q, X, and Z are the least popular.
- * Caesar ciphered message:
 - The letter D appears six times.
 - The letter W appears four times.
 - Therefore, it is probable that they represent the more popular letters.
- * In this case, D represents the letter A, and W represents the letter T.

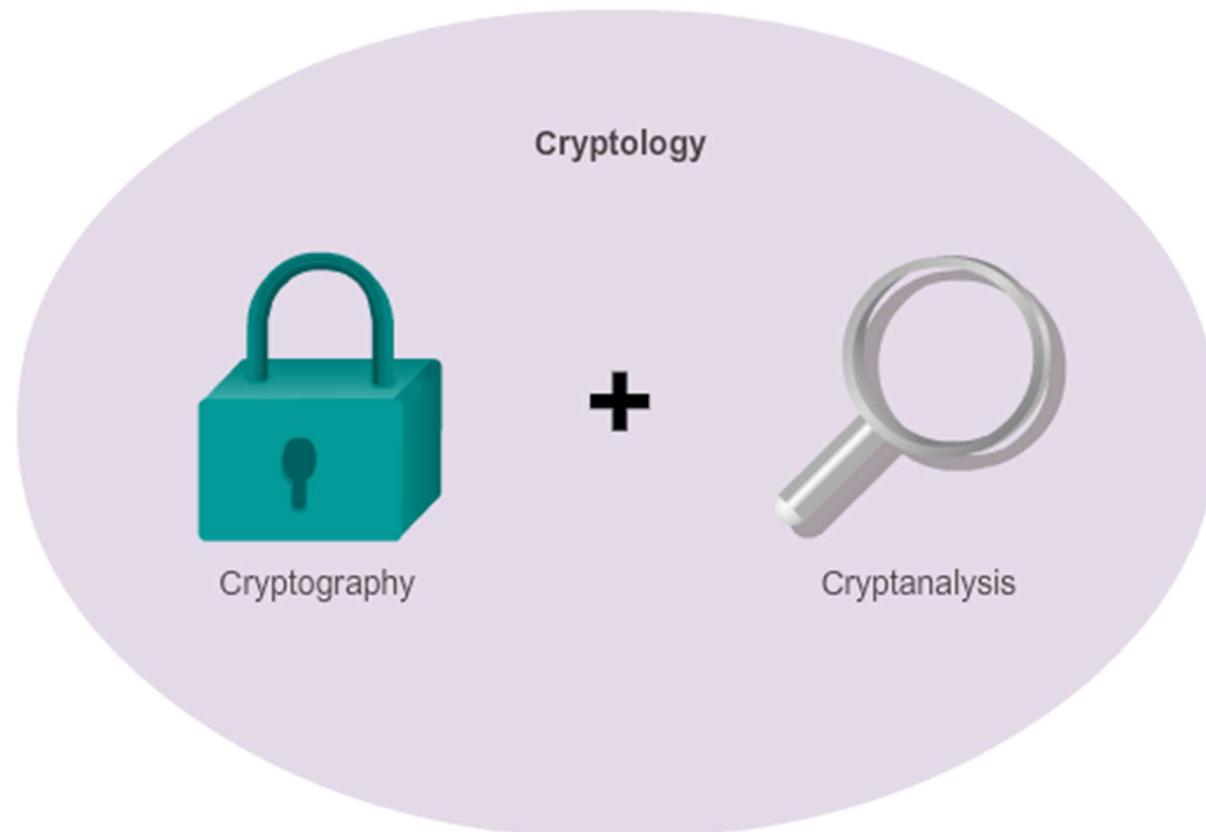


Ciphered Text



Cleartext

Making and Breaking Secret Codes



Making and Breaking Secret Codes Cont.

- * Cryptology is the science of making and breaking secret codes. It combines cryptography (development and use of codes), and cryptanalysis, (breaking of those codes).
- * There is a symbiotic relationship between the two disciplines, because each makes the other one better.
 - National security organizations employ members of both disciplines and put them to work against each other.
- * There have been times when one of the disciplines has been ahead of the other.
 - Currently, it is believed that cryptographers have the edge.

Cryptanalysis

- * Ironically, it is impossible to prove an algorithm secure. It can only be proven that it is not vulnerable to known cryptanalytic attacks.
- * There is a need for mathematicians, scholars, and security forensic experts to keep trying to break the encryption methods.
- * Cryptanalysis are most used employed by:
 - Governments in military and diplomatic surveillance.
 - Enterprises in testing the strength of security procedures.

Sample Cryptanalysis Job Description

Cryptanalysis
National Security Agency | Fort Meade, MD

Job Description

Cryptanalysis is one of the core technical disciplines necessary for the NSA to accomplish its mission and provide critical intelligence to the nation's leaders. In an ever-changing global environment, the need for Cryptanalysts will remain constant.

Traditionally, Cryptanalysis is the art and science of solving cryptograms (writings in cipher or code) or cryptographic systems (devices for enciphering and deciphering) through analysis without prior knowledge of the encryption method. In a code, a word or phrase is replaced with another word, number, or symbol. In a cipher, each letter is replaced with another letter, number or symbol. Using known techniques and imagination, a Cryptanalyst systematically identifies basic elements in a cipher code that may lead to its solution. Modern Cryptanalysis includes analysis of any type of hidden information, whether a traditional cipher or a telecommunication protocol.

ANSWERING THE TOUGH QUESTIONS:

Cryptanalysts utilize mathematics, computer programming, engineering, and language skills as well as new technologies and creativity to solve tomorrow's problems today. That's why the NSA is looking for people who are intelligent and imaginative, and who can contribute original ideas to the solution of complex challenges. Cryptanalysts must communicate clearly, concentrate long and hard on difficult problems, and not be discouraged if success is elusive. No specific major is targeted for Cryptanalysis; the NSA hires people with technical and non-technical degrees, ranging from mathematics to music, engineering to history, and computer programming to chemistry.

The Secret Is in the Keys

- * Authentication, integrity, and data confidentiality are implemented in many ways using various protocols and algorithms. Choice depends on the security level required in the security policy.

	Integrity	Authentication	Confidentiality
Common cryptographic hashes, protocols, and algorithms	MD5 (weaker) SHA (stronger)	HMAC-MD5 HMAC-SHA-1 RSA and DSA	DES (weaker) 3DES AES (stronger)

The Secret Is in the Keys Cont.

- * Security of encryption lies in the secrecy of the keys, not the algorithm.
- * Old encryption algorithms were based on the secrecy of the algorithm to achieve confidentiality.
- * With modern technology, algorithm secrecy no longer matters since reverse engineering is often simple; therefore, public-domain algorithms are often used. Now, successful decryption requires knowledge of the keys.
- * How can the keys be kept secret?

IE2022 – Introduction to Cyber Security

Lecture - 06

Cryptography II- Hash Functions and Key Management

Mr. Amila Senarathne

Cryptographic Hash Functions and Key Management

- * Reading Assignment
 - CCNA Security Curriculum, Chapter 7: Cryptographic Systems
- * Supplementary text
 - W. Stallings and L. Brown, “Computer Security, Principles and Practice, 2nd edition, Pearson, 2012, Chapter 2 :Cryptographic Tools.

Topics to be discussed

- * Cryptographic Hash Function
 - Cryptographic Hash Function Properties
 - MD5 and SHA
- * Keyed-Hash Message Authentication Code (HMAC)
- * Characteristics of Key Management

Cryptology - The Secret Is in the Keys

- * Authentication, integrity, and data confidentiality are implemented in many ways using various protocols and algorithms. Choice depends on the security level required in the security policy.

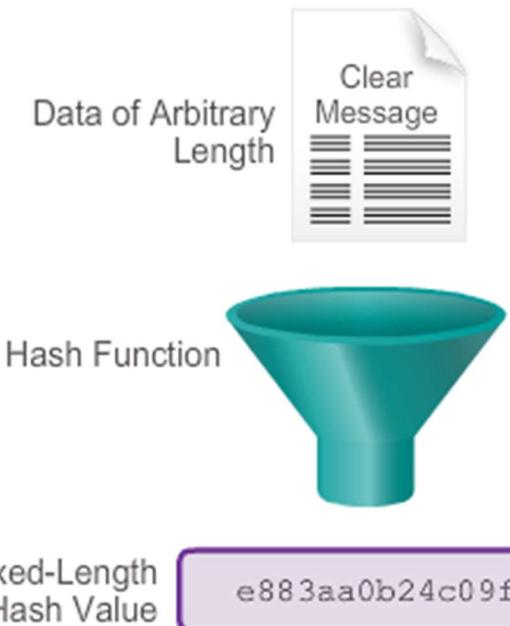
	Integrity	Authentication	Confidentiality
Common cryptographic hashes, protocols, and algorithms	MD5 (weaker) SHA (stronger)	HMAC-MD5 HMAC-SHA-1 RSA and DSA	DES (weaker) 3DES AES (stronger)

BASIC INTEGRITY AND AUTHENTICITY

Cryptographic Hash Functions

Creating a Hash

- * A hash function takes binary data (message), and produces a condensed representation, called a hash. The hash is also commonly called a Hash value, Message digest, or Digital fingerprint.
- * Hashing is based on a one-way mathematical function that is relatively easy to compute, but significantly harder to reverse.
- * Hashing is designed to verify and ensure:
 - Data integrity
 - Authentication

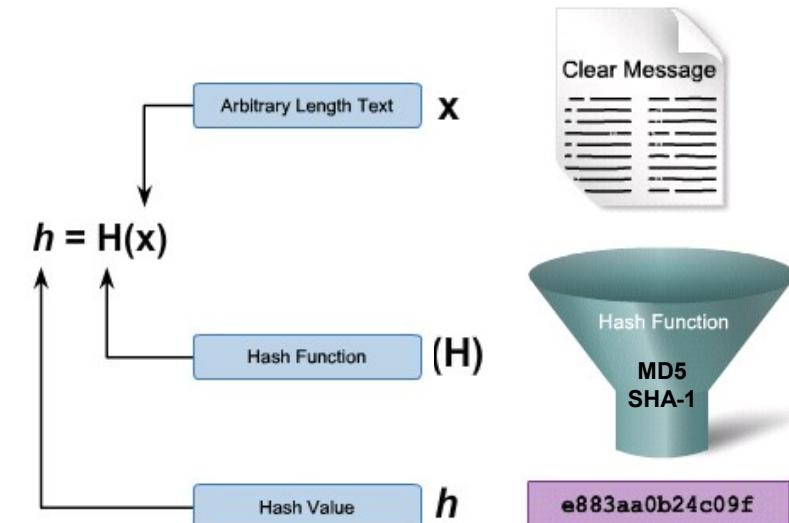


Cryptographic Hash Functions

- * Cryptographic hash function is applied in many different situations:
 - * To provide proof of authenticity when it is used with a symmetric secret authentication key, such as IP Security (IPsec) or routing protocol authentication.
 - * To provide authentication by generating one-time and one-way responses to challenges in authentication protocols, such as the PPP CHAP.
 - * To provide a message integrity check proof, such as those accepted when accessing a secure site using a browser.
 - * To confirm that a downloaded file (e.g., Cisco IOS images) has not been altered.

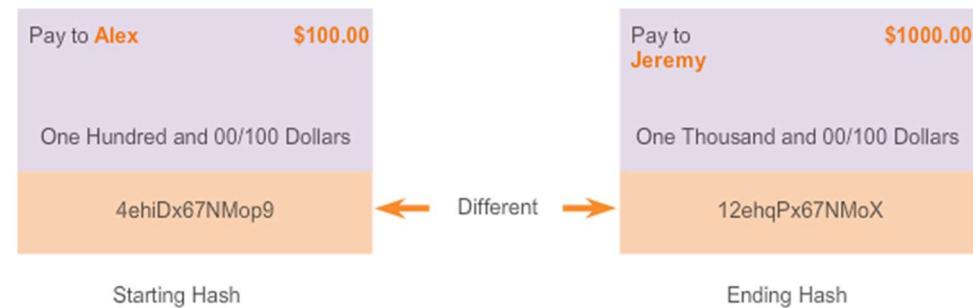
Cryptographic Hash Function Properties

- * Take an arbitrarily length of clear text data to be hashed.
- * Put it through a hash function.
- * It produces a fixed length message digest (hash value).
- * $H(x)$ is:
 - Relatively easy to compute for any given x .
 - One way and not reversible.
- * If a hash function is hard to invert, it is considered a one-way hash.



Well-Known Hash Functions

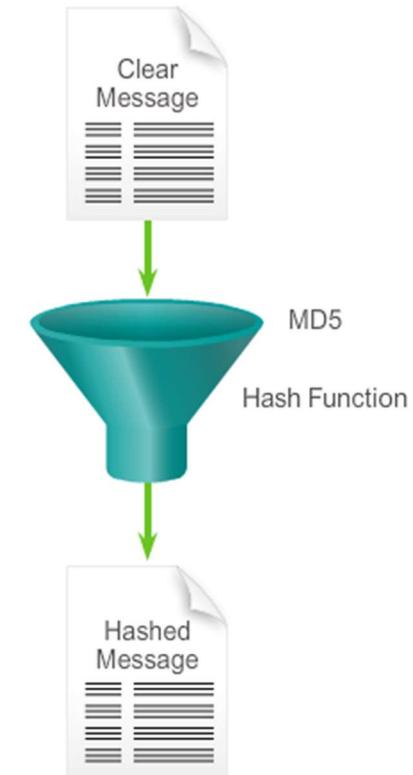
- * Hash functions are helpful when ensuring data is not changed accidentally, such as by a communication error.
- * Hash functions cannot be used to guard against deliberate changes.
- * There is no unique identifying information from the sender in the hashing procedure, so anyone can compute a hash for any data, as long as they have the correct hash function.
- * Hashing is vulnerable to man-in-the-middle attacks and does not provide security to transmitted data.
- * Two well-known hash functions are:
 - MD5 with 128-bit digests
 - SHA-256 with 256-bit digests



Message Digest 5 Algorithm

MD5 Hashing Algorithm

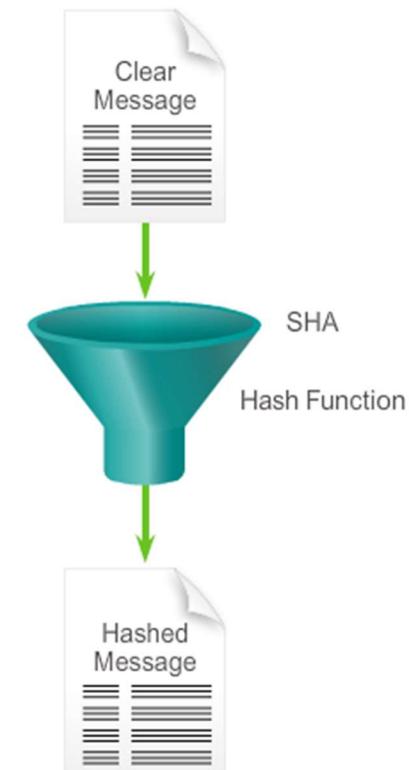
- * MD5 algorithm is a hashing algorithm that was developed by Ron Rivest.
- * Used in a variety of Internet applications today.
- * A one-way function that makes it easy to compute a hash from the given input data, but makes it unfeasible to compute input data given only a hash value.



Secure Hash Algorithm

- * U.S. National Institute of Standards and Technology (NIST) developed SHA, the algorithm specified in the Secure Hash Standard (SHS).
- * SHA-1, published in 1994, corrected an unpublished flaw in SHA.
- * SHA design is very similar to the MD4 and MD5 hash functions that Ron Rivest developed.

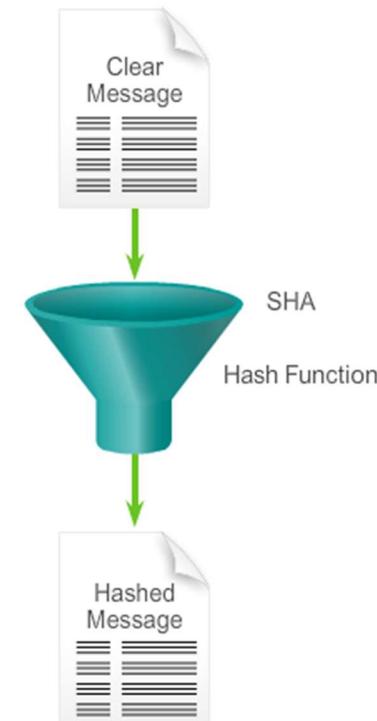
SHA Hashing Algorithm



Secure Hash Algorithm

- * SHA-1 algorithm takes a message of less than 2^{64} bits in length and produces a 160-bit message digest.
- * Slightly slower than MD5, but the larger message digest makes it more secure against brute-force collision and inversion attacks.
- * NIST published four additional hash functions in the SHA family, each with longer digests:
 - SHA-224 (224 bit)
 - SHA-256 (256 bit)
 - SHA-384 (384 bit)
 - SHA-512 (512 bit)

SHA Hashing Algorithm



MD5 Versus SHA-1

MD5	SHA-1
Based on MD4	Based on MD4
Computation involves 64 steps	Computation involves 80 steps
Algorithm must process a 128-bit buffer	Algorithm must process a 160-bit buffer
Faster	Slower
Less Secure	More secure

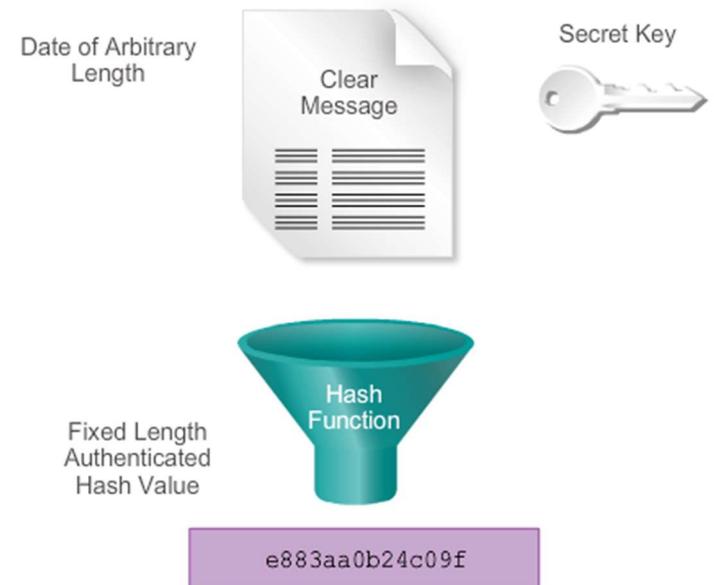
Keyed-Hash Message Authentication Code

- * HMAC (or K HMAC) is a message authentication code (MAC) that is calculated using a hash function and a secret key.
 - HMACs use an additional secret key as input to the hash function adding authentication to integrity assurance.
 - Hash functions are the basis of the protection mechanism of HMACs.
 - The output of the hash function now depends on the input data and the secret key.
- * Authenticity is guaranteed, because only the sender and the receiver know the secret key.
 - Only they can compute the digest of an HMAC function.
 - This characteristic defeats man-in-the-middle attacks and provides authentication of the data origin.

Keyed-Hash Message Authentication Code

- * The cryptographic strength of the HMAC depends on the:
 - Cryptographic strength of the underlying hash function.
 - Size and quality of the key.
 - Size of the hash output length in bits.
- * Cisco technologies use two well-known HMAC functions:
 - Keyed MD5 or HMAC-MD5 is based on the MD5 hashing algorithm.
 - Keyed SHA-1 or HMAC-SHA-1 is based on the SHA-1 hashing algorithm.

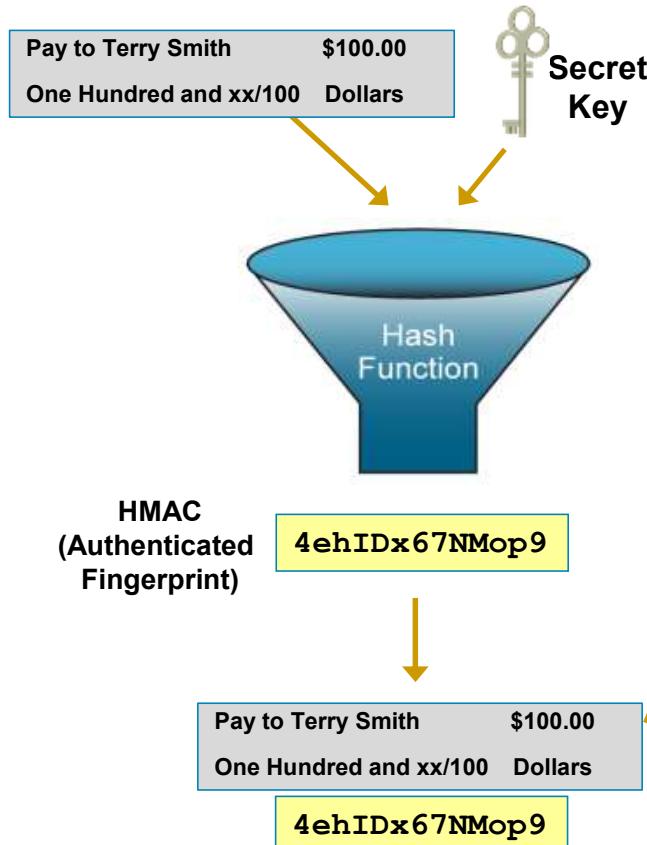
HMAC Hashing Algorithm



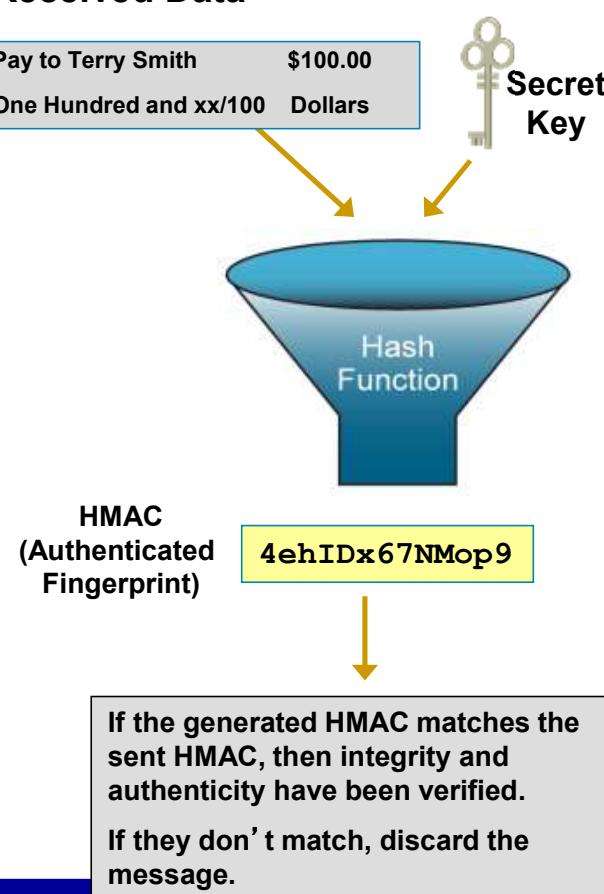
The same procedure is used for generation and verification of secure fingerprints.

HMAC Operation

Data



Received Data



Characteristics of Key Management

- * Often considered the most difficult part of designing a cryptosystem.
- * There are several essential characteristics of key management to consider:
 - Key generation
 - Key verification
 - Key storage
 - Key exchange
 - Key revocation and destruction

Characteristics of Key Management

- * Key Generation
 - Caesar chose the key of his cipher and the Sender/Receiver chose a shared secret key for the Vigenère cipher.
 - Modern cryptographic system key generation is usually automated.
- * Key Verification
 - Almost all cryptographic algorithms have some weak keys that should not be used (e.g., Caesar cipher ROT 0 or ROT 25).
 - With the help of key verification procedures, these keys can be regenerated if they occur.
- * Key Storage - Modern cryptographic system store keys in memory.

Characteristics of Key Management

- * Key Exchange
 - Key management procedures should provide a secure key exchange mechanism over an untrusted medium.
- * Key Revocation and Destruction
 - Revocation notifies all interested parties that a certain key has been compromised and should no longer be used.
 - Destruction erases old keys in a manner that prevents malicious attackers from recovering them.
- * Two terms that are used to describe keys are:
 - Key size - The measure in bits; also called the key length.
 - Keyspace - This is the number of possibilities that can be generated by a specific key length.

Characteristics of Key Management

- * The key length is the measure in bits and the keyspace is the number of possibilities that can be generated by a specific key length.
- * As key lengths increase, keyspace increases exponentially:
 - 2^2 key = a keyspace of 4
 - 2^3 key = a keyspace of 8
 - 2^4 key = a keyspace of 16
 - 2^{40} key = a keyspace of 1,099,511,627,776

Key Management - The Keyspace

- * Adding one bit to a key doubles the keyspace.
- * For each bit added to the DES key, the attacker would require twice the amount of time to search the keyspace.
- * Longer keys are more secure but are also more resource intensive and can affect throughput.

DES Key Length	Keyspace	# of Possible Keys
56 bit	2^5	72,000,000,000,000,000
57 bit	2^{57}	144,000,000,000,000,000
58 bit	2^{58}	288,000,000,000,000,000
59 bit	2^{59}	576,000,000,000,000,000
60 bit	2^{60}	1,152,000,000,000,000,000

Types of Cryptographic Keys

- * Symmetric keys that can be exchanged between two routers supporting a VPN.
- * Asymmetric keys that used in secure HTTPS applications.
- * Digital signatures that used when connecting to a secure website.
- * Hash keys that used in symmetric and asymmetric key generation, digital signatures, and other types of applications

	Symmetric Key	Asymmetric Key	Digital Signature	Hash
Protection up to 3 years	80	1248	160	160
Protection up to 10 years	96	1776	192	192
Protection up to 20 years	112	2432	224	224
Protection up to 30 years	128	3248	256	256
Protection against quantum computers	256	15424	512	512

Choosing Cryptographic Keys

- * Performance is another issue that can influence the choice of a key length.
- * An administrator must find a good balance between the speed and protective strength of an algorithm.



Shorter keys equal faster processing, but are less secure.



Longer keys equal slower processing, but are more secure.

IE2022 – Introduction to Cyber Security

Lecture - 07

Cryptography III - Symmetric-Key Algorithms

Mr. Amila Senarathne

Cryptographic Hash Functions and Symmetric-Key Algorithms

- * Reading Assignment
 - CCNA Security Curriculum, Chapter 7: Cryptographic Systems
- * Supplementary text
 - W. Stallings and L. Brown, “Computer Security, Principles and Practice, 2nd edition, Pearson, 2012, Chapter 2 :Cryptographic Tools.

Topics to be discussed

- * Symmetric Encryption Algorithms
- * Symmetric Encryption Techniques
 - Block Ciphers
 - Stream Ciphers
- * Choosing an Encryption Algorithm

Cryptology - The Secret Is in the Keys

- * Authentication, integrity, and data confidentiality are implemented in many ways using various protocols and algorithms. Choice depends on the security level required in the security policy.

	Integrity	Authentication	Confidentiality
Common cryptographic hashes, protocols, and algorithms	MD5 (weaker) SHA (stronger)	HMAC-MD5 HMAC-SHA-1 RSA and DSA	DES (weaker) 3DES AES (stronger)

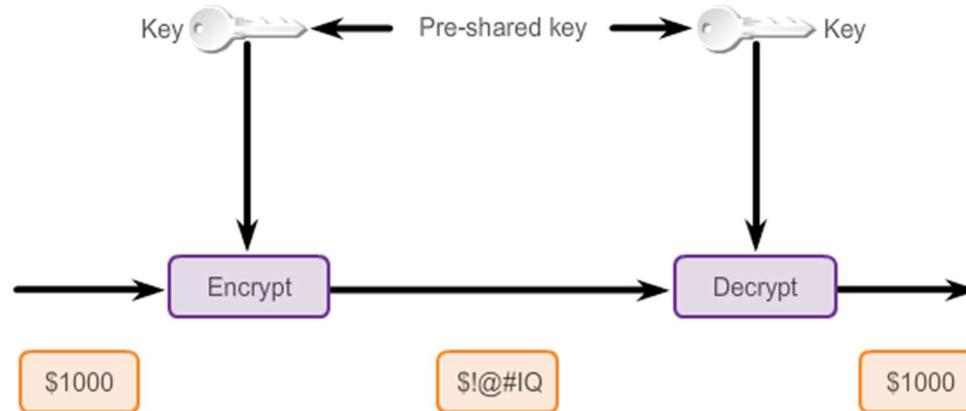
CONFIDENTIALITY

Cryptographic Encryption

- * Cryptographic encryption can provide confidentiality at several layers of the OSI model by incorporating various tools and protocols:
 - Proprietary link-encrypting devices provide data link layer confidentiality.
 - Network layer protocols, such as the IPsec protocol suite, provide network layer confidentiality.
 - Protocols, such as Secure Sockets Layer (SSL) or Transport Layer Security (TLS), provide session layer confidentiality.
 - Secure email, secure database session (Oracle SQL*net), and secure messaging (Lotus Notes sessions) provide application layer confidentiality.

Symmetric Encryption Algorithms

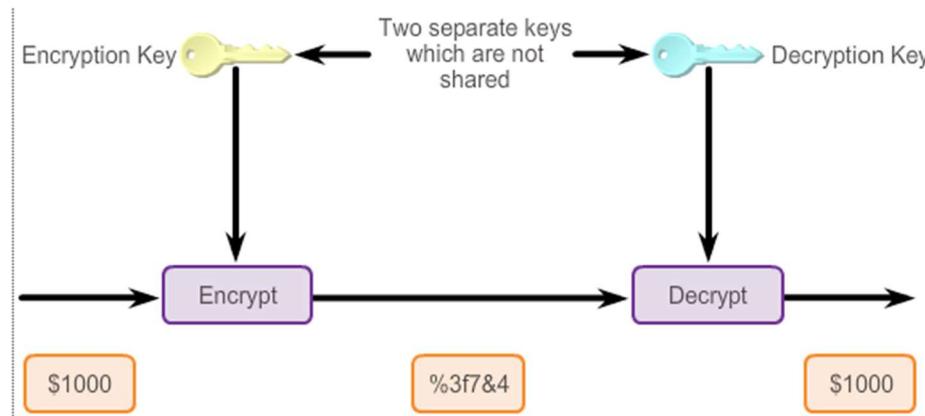
- * Symmetric encryption algorithms characteristics include:
 - Symmetric encryption algorithms are best known as shared-secret key algorithms.
 - The usual key length is 80 to 256 bits.
 - A sender and receiver must share a secret key.
 - They are usually quite fast (wire speed), because these algorithms are based on simple mathematical operations.
 - Examples of symmetric encryption algorithms are DES, 3DES, AES, IDEA, RC2/4/5/6, and Blowfish.



Asymmetric Encryption Algorithms

- * Asymmetric encryption algorithms characteristics include:

- Asymmetric encryption algorithms are best known as public key algorithms.
- The usual key length is 512 to 4,096 bits.
- A sender and receiver do not share a secret key.
- These algorithms are relatively slow, because they are based on difficult computational algorithms.
- Examples: RSA, ElGamal, elliptic curves, and DH.



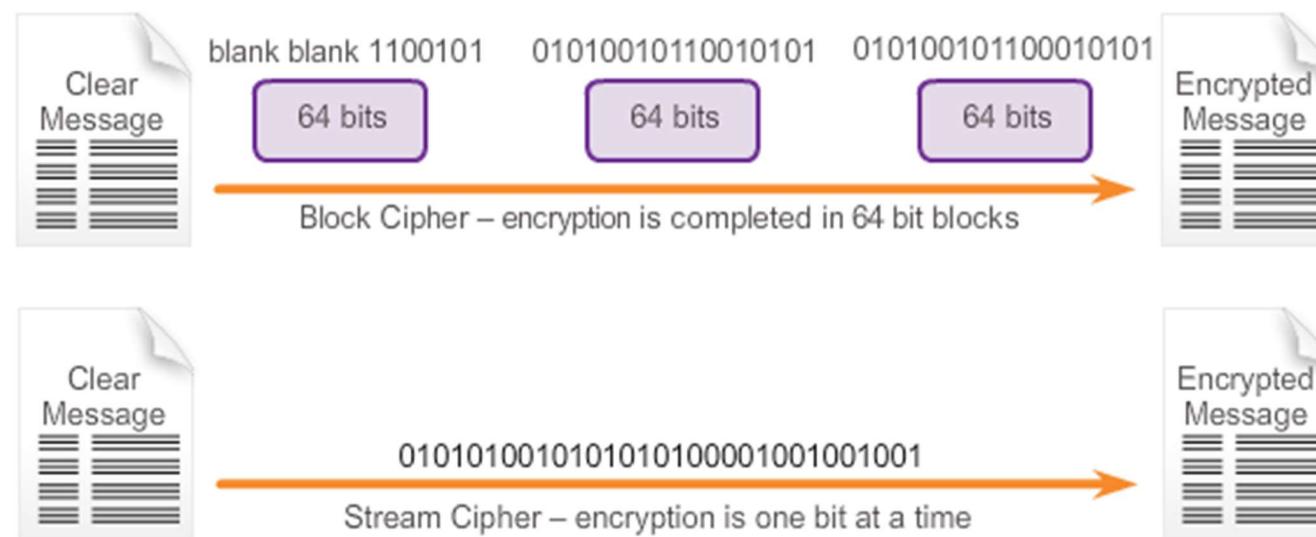
Symmetric Encryption Algorithms

- * Symmetric encryption algorithms, also called shared secret-key algorithms, use the same pre-shared secret key to encrypt and decrypt data. The pre-shared key is known by the sender and receiver before any encrypted communications begins.
- * Because both parties are guarding a shared secret, the encryption algorithms used can have shorter key lengths. Shorter key lengths mean faster execution.
- * For this reason symmetric algorithms are generally much less computationally intensive than asymmetric algorithms.

Symmetric Encryption Algorithm	Key length (in bits)
DES	56
3DES	112 and 168
AES	128, 192, and 256
Software Encryption Algorithm (SEAL)	160
The RC series	RC2 (40 and 64) RC4 (1 to 256) RC5 (0 to 2040) RC6 (128, 192, and 256)

Symmetric Encryption Techniques

- * There are two types of encryption method used:
 - Block Ciphers
 - Stream Ciphers

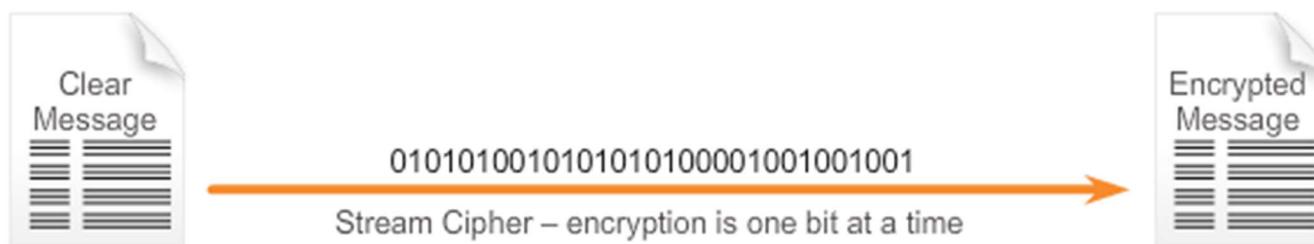


Block Ciphers

- * Block ciphers transform a fixed-length block of plaintext into a common block of ciphertext of 64 or 128 bits.
 - Block size refers to how much data is encrypted at any one time.
 - The key length refers to the size of the encryption key that is used.
 - This ciphertext is decrypted by applying the reverse transformation to the ciphertext block, using the same secret key.
- * Common block ciphers include:
 - DES with a 64-bit block size
 - AES with a 128-bit block size
 - RSA with a variable block size

Stream Ciphers

- * Stream ciphers encrypt plaintext one byte or one bit at a time.
 - Think of it like a block cipher with a block size of one bit.
 - The Vigenère cipher is an example of a stream cipher.
 - Can be much faster than block ciphers, and generally do not increase the message size.
- * Common stream ciphers include:
 - A5 used to encrypt GSM cell phone communications.
 - RC4 cipher.
 - DES can also be used in stream cipher mode.



Choosing an Encryption Algorithm

- * Is the algorithm trusted by the cryptographic community? Algorithms that have been resisting attacks for a number of years are preferred.
- * Does the algorithm adequately protect against brute-force attacks? With the appropriate key lengths, these attacks are usually considered unfeasible.
- * Does the algorithm support variable and long key lengths?
- * Does the algorithm have export or import restrictions?

Choosing an Encryption Algorithm

	DES	3DES	AES
Is the algorithm trusted by the cryptographic community?	Been replaced by 3DES	Yes	Verdict is still out
Does the algorithm adequately protect against brute-force attacks?	No	Yes	Yes

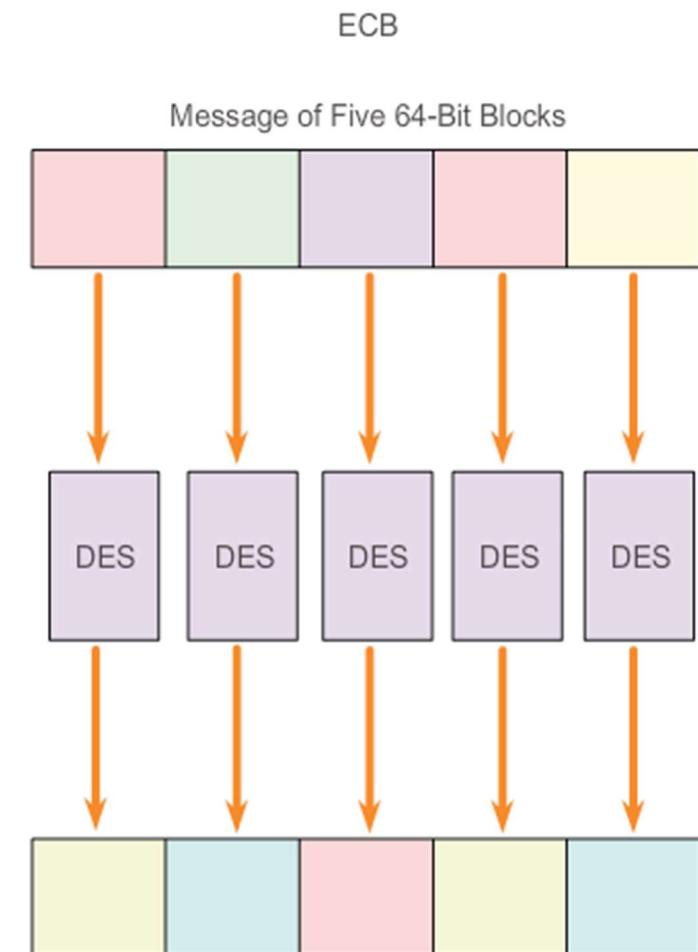
Data Encryption Standard

- * The most popular symmetric encryption standard.
 - Developed by IBM
 - Thought to be unbreakable in the 1970s
 - Shared keys enable the encryption and decryption
- * DES converts blocks of 64-bits of clear text into ciphertext by using an encryption algorithm.
 - The decryption algorithm on the remote end restores ciphertext to clear text.

DES Characteristics	
Description	Data Encryption Standard
Timeline	Standardized 1976
Type of Algorithm	Symmetric
Key size (in bits)	56 bits
Speed	Medium
Time to crack (Assuming a computer could try 255 keys per second)	Days (6.4 days by the COPACABANA machine, a specialized cracking device)
Resource Consumption	Medium

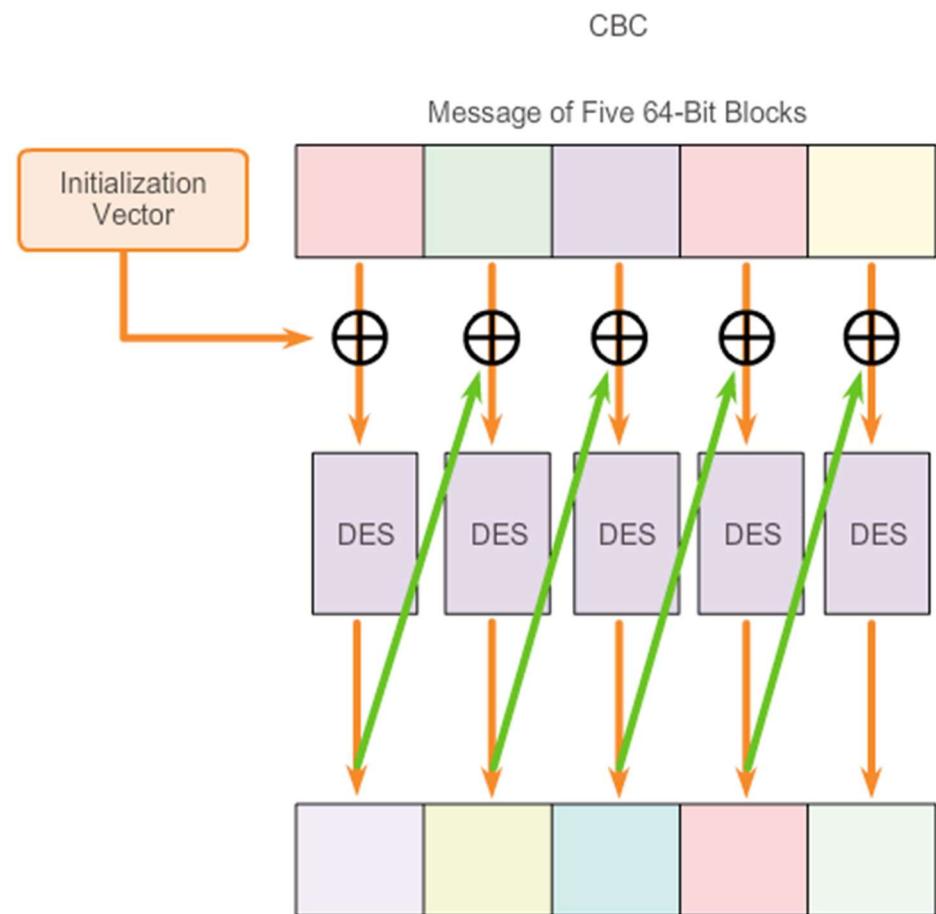
DES Operation - ECB

- * ECB mode serially encrypts each 64-bit plaintext block using the same 56-bit key.
- * If two identical plaintext blocks are encrypted using the same key, their ciphertext blocks are the same.
- * Therefore, an attacker could identify similar or identical traffic flowing through a communications channel.



DES Operation - CBC

- * CBC mode, each 64-bit plaintext block is XORed bitwise with the previous ciphertext block and then is encrypted using the DES key.
- * The encryption of each block depends on previous blocks.
- * Encryption of the same 64-bit plaintext block can result in different ciphertext blocks.



DES Operations Cont.

- * To encrypt or decrypt more than 64 bits of data, DES uses two common stream cipher modes:
 - Cipher feedback (CFB), which is similar to CBC and can encrypt any number of bits, including single bits or single characters.
 - Output feedback (OFB) generates keystream blocks, which are then XORed with the plaintext blocks to get the ciphertext.
- * The cipher uses previous ciphertext and the secret key to generate a pseudo-random stream of bits, which only the secret key can generate.

DES Summary

- * Because of its short key length, DES is considered a good protocol to protect data for a very short time.
 - 3DES is a better choice to protect data, because it has an algorithm that is very trusted and has higher security strength.
- * Recommendations:
 - Change keys frequently to help prevent brute-force attacks.
 - Use a secure channel to communicate the DES key from the sender to the receiver.
 - Consider using DES in CBC mode.
 - Test a key to see if it is a weak key before using it.

3DES - Improving DES with 3DES

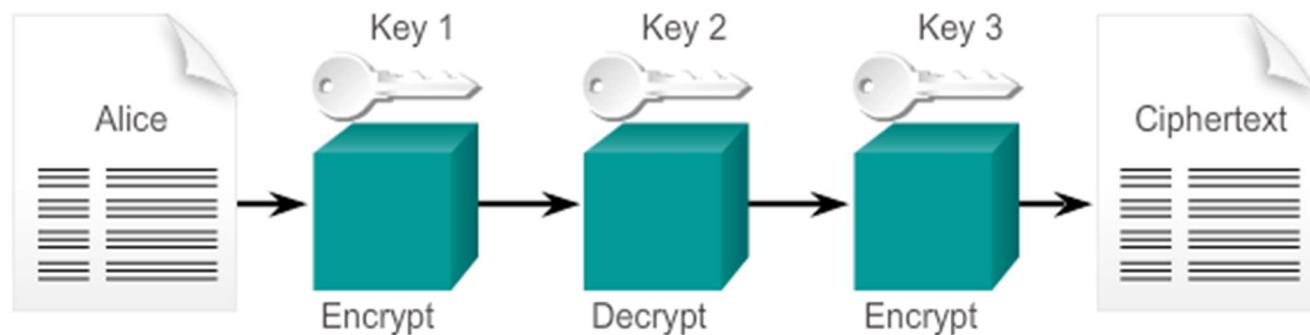
- * 3DES is 256 times stronger than DES.
- * It takes a 64-bit block of data and performs three DES operations in sequence:
 - Encrypts, decrypts, and encrypts.
 - Requires additional processing time.
 - Can use 1, 2, or 3 different keys (when used with only one key, it is the same as DES).
- * 3DES software is subject to U.S. export laws.

3DES - Improving DES with 3DES

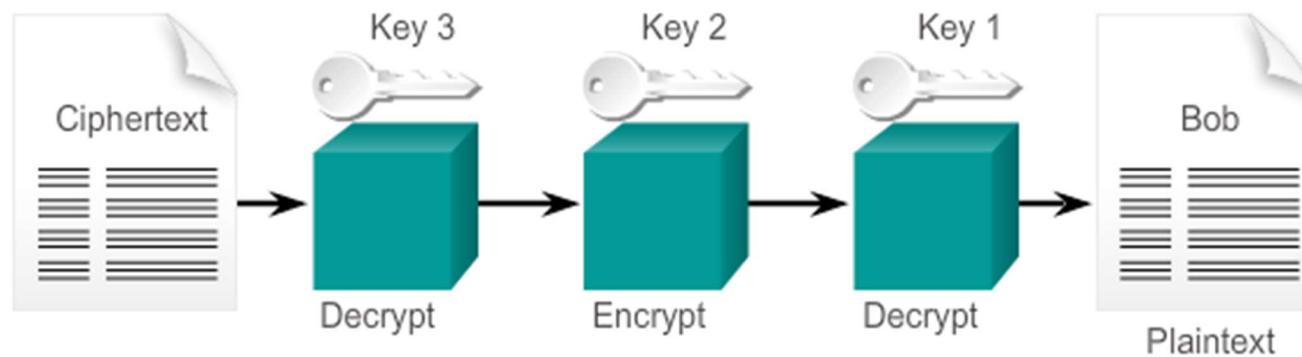
3DES Characteristics	
Description	Triple Data Encryption Standard
Timeline	Standardized 1977
Type of Algorithm	Symmetric
Key size (in bits)	112 and 168 bits
Speed	Low
Time to crack (Assuming a computer could try 255 keys per second)	4.6 Billion years with current technology
Resource Consumption	Medium

3DES - 3DES Operation

3DES Encryption



3DES Decryption



Advanced Encryption Standard (AES)

AES Origins

- * 1997, the AES initiative was announced, and the public was invited to propose encryption schemes to replace DES.
- * After a five-year standardization process in which 15 competing designs were presented and evaluated, the U.S. National Institute of Standards and Technology (NIST) selected the Rijndael block cipher as the AES algorithm..
 - Based on the Rijndael (“Rhine dahl”) algorithm.
 - It uses keys with a length of 128, 192, or 256 bits to encrypt blocks with a length of 128, 192, or 256 bits.
 - All 9 combinations of key length and block length are possible.
- * AES is now available in the latest Cisco router images that have IPsec DES/3DES functionality.

AES Summary

- * AES was selected to replace DES for a number of reasons:
 - The key length of AES makes the key much stronger than DES.
 - AES runs faster than 3DES on comparable hardware.
 - AES is more efficient than DES and 3DES on comparable hardware, usually by a factor of five when it is compared with DES.
 - AES is more suitable for high-throughput, low-latency environments, especially if pure software encryption is used.
- * However, AES is a relatively young algorithm and the golden rule of cryptography states that a mature algorithm is always more trusted.
- * 3DES is, therefore, a more trusted choice in terms of strength, because it has been tested and analyzed for 35 years.

Advanced Encryption Standard

Password:	<input type="text" value="SECRETKEY"/>
Plaintext:	<input type="text" value="FLANK EAST ATTACK AT DAWN"/>
<input type="button" value="Encrypt it"/>	<input type="text"/>
<input type="button" value="Decrypt it"/>	<input type="text"/>

In this example, the SECRETKEY key and plaintext are entered.

Password:	<input type="text" value="SECRETKEY"/>
Plaintext:	<input type="text" value="FLANK EAST ATTACK AT DAWN"/>
<input type="button" value="Encrypt it"/>	<input type="text" value="7zh/SaWlpaWD268p9aj+kkpkZuFG6bt8PEHt9TYV4W1R"/>
<input type="button" value="Decrypt it"/>	<input type="text"/>

They are now encrypted using 128 AES.

Password:	<input type="text" value="secretkey"/>
Plaintext:	<input type="text" value="FLANK EAST ATTACK AT DAWN"/>
<input type="button" value="Encrypt it"/>	<input type="text" value="7zh/SaWlpaWD268p9aj+kkpkZuFG6bt8PEHt9TYV4W1R"/>
<input type="button" value="Decrypt it"/>	<input type="text" value="■G+■A J■pi■TMg■B■>OVμ6\$E"/>

An attempt at deciphering the text using a lowercase, and incorrect key.

Password:	<input type="text" value="SECRETKEY"/>
Plaintext:	<input type="text" value="FLANK EAST ATTACK AT DAWN"/>
<input type="button" value="Encrypt it"/>	<input type="text" value="7zh/SaWlpaWD268p9aj+kkpkZuFG6bt8PEHt9TYV4W1R"/>
<input type="button" value="Decrypt it"/>	<input type="text" value="FLANK EAST ATTACK AT DAWN"/>

A second attempt at deciphering the text using the correct key displays the original plaintext.

Software-Optimized Encryption Algorithm

- * The Software-Optimized Encryption Algorithm (SEAL) is an alternative algorithm to software-based DES, 3DES, and AES.
 - Designed in 1993, it is a stream cipher that uses a 160-bit encryption key.
 - Because it is a stream cipher, data is continuously encrypted and, therefore, much faster than block ciphers.
 - However, it has a longer initialization phase during which a large set of tables is created using SHA (Secure Hash Algorithm).
- * SEAL has a lower impact on the CPU compared to other software-based algorithms.

Software-Optimized Encryption Algorithm

SEAL Scorecard

SEAL Characteristics	
Description	Software-Optimized Encryption Algorithm
Timeline	First published in 1994. Current version is 3.0 (1997)
Type of Algorithm	Symmetric
Key size (in bits)	160
Speed	High
Time to crack (Assuming a computer could try 255 keys per second)	Unknown but considered very safe
Resource Consumption	Low

RC Algorithms

- * The RC algorithms were designed all or in part by Ronald Rivest, who also invented MD5.
- * The RC algorithms are widely deployed in many networking applications because of their favorable speed and variable key-length capabilities.
- * There are several variations of RC algorithms including:
 - RC2
 - RC4
 - RC5
 - RC6

RC Algorithms Cont.

RC Algorithms Scorecard

Ron's Code or Rivest Codes Scorecard		
Description	RC2	RC4
Timeline	1987	1987
Type of Algorithm	Block cipher	Stream cipher
Key size (in bits)	40 and 64	1 - 256

Ron's Code or Rivest Codes Scorecard		
Description	RC5	RC6
Timeline	1994	1998
Type of Algorithm	Block cipher	Block cipher
Key size (in bits)	0 to 2040 bits (128 suggested)	128, 192, or 256

IE2022 - Introduction to Cyber Security

Lecture - 08

Asymmetric Encryption Algorithms and PKI

Mr. Amila Senarathne

Asymmetric Encryption Algorithms and Public Key Infrastructure

- * Reading Assignment
 - CCNA Security Curriculum, Chapter 7: Cryptographic Systems
- * Supplementary text
 - W. Stallings and L. Brown, “Computer Security, Principles and Practice, 2nd edition, Pearson, 2012, Chapter 2 :Cryptographic Tools.

Topics to be discussed

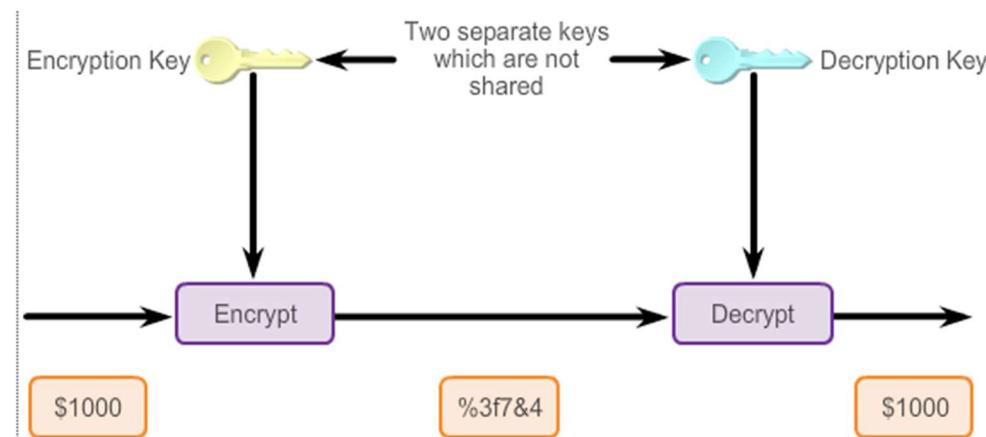
- * Basics of Public Key Cryptography (Asymmetric Encryption)
- * Digital Signatures
 - Properties of Digital Signature
 - Digital Signature Process
 - Digitally Signed Code
- * Diffie-Hellman Key Exchange
- * Asymmetric Encryption Algorithms
- * Public Key Infrastructure

PUBLIC KEY CRYPTOGRAPHY

Asymmetric Encryption Algorithms

Asymmetric encryption algorithms characteristics include:

- Asymmetric encryption algorithms are best known as public key algorithms.
- The usual key length is 512 to 4,096 bits.
- A sender and receiver do not share a secret key.
- These algorithms are relatively slow, because they are based on difficult computational algorithms.
- Examples: RSA, ElGamal, elliptic curves, and DH.



Asymmetric Key Algorithms

- * **Asymmetric algorithms are also called public-key algorithms.**
- * Public-key algorithms are asymmetric algorithms based on the use of two different keys, instead of one.
 - **Private key** - This key must be known *only* by its owner.
 - **Public key** - This key is known to everyone (*it is public*).
- * The key used for encryption is different from the key used for decryption.
 - However, the decryption key cannot, in any reasonable amount of time, be calculated from the encryption key and vice versa.
- * Public-key systems have a clear advantage over symmetric algorithms.
 - There is no need to agree on a common key for both the sender and the receiver.

Asymmetric Key Algorithms Cont.

- * Either key can be used for encryption, but the complementary matched key is required for decryption.
 - If a public key encrypts data, the matching private key decrypts data.
 - If a private key encrypts data, the matching public key decrypts data.

Asymmetric Key Characteristics

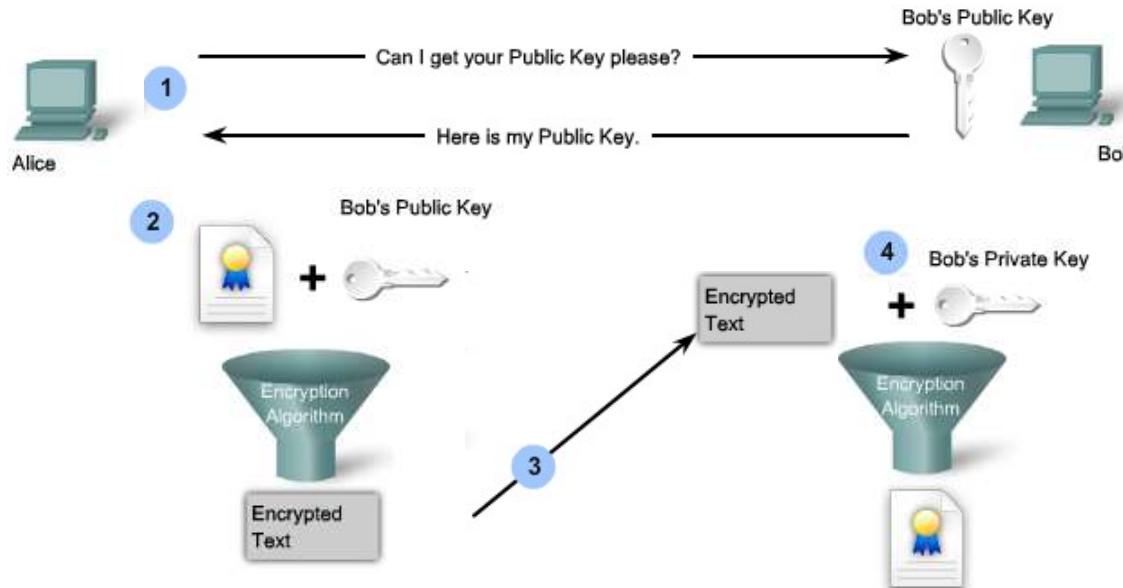


Confidentiality

- * Confidentiality is achieved when the encryption process is started with the public key.
- * When the public key is used to encrypt the data, the private key must be used to decrypt the data.
 - Only one host has the private key guaranteeing confidentiality.

Asymmetric Algorithms for Confidentiality

Public Key (Encrypt) + Private Key (Decrypt) = Confidentiality



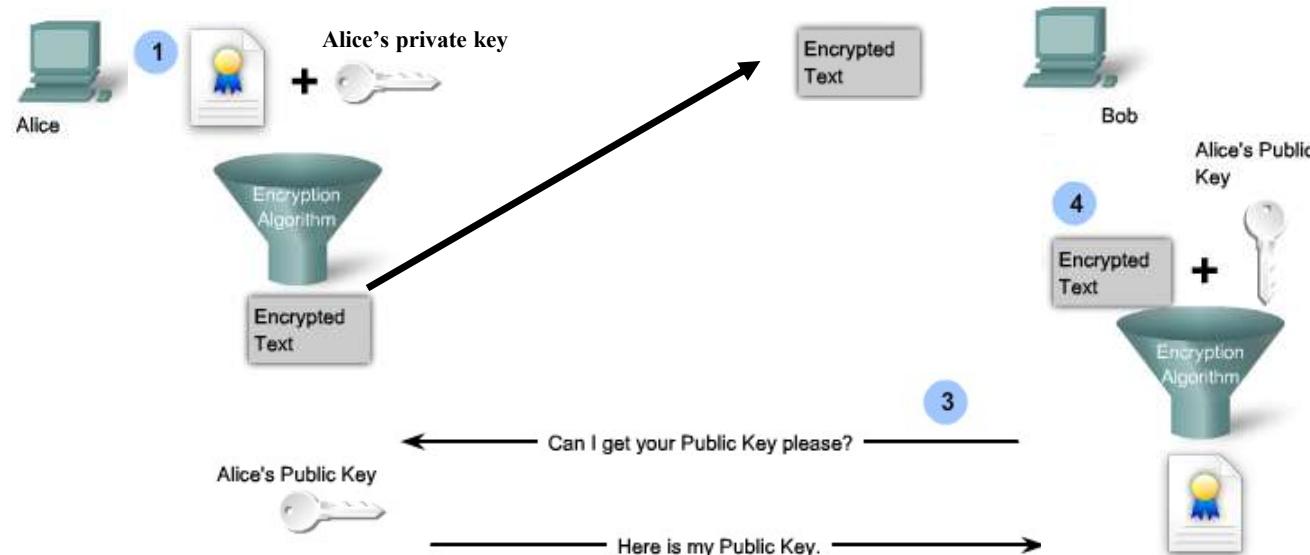
1. Alice asks Bob for his public key and Bob sends it to her.
2. Alice uses Bob's public key to encrypt a message using an agreed-upon algorithm.
3. Alice sends the encrypted message to Bob.
4. Bob uses his private key to decrypt and reveal the message.

Authentication

- * Authentication is achieved when the encryption process is started with the private key.
- * The corresponding public key must be used to decrypt the data.
- * Since only one host has the private key, only that host could have encrypted the message, providing authentication of the sender.

Asymmetric Algorithms for Authentication

Private Key (Encrypt) + Public Key (Decrypt) = Authentication



1. Alice encrypts a message with her private key.
2. Alice transmits the encrypted message to Bob.
3. To verify that the message actually came from Alice, Bob requests and acquires Alice's public key.
4. Bob uses the public key to successfully decrypt the message and authenticate that the message did, indeed, come from Alice.

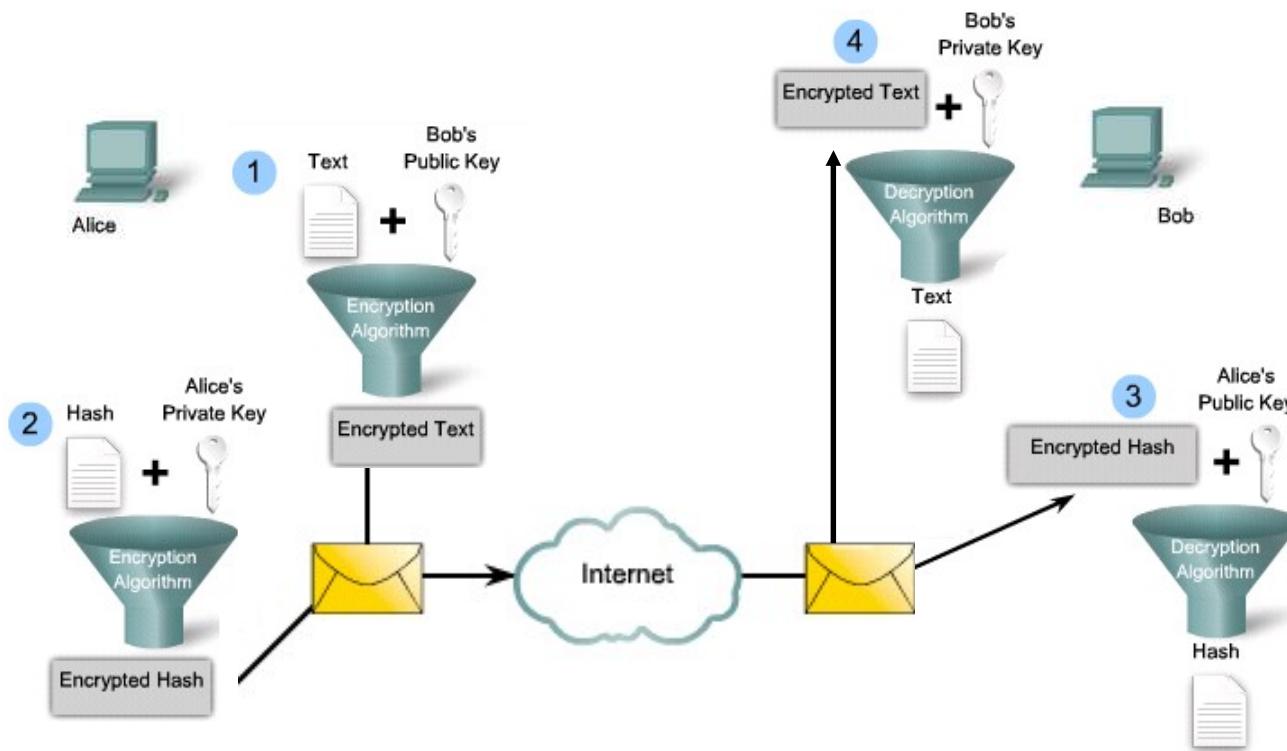
Symmetric Versus Asymmetric Key Algorithms

Asymmetric Algorithms

When sending a message that ensures message confidentiality, authentication and integrity, the combination of two encryption phases is necessary.

- ★ **Phase 1 - Confidentiality**
- ★ **Phase 2 - Authentication and Integrity**

Combining Authentication and Confidentiality



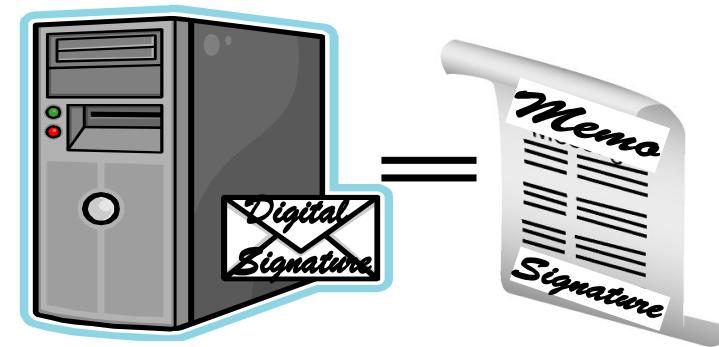
1. Alice encrypts a message using Bob's public key.
2. Alice encrypts a hash of the message using her private key.
3. Bob uses Alice's public key to decrypt and reveal the hash.
4. Bob uses his private key to decrypt and reveal the message.

Using Digital Signatures

- * Authenticity of digitally signed data
 - Digital signatures authenticate a source, proving that a certain party has seen and signed the data in question.
- * Integrity of digitally signed data
 - Digital signatures guarantee that the data has not changed from the time it was signed.
- * Nonrepudiation of the transaction
 - The recipient can take the data to a third party, and the third party accepts the digital signature as a proof that this data exchange did take place.
 - The signing party cannot repudiate that it has signed the data.

Properties

- * **The signature is authentic and not forgeable:** The signature is proof that the signer, and no one else, signed the document.
- * **The signature is not reusable:** The signature is a part of the document and cannot be moved to a different document.
- * **The signature is unalterable:** After a document is signed, it cannot be altered.
- * **The signature cannot be repudiated:** For legal purposes, the signature and the document are considered to be physical things. The signer cannot claim later that they did not sign it.

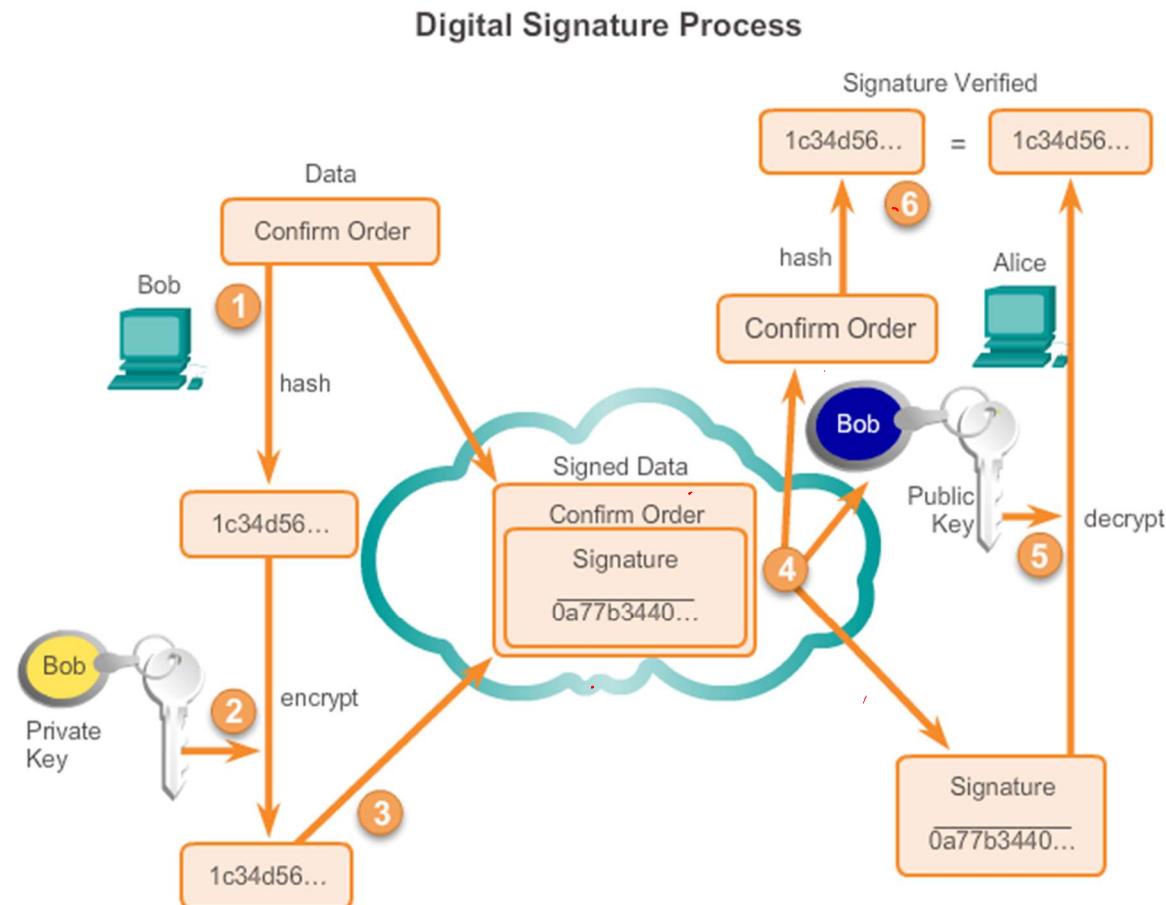


Digital Signature Process

There are six steps to the digital signature process, as shown in the figure (next slide):

1. The sending device, the signer, creates a hash of the document.
2. The sending device encrypts the hash with the private key of the signer.
3. The encrypted hash, known as the signature, is appended to the document.
4. The receiving device, the verifier, accepts the document with the digital signature and obtains the public key of the sending device.
5. The receiving device decrypts the signature using the public key of the sending device. This step unveils the assumed hash value of the sending device.
6. The receiving device makes a hash of the received document, without its signature, and compares this hash to the decrypted signature hash. If the hashes match, the document is authentic; it was signed by the assumed signer and has not changed since it was signed.

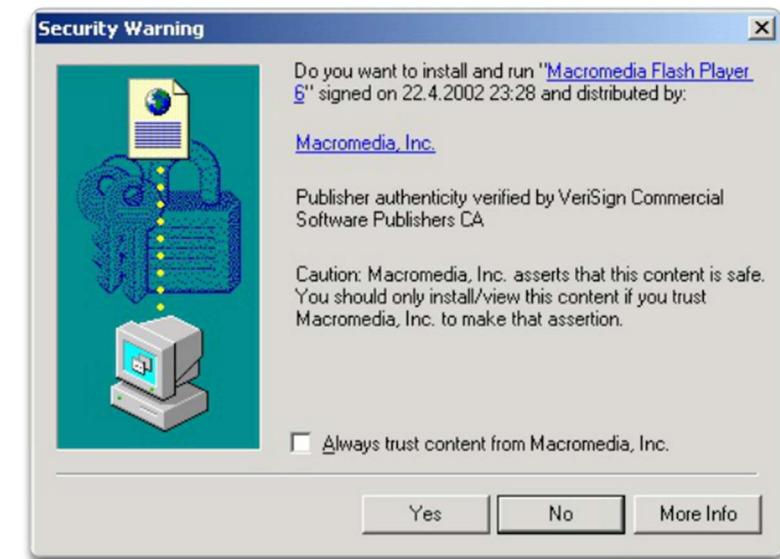
Digital Signature Process Cont.



Digitally Signed Code

Digitally signing code provides several assurances about the code:

- The code has not been modified since it left the software publisher.
- The code is authentic and is actually sourced by the publisher.
- The publisher undeniably publishes the code.
- This provides nonrepudiation of the act of publishing.



Symmetric Versus Asymmetric Key Algorithms

Asymmetric Algorithms Cont.

- * Well-known asymmetric key algorithms:
 - Diffie-Hellman
 - Digital Signature Standard (DSS), which incorporates the Digital Signature Algorithm (DSA)
 - RSA encryption algorithms
 - ElGamal
 - Elliptical curve techniques

Symmetric Versus Asymmetric Key Algorithms

Asymmetric Algorithms

Algorithm	Key length (in bits)	Description
Diffie-Hellman	512, 1024, 2048	<p>Public key algorithm invented in 1976 by Whitfield Diffie and Martin Hellman that allows two parties to agree on a key that they can use to encrypt messages.</p> <p>Security depends on the assumption that it is easy to raise a number to a certain power, but difficult to compute which power was used given the number and the outcome.</p>
Digital Signature Standard and Digital Signature Algorithm	512 - 1024	<p>Created by NIST and specifies DSA as the algorithm for digital signatures.</p> <p>DSA is a public key algorithm based on the ElGamal signature scheme.</p> <p>Signature creation speed is similar with RSA, but is 10 to 40 times as slow for verification.</p>
RSA encryption algorithms	512 to 2048	<p>Developed by Ron Rivest, Adi Shamir, and Leonard Adleman at MIT in 1977.</p> <p>It is an algorithm for public-key cryptography based on the difficulty of factoring very large numbers.</p> <p>It is the first algorithm known to be suitable for signing and encryption, and is one of the first great advances in public key cryptography.</p> <p>Widely used in electronic commerce protocols, and is believed to be secure given sufficiently long keys and the use of up-to-date implementations.</p>
ElGamal	512 - 1024	<p>An asymmetric key encryption algorithm for public-key cryptography which is based on the Diffie-Hellman key agreement.</p> <p>Developed in 1984 and used in GNU Privacy Guard software, PGP, and other cryptosystems.</p> <p>A disadvantage is that the encrypted message becomes very big, about twice the size of the original message, and for this reason, it is only used for small messages, such as secret keys.</p>
Elliptical curve techniques	160	<p>Elliptic curve cryptography was invented by Neil Koblitz in 1987 and by Victor Miller in 1986.</p> <p>Can be used to adapt many cryptographic algorithms, such as Diffie-Hellman or ElGamal.</p> <p>The main advantage of elliptic curve cryptography is that the keys can be much smaller.</p>

Diffie-Hellman Algorithm

- * Whitfield Diffie and Martin Hellman invented the Diffie-Hellman (DH) algorithm in 1976.
- * The DH algorithm is the basis of most modern automatic key exchange methods and is one of the most common protocols used in networking today.
- * DH is not an encryption mechanism
- * DH is not typically used to encrypt data.
 - It is a method to securely exchange the keys that encrypt data.
 - This key can then be used to encrypt subsequent communications using a symmetric key cipher.

Diffie-Hellman Algorithm Cont.

- * DH is commonly used when data is exchanged using an IPsec VPN, data is encrypted on the Internet using either SSL or TLS, or when SSH data is exchanged.
- * It is not an encryption mechanism and is not typically used to encrypt data, because it is extremely slow for any sort of bulk encryption.
- * It is common to encrypt the bulk of the traffic using a symmetric algorithm and use the DH algorithm to create keys that will be used by the encryption algorithm.

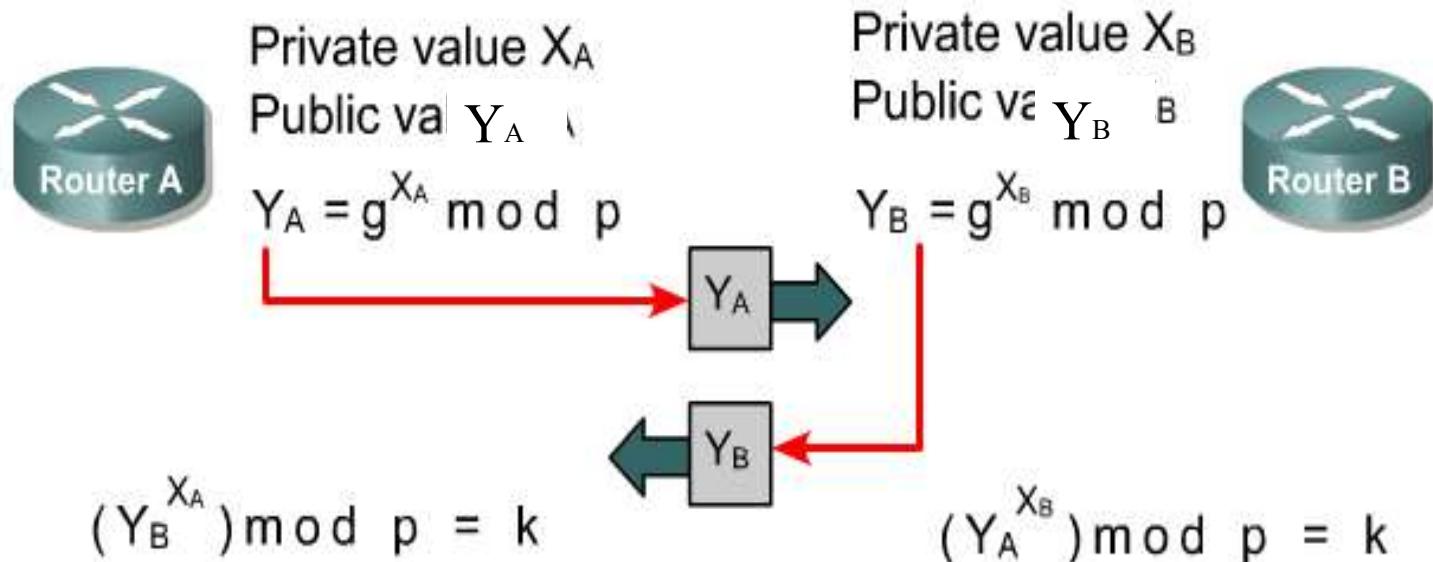
Diffie-Hellman Algorithm Cont.

DH Characteristics

Description	Diffie-Hellman Algorithm
Timeline	1976
Type of Algorithm	Asymmetric
Key size (in bits)	512, 1024, 2048
Speed	Slow
Time to crack (Assuming a computer could try 255 keys per second)	Unknown but considered very safe
Resource Consumption	Medium

Diffie-Hellman Key Exchange DH Operation

Performs authenticated key exchange



DH Operation Cont.

Alice and Bob DH Key Exchange

Alice			Bob		
Shared	Secret	Calc	Shared	Secret	Calc
5, 23			5, 23		
	6	$5^6 \text{ mod } 23 = 8$			



- Bob and Alice agree to use a base number $g=5$ and prime number $p=23$.
- Alice chooses a secret integer $a=6$.
- Alice sends Bob ($g^a \text{ mod } p$) or $5^6 \text{ mod } 23 = 8$.

DH Operation Cont.

Modulo

- In computing, the modulo operation finds the remainder of division of one number by another.
- Given two numbers, **X** and **Y**, a modulo **N** (abbreviated as a mod N) is the remainder, on division of **a** by **N**.
- For instance:
 - "8 mod 3" would evaluate to 2.
 - "9 mod 3" would evaluate to 0.

DH Operation Cont.

Alice and Bob DH Key Exchange

Alice			Bob		
Shared	Secret	Calc	Shared	Secret	Calc
5, 23			5, 23		
	6	$5^6 \text{ mod } 23 = 8$			
		$19^6 \text{ mod } 23 = 2$		15	$5^{15} \text{ mod } 23 = 19$
					$8^{15} \text{ mod } 23 = 2$

- Meanwhile Bob chooses a secret integer $b = 15$.
- Bob sends Alice $(g^a \text{ mod } p)$ or $5^{15} \text{ mod } 23 = 19$.
- Alice computes $(x^a \text{ mod } p)$ or $19^6 \text{ mod } 23 = 2$.
- Bob computes $(x^a \text{ mod } p)$ or $8^6 \text{ mod } 23 = 2$.

DH Operation Cont.

Alice and Bob DH Key Exchange

Alice			Bob		
Shared	Secret	Calc	Shared	Secret	Calc
5, 23			5, 23		
	6	$5^6 \text{ mod } 23 = 8$			
		$19^6 \text{ mod } 23 = 2$		15	$5^{15} \text{ mod } 23 = 19$
					$8^{15} \text{ mod } 23 = 2$

- The result (2) is the same for both Alice and Bob.
- They will now use this as the secret key for encryption.

DH Operation Cont.

Alice and Bob DH Key Exchange

- The initial secret integer used by Alice (6) and Bob (15) are very, very large numbers (1,024 bits).
 - **8 bits = 10101010**
 - **1,024 bits =**

Digital Signature Algorithm

- * Well-known asymmetric algorithms, such as RSA or Digital Signature Algorithm (DSA), are typically used to perform digital signing.
- * In 1994, the U.S. NIST selected the DSA as the DSS. DSA is based on the discrete logarithm problem and can only provide digital signatures.
- * A network administrator must decide whether RSA or DSA is more appropriate for a given situation.
 - DSA signature generation is faster than DSA signature verification.
 - RSA signature verification is much faster than signature generation.

Digital Signature Algorithm Cont.

DSA Scorecard

DSA Characteristics	
Description	Digital Signature Algorithm (DSA)
Timeline	1994
Type of Algorithm	Provides digital signatures
Advantages	Signature generation is fast
Disadvantages	Signature verification is slow

Rivest, Shamir, and Alderman

RSA Asymmetric Algorithm

- * RSA is one of the most common asymmetric algorithms.
- * Ron Rivest, Adi Shamir, and Len Adleman invented the RSA algorithm in 1977.
- * Patented public-key algorithm.
 - The patent expired in September 2000.
 - The algorithm is now in the public domain.

RSA Characteristics	
Description	Ron Rivest, Adi Shamir, and Len Adleman
Timeline	1977
Type of Algorithm	Asymmetric algorithm
Key size (in bits)	512 - 2048
Advantages	Signature verification is fast
Disadvantages	Signature generation is slow

Rivest, Shamir, and Alderman

RSA Summary

- * RSA is about 100 times slower than DES in hardware.
- * RSA about 1,000 times slower than DES in software. This performance problem is the main reason that RSA is typically used only to protect small amounts of data.
- * RSA is mainly used to ensure confidentiality of data by performing encryption, and to perform authentication of data or nonrepudiation of data, or both, by generating digital signatures.

Public Key Infrastructure Overview

- ★ PKI is the service framework needed to support large-scale public key-based technologies. Scalable solutions that are an extremely important authentication solution for VPNs.
- ★ PKI is a set of technical, organizational, and legal components that are needed to establish a system that enables large-scale use of public key cryptography to provide authenticity, confidentiality, integrity, and nonrepudiation services.
- ★ The PKI framework consists of the hardware, software, people, policies, and procedures needed to create, manage, store, distribute, and revoke digital certificates.

Public Key Infrastructure Overview

Driver License PKI Analogy



PKI Framework

- * PKI Certificates-are published public information containing the binding between the names and public keys of entities.
- * PKI Certificate Authority (CA)
 - A trusted third-party entity that issues certificates.
 - A CA always signs the certificate of a user.
 - Every CA also has a certificate containing its public key, signed by itself.
 - This is called a CA certificate or, more properly, a self-signed CA certificate.

Components of a PKI

- * Building a large PKI involves a huge amount of organizational and legal work.
- * There are five main components of a PKI:
 - PKI users, such as people, devices, and servers
 - CAs for key management
 - Storage and protocols
 - Supporting organizational framework, known as practices and user authentication using Local Registration Authorities (LRAs)
 - Supporting legal framework

Components of a PKI Cont.

- * The trust in the certificate is usually determined by how rigorous the procedure was that verified the identity of the holder when the certificate was issued:
 - Class 0 – Used for testing purposes in which no checks have been performed.
 - Class 1 - Used for individuals with a focus on email.
 - Class 2 - Used for organizations for which proof of identity is required.
 - Class 3 - Used for servers and software signing for which independent verification and checking of identity and authority is done by the issuing certificate authority.
 - Class 4 - Used for online business transactions between companies.
 - Class 5 - Used for private organizations or governmental security.

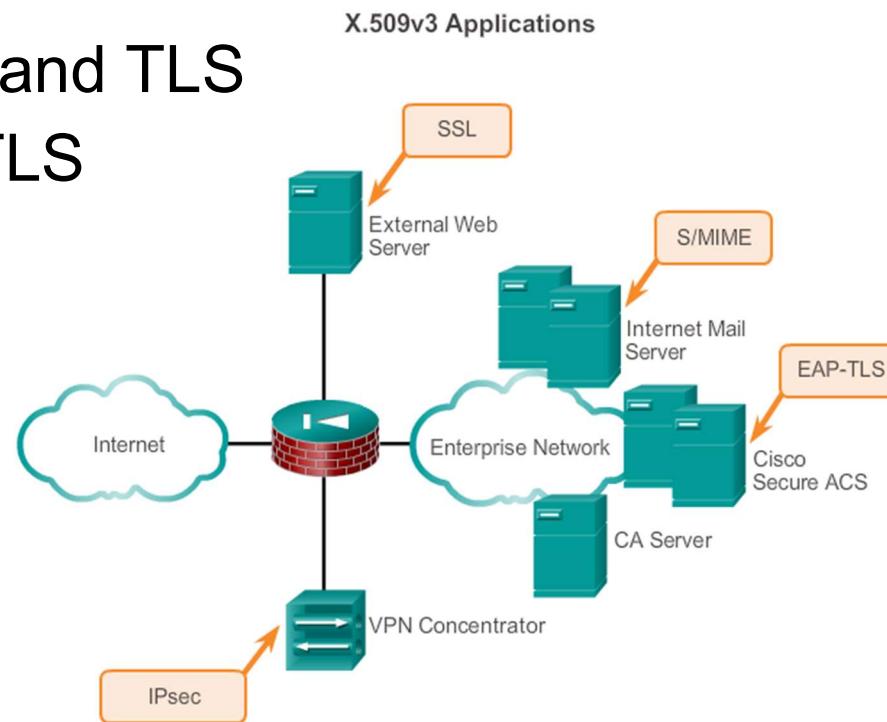
Interoperability of Different PKI Vendors

- ★ Interoperability between different PKI vendors is still an issue.
- ★ To address this interoperability concern, the IETF formed the Public-Key Infrastructure X.509 (PKIX) workgroup, that is dedicated to promoting and standardizing PKI in the Internet.
- ★ This workgroup has published a draft set of standards, X.509, detailing common data formats and PKI-related protocols in a network.



X.509 Standard

- * Defines basic PKI formats, such as the certificate and certificate revocation list (CRL) format to enable basic interoperability.
- * Widely used for years:
 - Secure web servers: SSL and TLS
 - Web browsers: SSL and TLS
 - Email programs: S/MIME
 - IPsec VPN: IKE



PKI Summary

- * PKI as an authentication mechanism has several characteristics:
 - To authenticate each other, users must obtain the certificate of the CA and their own certificate.
 - Public-key systems use asymmetric keys in which one is public and the other one is private.
 - One of the features of these algorithms is that whatever is encrypted using one key can only be decrypted using the other key.
 - This provides nonrepudiation.
 - Key management is simplified, because two users can freely exchange the certificates.
 - The validity of the received certificates is verified using the public key of the CA, which the users have in their possession.
 - Because of the strength of the algorithms involved, administrators can set a very long lifetime for the certificates, typically a lifetime that is measured in years.

Summary

- * Secure communications employs cryptographic methods to protect the confidentiality, integrity, authentication and nonrepudiation of network traffic when traversing the public Internet.
- * Cryptology is the combination of:
 - **Cryptography** - Related to the making and using of encryption methods.
 - **Cryptanalysis** - Related to the solving or breaking of a cryptographic encryption method.
- Cryptographic hashes play a vital role when securing network traffic. For example:
 - Integrity is provided by using the MD5 algorithm or the SHA-1 algorithm.
 - Authenticity is provided using HMAC.
 - Confidentiality is provided using various encryption algorithms.

Summary Cont.

- * Encryption can be implemented using a:
 - **Symmetric algorithm** - Various symmetric encryption algorithms can be used, including DES, 3DES, AES, or SEAL.
 - Each option varies with regard to the degree of protection and the ease of implementation.
 - DH is used to support DES, 3DES, and AES.
 - **Asymmetric algorithm** - These can use digital signatures, such as the RSA algorithm, to provide authentication and confidentiality. Asymmetric encryption is usually implemented using PKI.

IE2022 - Introduction to Cyber Security

Lecture - 09
User Authentication
Mr. Amila Senarathne

Reading Assignment:

- W. Stallings and L. Brown, “Computer Security, Principles and Practice,, Pearson, Chapter 3.
 - Other related materials
- ⋮

Authentication - Definition (RFC 2828)

The process of verifying an identity claimed by or for a system entity. An authentication process consists of two steps:

- **Identification step:** Presenting an identifier to the security system. (Identifiers should be assigned carefully, because authenticated identities are the basis for other security services, such as access control service.)
- **Verification step:** Presenting or generating authentication information that corroborates the binding between the entity and the identifier.

Identification

- An ID provides security because
 - The ID determines if the user is authorized to access the system
 - The ID determines the privileges given to the user
 - e.g., superuser has the highest privilege while guest/anonymous has the least privilege
 - The ID is used as discretionary access control
 - e.g., access rights (read, write, execute) to a file

Vulnerabilities of I&A

Some of I&A's more common vulnerabilities that may be exploited to gain unauthorized system access include:

- Weak authentication methods
- The potential for users to bypass the authentication mechanism
- The lack of confidentiality and integrity for the stored authentication information
- The lack of encryption for authentication and protection of information transmitted over a network
- The user's lack of knowledge on the risks associated with sharing authentication elements (e.g., passwords, security tokens)

Means of Authentication

There are four general means of authenticating a user's identity, which can be used alone or in combination:

- **Something the individual knows:** Examples includes a password, a personal identification number (PIN), or answers to a prearranged set of questions.
- **Something the individual possesses:** Examples include electronic key cards, smart cards, and physical keys. This type of authenticator is referred to as a token.
- **Something the individual is (static biometrics):** Examples include recognition by fingerprint, retina, and face.
- **Something the individual does (dynamic biometrics):** Examples include recognition by voice pattern, handwriting characteristics, and typing rhythm.

Password-Based Authentication

Most widely used means of authentication

- The system maintains a password file indexed by ID
- Typically the system stores one-way hash function of the password
- When a user enters a password, the system compares it with the password for the ID in the file
- Authentication using passwords is vulnerable to attacks



Vulnerabilities of Passwords

Offline dictionary attack

- This attack is possible if the hacker can gain access to the system's password file and compares the password hash against the hashes of common words
- Countermeasures : controls to prevent unauthorized access to the password file, intrusion detection measures to identify a compromise, and rapid reissuance of passwords should the password file be compromised.

Specific account attack:

- The attacker targets a specific account and submits password guesses until the correct password is discovered.
- Countermeasures : account lockout mechanism, which locks out access to the account after a number of failed login attempts. Typical practice is no more than five access attempts.

Vulnerabilities of Passwords

Popular password attack

- Use a popular password and try it against a wide range of user IDs. A user's tendency is to choose a password that is easily remembered
- Countermeasures: Password policies and scanning the IP addresses of authentication requests and client cookies for submission patterns.

Password guessing against single user

- The attacker attempts to gain knowledge about the account holder and system password policies and uses that knowledge to guess the password.
- Countermeasures: training in and enforcement of password policies that make passwords difficult to guess. Such policies address the secrecy, minimum length of the password, character set, prohibition against using well-known user identifiers, and length of time before the password must be changed.

Vulnerabilities of Passwords

Exploiting user mistakes

- User is more likely to write it down because it is difficult to remember. A user may intentionally share a password.
- Also, attackers are frequently successful in obtaining passwords by using social engineering tactics that trick the user or an account manager into revealing a password. Many computer systems are shipped with preconfigured passwords for system administrators.
- Countermeasures: user training, intrusion detection, and simpler passwords combined with another authentication mechanism.

Workstation hijacking

- The attacker waits until a logged-in workstation is unattended.
- Countermeasures : automatically logging the workstation out after a period of inactivity and Intrusion detection schemes can be used to detect changes in user behavior.

Vulnerabilities of Passwords

Exploiting multiple password use

- Attacks can also become much more effective or damaging if different network devices share the same or a similar password for a given user.
- Countermeasures: policy that forbids the same or similar password on particular network devices.

Electronic monitoring

- Passwords communicated across a network to log on to a remote system is vulnerable to eavesdropping.
- Simple encryption will not fix this problem, because the encrypted password is, in effect, the password and can be observed and reused by an adversary

Multi-factor Authentication

Using A combination of more than one method, such as token and password (or personal identification number [PIN] or token and biometric device)

- • • **Two-factor authentication** is a security process in which the user provides **two** means of identification, one of which is typically a physical token, such as a card, and the other of which is typically something memorized, such as a security code
- • •
- • •
- • •

Single Sign-On (SSO)

SSO can generally be defined as the process for consolidating all organization platform-based administration, authentication and authorization functions into a single centralized administrative function. This function would provide the appropriate interfaces to the organization's information resources, which may include:

- Client-server and distributed systems
- Mainframe systems
- Network security including remote access mechanisms

SSO Advantages

- Multiple passwords are no longer required; therefore, a user may be more inclined and motivated to select a stronger password.
- It improves an administrator's ability to manage users' accounts and authorizations to all associated systems.
- It reduces administrative overhead in resetting forgotten passwords over multiple platforms and applications.
- It reduces the time taken by users to log into multiple applications and platforms.

SSO Disadvantages

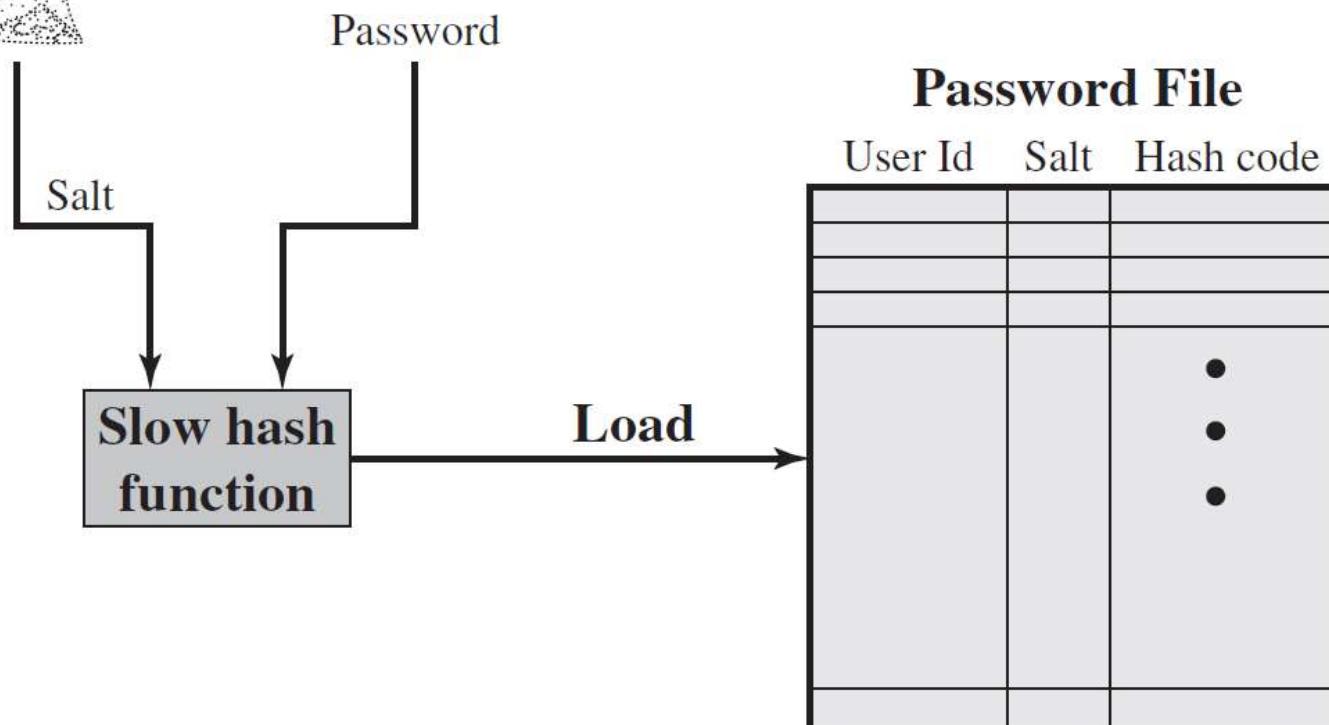
- Support for all major operating system environments is difficult. SSO implementations will often require a number of solutions integrated into a total solution for an enterprise's IT architecture.
- The costs associated with SSO development can be significant when considering the nature and extent of interface development and maintenance that may be necessary.
- The centralized nature of SSO presents the possibility of a single point of failure and total compromise of an organization's information assets. For this reason, "strong authentication" in the form of complex password requirements and the use of biometrics is frequently implemented.

Hashed passwords with salt

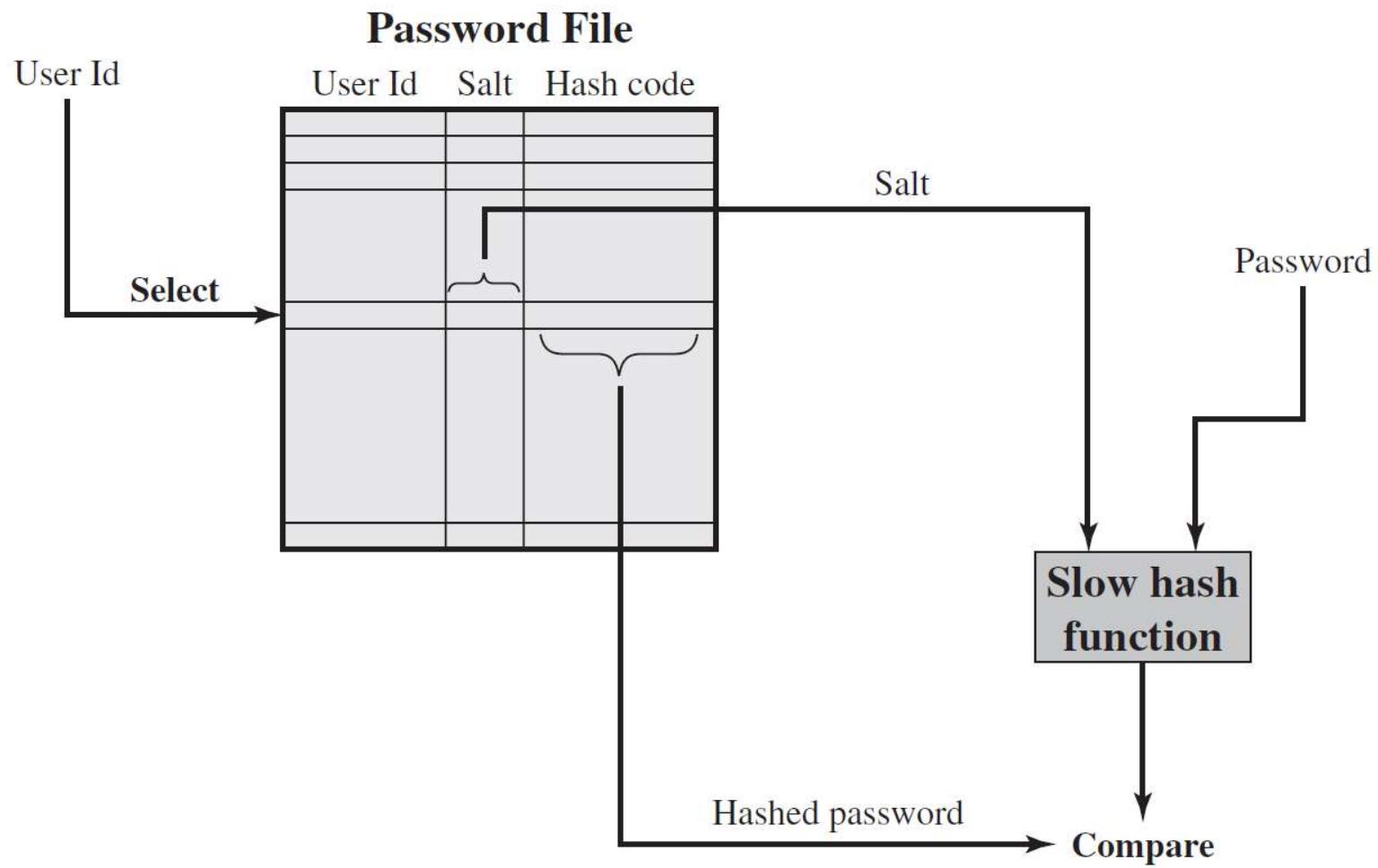
- Most systems, e.g., Linux, store hashed passwords and a salt value for better security
- Steps to store a password:
 - Given a password (selected by user or assigned by system), the system generates a fixed length pseudorandom/random number, called salt
 - Older system uses time when the password is created to generate the salt
 - Use hash function to generate a fixed length hashed code of the password and its salt
 - Store the hashed code and a plaintext copy of the salt in the password file

Hashed passwords with salt

- Steps to verify a password:
 - Given a user ID and a password, the system uses the ID to retrieve the plaintext salt and the encrypted password
 - Use the salt and the supplied password as input to the encryption function
 - If the result matches the stored encrypted value, the password is accepted
 - . . .



(a) Loading a new password



(b) Verifying a password

Purposes of Using Salt

- **To prevents duplicate passwords in the password file**

Each password is assigned a different salt value. Thus even if two users use the same password, the stored hashed passwords would be different

- **To significantly increases the difficulty of offline dictionary attacks**
 - A b bit salt will increase the number of possible passwords by a factor of 2^b , and thus guessing password would be harder
- **To makes almost impossible to find out if a person use the same password on two or more systems**

Remote user authentication

- Remote user authentication raises additional security threats such as eavesdropping and replay attack
 - The counter measure generally relies on challenge-response protocol, such as Kerberos

Challenge-response protocol

- Steps of a simple challenge-response protocol
- User transmits his/her ID to the remote host
- The host generates a random number r , called a nonce, and returns it to the user.
- The host also specifies two functions, a hash function $h()$ and $f()$ to be used for the user's response
 - The host keeps function $h()$ for the password of each of its users $U \rightarrow h(P(U))$
 - This is the challenge
- The user must send a correct response $f(r', h(P'))$ to the host
 - $r'=r$ and P' is the user's password
- The host calculates $f(r, h(P(U)))$ and compares it with the received $f(r', h(P'))$

Challenge-Response Protocol

Client	Transmission	Host
U , user	$U \rightarrow$	
	$\leftarrow \{r, h(), f()\}$	random number $h()$, $f()$, functions
P' password r' , return of r	$f(r', h(P')) \rightarrow$	
	\leftarrow yes/no	if $f(r', h(P')) =$ $f(r, h(P(U)))$ then yes else no

(a) Protocol for a password

- From W. Stallings and L. Brown, "Computer Security, Principles and Practice, 2nd edition

Table 3.4 Some Potential Attacks, Susceptible Authenticators, and Typical Defenses

Attacks	Authenticators	Examples	Typical Defenses
Client attack	Password	Guessing, exhaustive search	Large entropy; limited attempts
	Token	Exhaustive search	Large entropy; limited attempts, theft of object requires presence
	Biometric	False match	Large entropy; limited attempts
Host attack	Password	Plaintext theft, dictionary/exhaustive search	Hashing; large entropy; protection of password database
	Token	Passcode theft	Same as password; 1-time passcode
	Biometric	Template theft	Capture device authentication; challenge response
Eavesdropping, theft, and copying	Password	"Shoulder surfing"	User diligence to keep secret; administrator diligence to quickly revoke compromised passwords; multifactor authentication
	Token	Theft, counterfeiting hardware	Multifactor authentication; tamper resistant/evident token
	Biometric	Copying (spoofing) biometric	Copy detection at capture device and capture device authentication
Replay	Password	Replay stolen password response	Challenge-response protocol
	Token	Replay stolen passcode response	Challenge-response protocol; 1-time passcode
	Biometric	Replay stolen biometric template response	Copy detection at capture device and capture device authentication via challenge-response protocol
Trojan horse	Password, token, biometric	Installation of rogue client or capture device	Authentication of client or capture device within trusted security perimeter
Denial of service	Password, token, biometric	Lockout by multiple failed authentications	Multifactor with token

- From W. Stallings and L. Brown, "Computer Security, Principles and Practice, 2nd edition

Password Selection Strategies

- User education
- Computer-generated passwords
- Reactive password checking
- Proactive password checking

IE2022 - Introduction to Cyber Security

Lecture - 10

Access Control

Mr. Amila Senarathne

Reading Assignment:

- W. Stallings and L. Brown, “Computer Security, Principles and Practice,, Pearson, Chapter 4.
 - Other related materials
- ⋮

Access Control

ITU-T Recommendation X.800's definition

Access control is the prevention of unauthorized use of a resource, including the prevention of use of a resource in an unauthorized manner.

- Access control is a critical element in computer security because the main objective of computer security is
 - To prevent unauthorized users from accessing resources
 - To prevent legitimate users from accessing unauthorized resources
 - To enable users to access resources in an authorized way

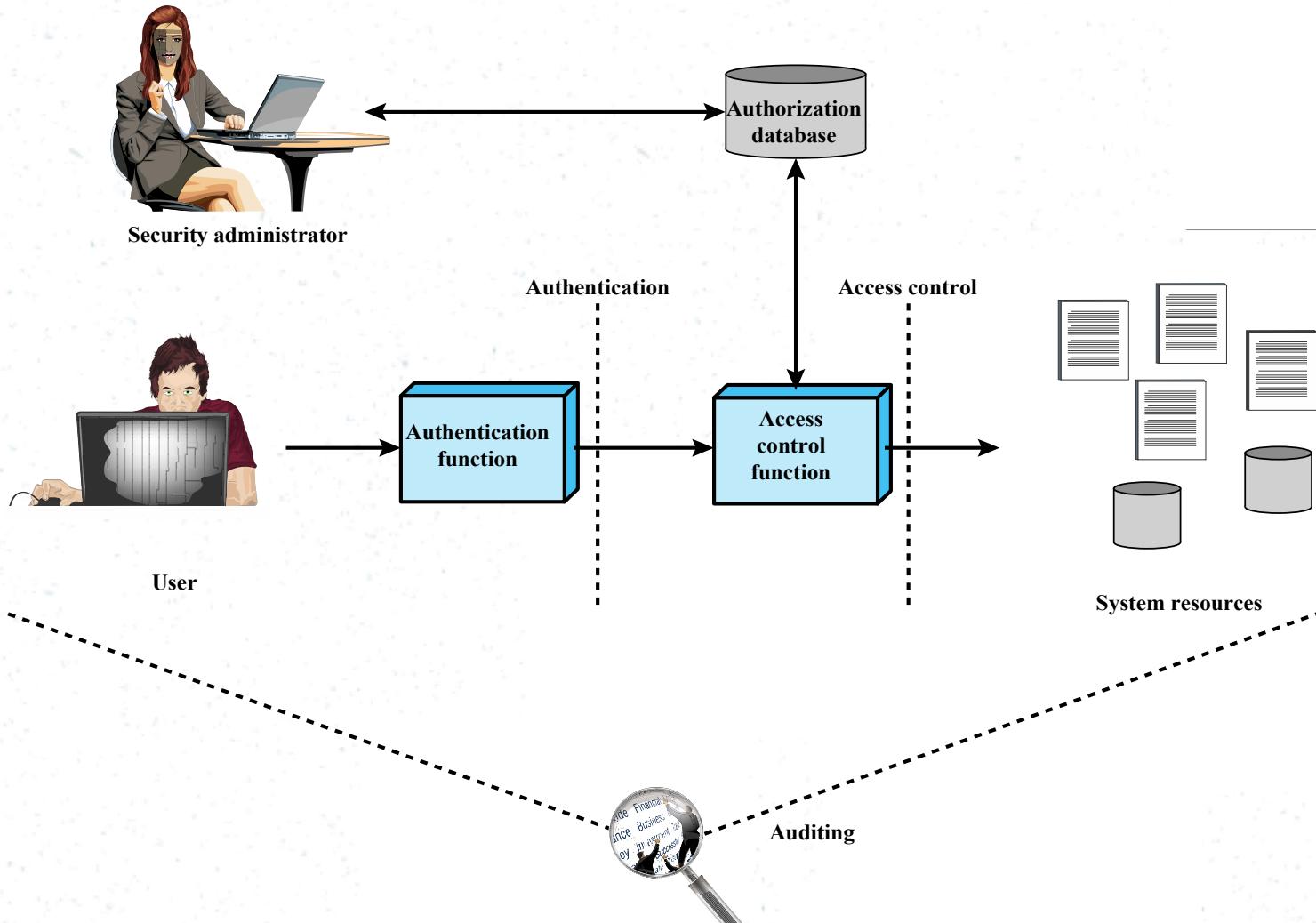


Figure 4.1 Relationship Among Access Control and Other Security Functions

Access Control Policies

An access control policy, which can be embodied in an authorization database, dictates what types of access are permitted, under what circumstances, and by whom. Access control policies are generally grouped into the following categories:

- Discretionary Access Control (DAC)
- Mandatory Access Control (MAC)
- Role-Based Access Control(RBAC)
- • •
- • •

Discretionary Access Control (DAC)

- Controls access based on the identity of the requestor and on access rules (authorizations) stating what requestors are (or are not) allowed to do. This policy is termed discretionary because an entity might have access rights that permit the entity, by its own volition, to enable another entity to access some resource.

Access Control Policies

Mandatory access control (MAC):

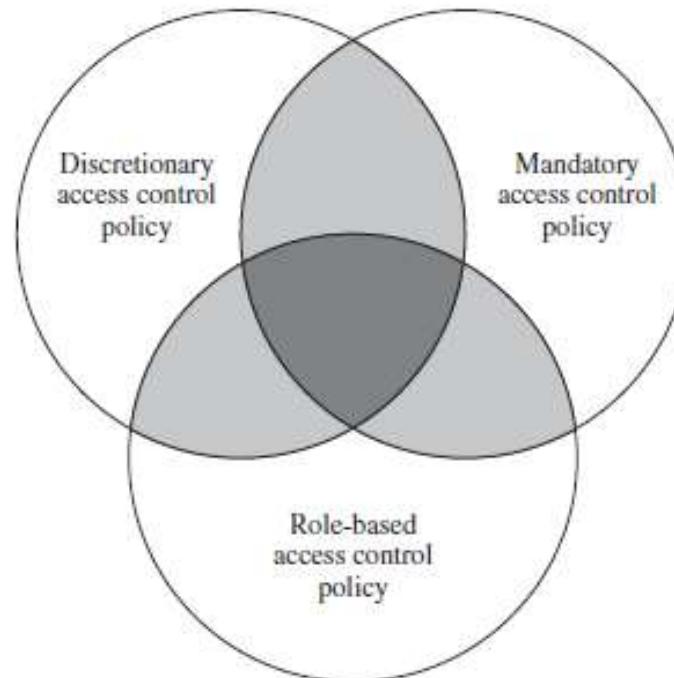
Controls access based on comparing Security labels (which indicate how sensitive or critical system resources are) with security clearances (which indicate system entities are eligible to access certain resources).

This policy is termed mandatory because an entity that has clearance to access a resource may not, just by its own volition, enable another entity to access that resource.

Role-based access control (RBAC):

Controls access based on the roles that users have within the system and on rules stating what accesses are allowed to users in given roles.

Access Control Policies



- DAC, MAC, and RBAC are not mutually exclusive. A system may implement two or even three of these policies for some or all types of access.

Elements of Access Control

- **Subject: an entity that accesses object**

A subject is an application or a user that is represented by a process in the system that takes on the user's attribute, e.g., access right

Three classes of subject: owner, group, world

- **Object: the resource which access is to be controlled**

Example: records, files, mailbox, program, messages

- **Access Right: the way a subject may access an object**

Access right includes: read, write, execute, delete, create, search

Discretionary Access Control

General access control in OS uses an access matrix

- One dimension (column) consists of subjects that need to access objects
- The other dimension (row) lists the objects that can be accessed
- Each entry in the matrix contains access rights of the subject in that row for the object in that column

Access Matrix

		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

ACLs and Capability Tickets

Access matrix is usually sparse, and implemented by decomposing it into one or two ways:

- By columns resulting in Access Control Lists (ACLs) for all objects
- By rows resulting in capability list/tickets for all subjects/users

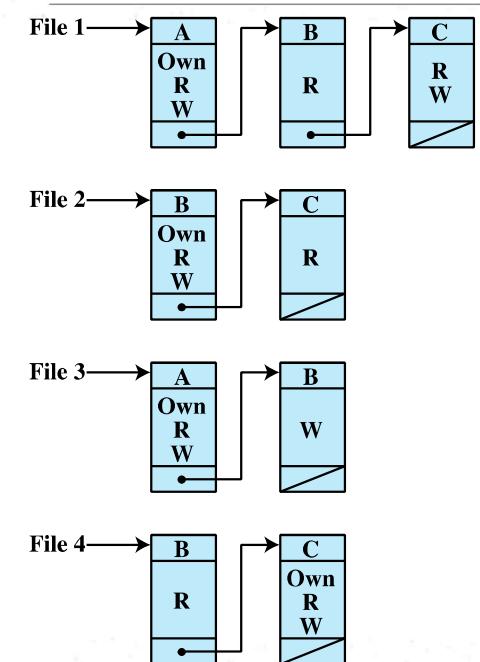
• • • **Each list for an object in ACL lists users and their access rights to access the object**

- • • • **ACL may contain a default or public entry to allow users that are not explicitly listed to have a default access right**
- • • **Access rights should follow the least privilege or read-only access**
- **Elements in the list can be an individual or group users**

Access Control Lists (ACLs)

ACL is efficient when we want to know which subjects have what access rights to a particular object

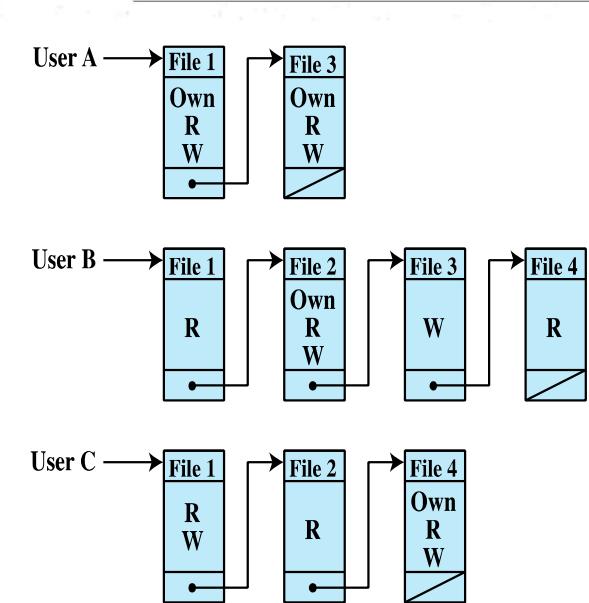
- However, it is harder to determine what access rights a specific user has on which objects



Capability Tickets

Each capability ticket what access rights a particular user has on the objects in the list

- Each user has a number of tickets and they can loan or give the tickets to other users



Capability Tickets

Tickets may be spread around the system

- The tickets cause a greater security problem than ACL

The OS must protect and guarantee the integrity of each ticket; the ticket must be unforgeable

- OS keeps all tickets for the user in a memory region inaccessible by users

- Users must use a system call to request for their tickets

For distributed system, the ticket is in a form of a token

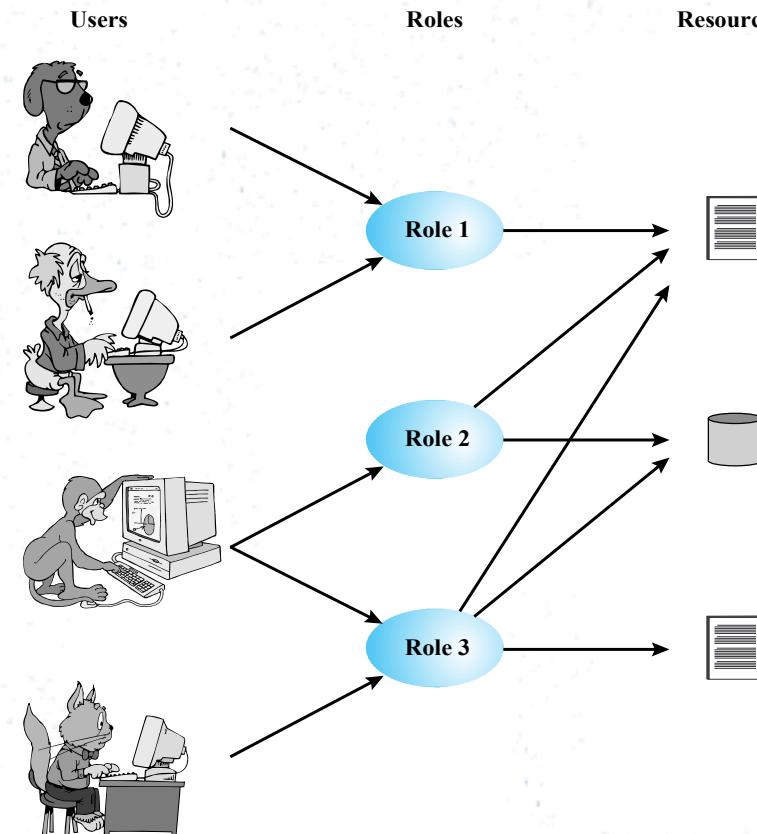
- A token can be a large random password, or a cryptographic message authentication code whose value is verified by the corresponding resource when requesting for access

Role-based Access Control (RBAC)

RBAC defines the access rights based on the roles the users assume in the system rather than the user's identity like in DAC

- Role is a job function in the organization
- RBAC assign rights to the roles, not the users
- Users are assigned roles either statically or dynamically
- The relationship between users and roles are many-to-many

Role-based Access Control (RBAC)



Role-based Access Control (RBAC)

- Access matrix representation can be used to describe the key elements of RBAC;

	R ₁	R ₂	• • •	R _n
U ₁	X			
U ₂	X			
U ₃		X		X
U ₄				X
U ₅				X
U ₆				X
•				
•				
U _m	X			

	OBJECTS								
	R ₁	R ₂	R _n	F ₁	F ₁	P ₁	P ₂	D ₁	D ₂
ROLES	control	owner	owner control	read *	read owner	wakeup	wakeup	seek	owner
R ₁									
R ₂		control		write *	execute			owner	seek *
•									
•									
R _n			control		write	stop			

Access Control Requirements

Concepts and features that should be supported by an access control system

- Reliable Input
- Support for fine and coarse specifications
- Least privilege
- Separation of duty
- Open and Closed policy
- Policy combination and conflict resolution
- Administrative policy
- Dual control

Access Control Requirements

Reliable Input

An access control system assumes that a user is authentic; thus, an authentication mechanism is needed as a front end to an access control system. Other inputs to the access control system must also be reliable. For example, some access control restrictions may depend on an address, such as a source IP address or medium access control address. The overall system must have a means of determining the validity of the source for such restrictions to operate effectively.

Access Control Requirements

Support for fine and coarse specifications : The access control system should support fine-grained specifications, allowing access to be regulated at the level of individual records in files, and individual fields within records. The system should also support fine-grained specification in the sense of controlling each individual access by a user rather than a sequence of access requests. System administrators should also be able to choose coarse-grained specification for some classes of resource access, to reduce administrative and system processing burden.

Least privilege : This is the principle that access control should be implemented so that each system entity is granted the minimum system resources and authorizations that the entity needs to do its work. This principle tends to limit damage that can be caused by an accident, error, or fraudulent or unauthorized act.

Access Control Requirements

Separation of duty : This is the practice of dividing the steps in a system function among different individuals, so as to keep a single individual from subverting the process. This is primarily a policy issue; separation of duty requires the appropriate power and flexibility in the access control system, including least privilege and fine-grained access control. Another useful tool is history-based authorization, which makes access dependent on previously executed accesses.

Open and closed policies: The most useful, and most typical, class of access control policies are closed policies. In a closed policy, only accesses that are specifically authorized are allowed. In some applications, it may also be desirable to allow an open policy for some classes of resources. In an open policy, authorizations specify which accesses are prohibited; all other accesses are allowed.

Access Control Requirements

Policy combinations and conflict resolution: An access control mechanism may apply multiple policies to a given class of resources. In this case, care must be taken that there are no conflicts such that one policy enables a particular access while another policy denies it. Or, if such a conflict exists, a procedure must be defined for conflict resolution.

- **Administrative policies:** As was mentioned, there is a security administration function for specifying the authorization database that acts as an input to the access control function. Administrative policies are needed to specify who can add, delete, or modify authorization rules. In turn, access control and other control mechanisms are needed to enforce the administrative policies.
- **Dual control:** When a task requires two or more individuals working in tandem.

IE2022 – Introduction to Cyber Security

Lecture - 11

Malicious Software (Malware)

Mr. Amila Senarathne

Reading Assignment

- * W. Stallings and L. Brown, “Computer Security, Principles and Practice, 2nd edition, Pearson, 2012, Chapter 6.

Topics to be discussed

- * Definition of malwares
- * Malware propagation
- * Malware payloads
- * Malware countermeasures
- * Malware detection mechanisms

Malicious Software (Malware)

Malicious Software (Malware) is defined by NIST as:

A program that is inserted into a system, usually covertly, with the intent of compromising the confidentiality, integrity, or availability of the victim's data, applications, or operating system or otherwise annoying or disrupting the victim.

–Malware is one of the most significant threats to computer systems

- * Application programs
- * Utility program (editor, compiler)
- * Kernel program
- * Websites and server
- * Spam emails to trick users, etc

Name	Description
Adware	Advertising that is integrated into software. It can result in pop-up ads or redirection of a browser to a commercial site.
Attack Kit	Set of tools for generating new malware automatically using a variety of supplied propagation and payload mechanisms
Auto-rooter	Malicious hacker tools used to break into new machines remotely.
Backdoor (trapdoor)	Any mechanisms that bypasses a normal security check; it may allow unauthorized access to functionality in a program, or onto a compromised system.
Downloaders	Code that installs other items on a machine that is under attack. It is normally included in the malware code first inserted on to a compromised system to then import a larger malware package.
Drive-by-Download	An attack using code in a compromised web site that exploits a browser vulnerability to attack a client system when the site is viewed.
Exploits	Code specific to a single vulnerability or set of vulnerabilities.
Flooders (DoS client)	Used to generate a large volume of data to attack networked computer systems, by carrying out some form of denial-of-service (DoS) attack.
Keyloggers	Captures keystrokes on a compromised system.
Logic bomb	Code inserted into malware by an intruder. A logic bomb lies dormant until a predefined condition is met; the code then triggers an unauthorized act.

Table 6.1 lists the common malware terminology used throughout Chapter 6 : From Stallings & Brown textbook

Name	Description
Macro Virus	A type of virus that uses macro or scripting code, typically embedded in a document, and triggered when the document is viewed or edited, to run and replicate itself into other such documents.
Mobile code	Software (e.g., script, macro, or other portable instruction) that can be shipped unchanged to a heterogeneous collection of platforms and execute with identical semantics.
Rootkit	Set of hacker tools used after attacker has broken into a computer system and gained root-level access.
Spammer programs	Used to send large volumes of unwanted e-mail.
Spyware	Software that collects information from a computer and transmits it to another system by monitoring keystrokes, screen data and/or network traffic; or by scanning files on the system for sensitive information.
Trojan horse	A computer program that appears to have a useful function, but also has a hidden and potentially malicious function that evades security mechanisms, sometimes by exploiting legitimate authorizations of a system entity that invokes the Trojan horse program.
Virus	Malware that, when executed, tries to replicate itself into other executable machine or script code; when it succeeds the code is said to be infected. When the infected code is executed, the virus also executes.
Worm	A computer program that can run independently and can propagate a complete working version of itself onto other hosts on a network, usually by exploiting software vulnerabilities in the target system.
Zombie, bot	Program activated on an infected machine that is activated to launch attacks on other machines.

Table 6.1 lists the common malware terminology used throughout Chapter 6 : From Stallings & Brown textbook

Malware Classification

- * There is no universally accepted classification
 - One classification is based on:
 - * How malware first spreads/propagates to reach its target
 - * The payloads/actions malware performs on the target
 - Other malware classification:
 - * Parasitic software that needs a host program (e.g., virus)
 - * Self contained software (e.g., worms, trojans)
 - * Malware that do not replicate (trojans, email spam)
 - * Malware that replicates (virus, worms)
- * Malware can be created using a crimeware – a toolkit that can be used to create malware with various propagation and payload
 - e.g., Zeus Crimeware Toolkit
- * The source of malware can be individuals and organizations

Malware Propagation/Payloads

Propagations:

- * By infection - Infecting existing program that spread to other system (e.g., virus)
- * By exploiting vulnerability - Attacking software vulnerabilities that allow malware to be downloaded/spread, e.g., worm
- * By social engineering - Tricking users to install the malware (trojan, phishing)

Payloads:

- * Corrupting host systems and data
- * Stealing system resources/service to make it zombie/botnet
- * Stealing system information (login, password, other personal details)
- * Stealthing – hiding within the host system to avoid detection

Propagation by Infection

- * Virus: a fragment of program that attaches to some executable code
 - First appeared in early 1980's
 - The name is by Fred Cohen
- * Virus modifies existing program with a routine to replicate the virus code to go infecting other content
- * The program fragment can be:
 - Machine code that infects existing programs
 - Scripting code that is used to support data files in MS Words, Excels, and Adobe PDF
- * The operations that the virus can do depends on the rights of the program it is attached to:
 - It operates secretly when the host program runs
 - It can erase files/programs

Virus components and phases

- ★ Virus has three components (Aycock, J, 2006):
 - Infection mechanism:– the tool for the virus to propagate
 - * also called infection vector
 - Trigger: when the payload is activated
 - * also called logic bomb
 - Payload: what the virus does
- ★ Virus has four phases in its lifetime:
 - Dorman phase: the virus is idle
 - * It will eventually be activated by some events
 - Propagation phase: the virus put a copy of itself into other program or disk
 - * the copy may or may not be identical to avoid detection
 - Triggering phase: the virus is activated to perform its intended function
 - Execution phase: the function is performed
 - * It can be harmless but annoying or damaging

Executable Virus

- * A machine executable code virus can be
 - Pre-pended or post-pended to some executable program
 - Embedded into some executable program
- * The infected program will first execute the virus code before executing the original program
- * The general virus structure is shown as follows (Fig. 6.3, textbook; also from Cohen 94)
 - It can be easily detected since the infected program is longer (in bytes) than the original
 - A simple way to avoid easy detection is by compressing the code so that both infected and original program are the same size (See Figure 6.2, textbook)

Simple virus – example-1 (from Stallings & Brown)

```
program V :=  
  
{goto main;  
 1234567;  
  
 subroutine infect-executable :=  
   {loop:  
     file := get-random-executable-file;  
     if (first-line-of-file = 1234567)  
       then goto loop  
     else prepend V to file; }  
  
 subroutine do-damage :=  
   {whatever damage is to be done}  
  
 subroutine trigger-pulled :=  
   {return true if some condition holds}  
  
main:  main-program :=  
       {infect-executable;  
        if trigger-pulled then do-damage;  
        goto next; }  
  
next:  
  
}
```

Figure 6.1 A Simple Virus

Virus logic with compression

Consider a program P1 infected with virus CV and an uninfected program P2. Assume P1+CV becomes P1'.

When P1 is executed, its virus will do the following:

- 1) Compress P2 to create P2' such that the size of P2'+CV=P2
- 2) Prepend a copy of CV to P2'
- 3) Uncompress P1'
- 4) P1 is executed.

- * This example shows how the virus propagate undetected.
- * Virus infection can be avoided if the virus can be blocked from entering the system.

Simple virus – example-1 (from Stallings & Brown)

```
program CV :=  
  
{goto main;  
 01234567;  
  
 subroutine infect-executable :=  
   {loop:  
     file := get-random-executable-file;  
     if (first-line-of-file = 01234567) then goto loop;  
     (1)      compress file;  
     (2)      prepend CV to file;  
   }  
  
 main: main-program :=  
   {if ask-permission then infect-executable;  
   (3)      uncompress rest-of-file;  
   (4)      run uncompressed file;}  
 }
```

Figure 6.2 Logic for a Compression Virus

Compression –operations (from Stallings & Brown)

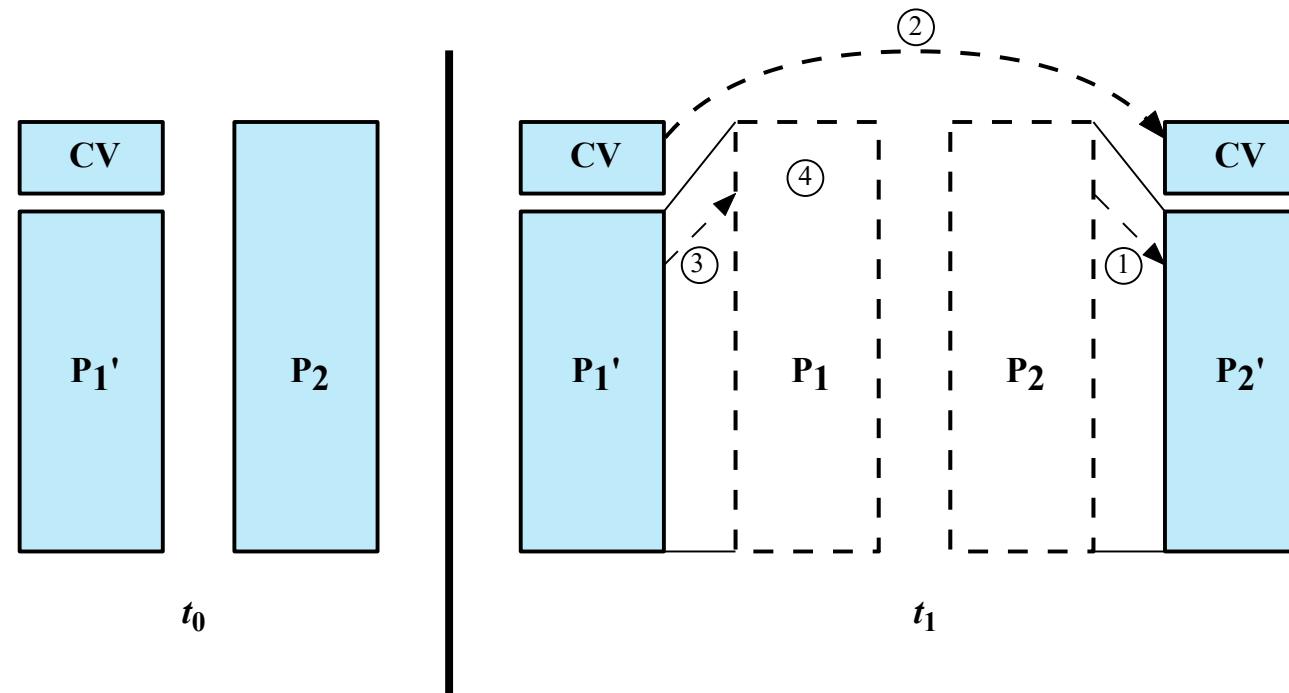


Figure 6.3 A Compression Virus

Virus Classification

- * Ayccock, classifies viruses into two classes:
 - By the targets that the viruses attack
 - By the method the viruses hide from detection

- * Virus Categories by its targets:
 - Boot Sector Infector – infecting the master boot record and spreading when the system is booted
 - File Infector – infecting executable files
 - Macro virus – infecting macro or scripting files interpreted by the application
 - Multipartite virus – infecting files in multiple ways

Virus Classification

- * Virus categories by its concealment:

- Encrypted virus
 - * A portion of the virus creates a random encryption key and encrypts the remainder of the virus
 - * Store the key with the virus
 - * When an infected program is executed, the virus decrypts itself
 - * When the virus spreads, it creates a different random key.
 - * No constant bit pattern is noticed since each virus has different key
- Stealth virus – a form of virus and its payload that are intentionally designed to hide from detection
- Polymorphic virus – a virus that is hard to detect by its signature since it mutates/changes with every infection
- Metamorphic virus – like polymorphic, it mutates in every infection
 - * However, metamorphic completely rewrites itself at every iteration that make it even harder to detect

Macro and Scripting Viruses

Threats of macro or scripting code viruses:

- * A macro virus is platform independent.
 - Any system that support the applications using the macro can be infected
 - MS office, PDF
- * It infects documents, not code
 - More documents are input to the system than code
- * It can easily spread
 - Documents are commonly shared
- * Traditional file system access controls are not effective in preventing their spread
 - Users are expected to modify documents

Propagation by exploiting vulnerability

- * A worm is a program that looks for other machines to infect
 - Each infected machine is also used to find other machines
 - John Brunner (1975) uses the term “worm” and its concept in his novel book “The Shockwave Rider”
 - The first known worm was not malicious, implemented in Xerox labs
 - The Morris worm: a well-known worm spread on Unix in 1988
- * Several worms are also viruses, e.g., Melissa, Nimda
- * A worm exploits software vulnerabilities in either client or server.
- * Worms can spread through network connections, shared media (USB, CD, DVD), and email
 - A worm can spread faster than a virus, e.g., Code Red worm infected 360k computers in 14 hours

Means for worm replications

- * Electronic mail or instant messenger facility
 - Melissa, Nimda, Mydoom, Warezov
- * File sharing
 - Conficker, Stuxnet
- * Remote execution capability
- * Remote file access or transfer capability
- * Remote login capability

Worm phases

- ★ Similar to virus, worms has four phases: dormant, propagation, triggering, execution.
- ★ Propagation phase performs:
 - Search for means to access other systems to infect
 - * Host tables, address books, buddy lists, trusted peers, target host addresses, and others.
 - Use the access to transfer a copy of itself and execute the copy
 - Worm can check if the system has been infected

How does worm find a target?

- * The first step, scanning or fingerprinting, is a function for network worm to search for other system to infect
 - Identify systems running vulnerable service.
- * Types of network address scanning strategies
 - Random – use random IP addresses
 - Hit-List – compile a long list of vulnerable machines and infect the machines on the list
 - Topological – use information contained on infected machine to find more hosts to scan
 - Local subnet – look for targets within the same local network behind firewalls

Worm propagation model

- * The speed of propagation depends on:
 - the mode of propagation: by email? By file sharing? Etc.
 - the exploited vulnerability,
 - the similarity to previous attack
- * Three phases of propagation:
 - Initial phase: the number of host increases exponentially
 - Middle phase: linear growth
 - Finish phase: slow since remaining hosts are mostly infected

Trojan Horses

- * A trojan horse contains a hidden code that when called performed unwanted or harmful function
 - It may be a useful program or utility
- * Some possible harmful function:
 - Gain access to sensitive personal information and send a copy of it to the attacker.
- * Users must be careful when downloading software from unknown source

Payload – System Corruption

- * Payload is the action that the malware takes on the target
 - Some malware does not have payload
- * Some possible payload: data destruction, real-world damage, logic bomb
- * Examples of malware that destruct data:
 - Chernobyl virus: it deletes data on the infected system by overwriting the first megabyte of the hard disk with zeroes
 - Klez mass-mailing worm: on trigger date, it causes files on hardware to become empty
 - Cyborg trojan and Gpcode encrypt the user's data and ask for ransom to decrypt the data

Payload – System Corruption

- * Examples of malware that cause real world damage:
 - The payload aims to damage the physical system
 - Chernobyl virus also attempts to overwrite the BIOS code that boot the computer
 - * If successful, the BIOS chip must be replaced or reprogrammed
 - Stuxnet worm: targets specific industrial control system software
 - * The worm replaces the original code and drive the controller equipment beyond its normal operation to cause failure and damage
- * Logic bomb is a code as part of a malware that will explode when certain conditions are met such as,
 - Presence or absence of certain files or devices
 - A particular date or day
 - A particular user running the program

Payload – Attack Agent

- * A bot is a compromised machine that can be remotely controlled by the attacker (the bot master)
 - Also called robot, drone, or zombie
 - Botnet is a collection (hundreds, thousands, even millions) of bots under the control of the bot master.
- * Some uses of botnet:
 - Distributed DoS
 - Spamming
 - Sniffing traffic
 - Keylogging
 - Spreading new malware
 - Manipulating online polls/games

Payload – Information Theft

- * Some malware (keyloggers, phishing, spyware) gathers data stored on the infected system for use by the attacker
 - Login and password, Bank account, Gaming, etc.
 - These attacks target the information confidentiality.
- * The attacker installs keylogger that captures keystrokes on the system
 - This allows the attacker get the login and password even when they are sent over encrypted channels (e.g., HTTPS).
 - Keyloggers can return only desired keywords using some form of filtering mechanism
 - * e.g., login, password, paypal.com
 - Countermeasure: use graphical applet to enter critical information that cannot be captured by the traditional keyloggers
 - More general spyware can monitor a wide range of activity on the victim

Payload – Information Theft

- * The attacker can send a SPAM with URL that links to a fake Web site similar to some banking → phishing attack
 - Careless users may follow the link and provide their critical information
 - Spear phishing attack is targeting better researched victims that include information specific to the victims

Payload – Stealthing

- * Some malware hides its existence on the victim's machine but provides covert access to that system
 - backdoor, rootkit
- * A backdoor or trapdoor is a secret entry into a program that allows the attacker to gain access without going through regular security check
 - Programmers use this backdoor, called maintenance hook, to debug and test program
 - It is difficult to implement OS control for backdoor in applications
- * A rootkit is a malware with supervisory access rights
 - It has access to all OS services and functions
 - It can add/change programs and files, monitor processes, and hide
 - It hides by corrupting the mechanisms for monitoring processes, files, and registries on the computer

Countermeasures

- * The best solution: prevention
 - do not allow malware to get into the system or prevent the malware to modify the system
 - In general almost impossible
- * Four elements of prevention (NIST): policy, awareness, vulnerability mitigation, and threat mitigation.
- * If prevention fails, technical mechanism can be used for threat mitigation:
 - Detection: know that infection has occurred and where it is located
 - Identification: find out the specific malware
 - Removal: remove all traces of the infection to prevent further spread
- * If detection succeeds but identification and removal fail, replace all infected files with clean files from backup

Requirements for countermeasures

- * Generality: the approach can address a wide variety of attacks
- * Timeliness: it should respond quickly
- * Resiliency: it is resistant against the attacker's hiding technique
- * Minimal denial of service cost: it does not significantly reduce the system capacity and disrupt normal operation
- * Transparency: it does not require modification to application and system software as well as hardware
- * Global and local coverage: it can deal with attack from both inside and outside the network

Where to run antivirus program?

- * Run some host-based antivirus program on the infected system
- * Run antivirus on the perimeter security mechanisms in the firewall or as part of the intrusion detection mechanism
- * Use distributed mechanism that gather data from both host-based and perimeter

Host-based scanners

First generation: simple scanners

- * Require a malware signature to identify it
- * Can detect only known malware
- * The scanner may keep the length of programs and looks for changes in length

Host-based scanners

Second generation: heuristic scanners

- * It does not rely on malware signature but uses heuristic rules to search for probable malware instances
- * It looks for fragments of code that are often associated with malware
- * It uses integrity checking
 - It may add checksum to each program
 - * if malware modifies the program without changing the checksum, it will be detected.
 - More sophisticated malware is able to change the checksum when it modify the program
 - * Counter this using encrypted hash function with the encryption key stored somewhere else

Host-based scanners

Third generation: activity traps

- ★ It is memory resident program that identify malware from its action rather than its structure
- ★ It does not need signatures or heuristics for wide variety of malware
- ★ It needs to identify small set of actions that indicate malicious activity

Host-based scanners

Fourth generation: full-featured protection

- * It includes scanning and activity traps
- * It also includes access control capability

IE2022 - Introduction to Cyber Security

Lecture - 11

Legal and Ethical aspects of Information Security

Mr. Amila Senarathne

Topics to be discussed

- * Program and data protection by patents, copyrights, and trademarks
- * Computer crime
- * Ethical analysis of computer security situations
- * Codes of professional ethics

References

- * Security in Computing - Legal and Ethical Issues in Computer Security
- * Other related materials

Data Acquisition

- * Data acquisition is the process of obtaining data from a digital device using peripheral equipment and media.
- * There are two types of acquisition;
 - Static acquisition
 - Live acquisition

Law and Computer Security

- * International, national, state, and city laws can affect privacy and secrecy
 - These statutes often apply to the rights of individuals to keep personal matters private.
- * Laws regulate the use, development, and ownership of data and programs
 - Patents, copyrights, and trade secrets are legal devices to protect the rights of developers and owners of programs and data.
- * Laws affect actions that can be taken to protect the secrecy, integrity, and availability of computer information and service

Challenges

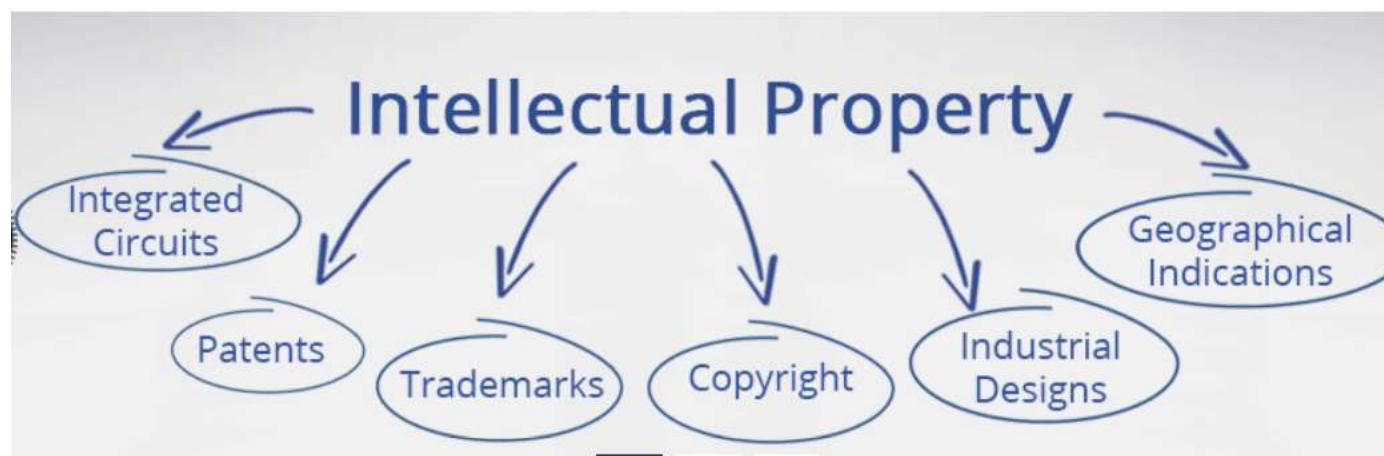
- * Law does not always provide an adequate control
- * Laws do not yet address all improper acts committed with computers
- * Some judges, lawyers, and police officers do not understand computing, so they cannot determine how computing relates to other, more established, parts of the law
- * Lack of technical expertise of legal personnel

Protecting Programs and Data

- * Common legal devices include:
 - Copyrights
 - Patents
 - Trade Secrets
- * Copyrights, patents, and trade secrets are legal devices that can protect computers, programs, and data. However, in some cases, precise steps must be taken to protect the work before anyone else is allowed access to it.

Sri Lankan Context

- * INTELLECTUAL PROPERTY ACT, No. 36 OF 2003
 - The National Intellectual Property Office of Sri Lanka established under the Intellectual Property Act No 36 of 2003 is mandated with the administration of the intellectual Property System in Sri Lanka.



World Intellectual Property Organization

- * WIPO is one of the 15 specialized agencies of the United Nations (UN).
- * WIPO was created in 1967 "to encourage creative activity, to promote the protection of intellectual property throughout the world"
- * WIPO currently has 192 member states, administers 26 international treaties
- * Headquartered in Geneva, Switzerland.

Copyrights

- * Designed to protect the expression of ideas
 - Copyright applies to a creative work, such as a book, story, photograph, song, or pencil sketch. The right to copy an expression of an idea is protected by a copyright.
- * Ideas are free but once expressed (in a tangible medium) must be protected
- * Intention of a copyright is to allow regular and free exchange of ideas
 - The law protects an individual's right to earn a living, while recognizing that exchanging ideas supports the intellectual growth of society.
- * Gives the author the exclusive right to make copies of the expression and sell them to the public
 - copyright law also has the concept of a first sale; after having bought a copyrighted object, the new owner can give away or resell the object

Copyrights

- * Copyright must apply to original work
- * It lasts for few years after which it is considered public domain
- * Copyright object is subject to fair use
- * Product used in a manner for which it was intended and does not interfere with the author's rights, e.g. comment, criticism, teaching, scholarly research
- * Unfair use of copyrighted object is called piracy

Copyrights

- * A U.S. copyright now lasts for 70 years beyond the death of the last surviving author, 95 years after the date of publication for organizations
- * The international standard is 50 years after the death of the last author or 50 years from publication
 - World Intellectual Property Organization treaty of 1996, an international copyright standard to which lot of countries adhere.
- * Duration of copyright in Sir Lanka
 - Lifetime of the author and for a further period of 70 years from the date of his death (p.m.a.); a work of applied art is protected for 25 years from the date of the making of the work.

Copyrights for Computer Software

- * Computer program can be copyrighted (depending on the law of the country)
- * Copying the code intact is prohibited
- * Challenge 1
 - Algorithm is the idea, and the statements of the programming language are the expression of the idea
 - Protection is allowed for the program statements themselves, but not for the algorithmic concept
 - Copying the code intact is prohibited, but re-implementing the algorithm is permitted

Copyrights for Computer Software

* Challenge 2

- copyright protection for computer works is the requirement that the work be published.
- A program may be published by distribution of copies of its object code, for example, on a disk. However, if the source code is not distributed, it has not been published.
- An alleged infringer cannot have violated a copyright on source code if the source code was never published.

Digital Millennium Copyright Act of 1998

- * The Digital Millennium Copyright Act (DMCA) is a 1998 United States copyright law that implements two 1996 treaties of the World Intellectual Property Organization (WIPO).
 - Digital objects can be subject to copyright
 - It is a crime to circumvent or disable antipiracy functionality built into an object
 - It is a crime to manufacture, sell, or distribute devices that disable antipiracy functionality or that copy digital objects

Digital Millennium Copyright Act of 1998

- * However, these devices can be used (and manufactured, sold, or distributed) for research and educational purposes
- * It is acceptable to make a backup copy of a digital object as a protection against hardware or software failure or to store copies in an archive
- * Libraries can make up to three copies of a digital object for lending to other libraries
- * Problems is deciding what is considered piracy
 - Example, how do you transfer music from your CD to MP3 which is considered a reasonable fair use?

Patents

- * Patent Office must be convinced that the invention deserves a patent
- * Patents were intended to apply to the results of science, technology, and engineering
 - whereas copyrights were meant to cover works in the arts, literature, and written scholarship. (distinction between patents and copyrights)
- * A patent can be valid only for something that is truly novel or unique
 - usually one patent for a given invention
- * Patent law has expanded to include computer software
 - Recognizing that algorithms, like processes and formulas, are inventions.

Patent Infringement

- * Copyright: holder can decide which violations prosecute
- * Patent: all violations must be prosecuted or patent can be lost
- * Suing for patent infringement may cause the patent owner to lose the patent. Infringer may argue that:
 - This isn't infringement (different inventions)
 - The patent is invalid (a prior infringement was not opposed)
 - The invention is not novel
 - The infringer invented the object first

Applicability of Patents to Computer Objects

- * Patent law has expanded to include computer software
 - Recognizing that algorithms, like processes and formulas, are inventions.
- * Patent Offices have issued thousands of software patents
- * One of the most desired protection for software with value algorithms to protect
- * Because of the time and expense involved in obtaining and maintaining a patent, this form of protection may be unacceptable for a small-scale software writer.

Trade Secrets

- * A trade secret is information that gives one company a competitive edge over others
- * Unlike a patent or copyright it must be kept a secret
 - Employees should not disclose secrets
 - Owners must protect the secrets
 - File encryption
 - Make employees sign a statement not to disclose a secret

Trade Secrets : Applicability to Computer Objects

- * Trade secret protection allows distribution of the result of a secret (the executable program) while still keeping the program design hidden
- * It does not cover copying a product (specifically a computer program)
- * It makes it illegal to steal a secret algorithm and use it in another product
- * If someone obtains it improperly, the owner can recover
 - Profits
 - Damages
 - Lost revenues
 - Legal cost

Trade Secrets: Applicability to Computer Objects

- * Applies very well to computer software (can protect the algorithm as a secret)
- * Difficulty with computer programs is that reverse engineering works.
 - Decompiler and disassembler programs can produce a source version of an executable program.
- * Enforcement Problems
- * Does not help if program/code is decoded – trade secret protection disappears
 - Additional protection/safeguards is needed
 - Make copies of sensitive documents
 - * Control access to files
 - * Non-disclosure agreements

Copyright, Patent and Trade Secrets

	Copyright	Patent	Trade Secret
Protects	Expression of idea, not idea itself	Invention—the way something works	A secret, competitive advantage
Protected object made public	Yes; intention is to promote publication	Design filed at Patent Office	No
Requirement to distribute	Yes	No	No
Ease of filing	Very easy, do-it-yourself	Very complicated; specialist lawyer suggested	No filing
Duration	Life of human originator plus 70 years, or total of 95 years for a company	19 years	Indefinite
Legal protection	Sue if unauthorized copy sold	Sue if invention copied	Sue if secret improperly obtained

Protecting Computer Objects

- * Hardware
 - Patented
- * Firmware – Chips and microcode
 - Patented
 - Data (algorithms, instructions and programs inside it) are not patentable
 - Trade secret – for code inside chip
- * Object Code Software
 - Copyrighted

Protecting Computer Objects

- * Source Code Software

- Trade secret
 - Copyrighted

- * Documentation

- Copyrighted

- * Web Content

- Copyrighted

Computer Crime

- * Computer crime (Cybercrime) is defined as a crime where a computer is the object of the crime or is used as a tool to commit an offense.
- * Least clear area of law in computing
- * Separate category for computer crime
 - No access to the physical object → Is it a serious crime?
 - Rules of evidence → How to prove the authenticity?
 - Threats to integrity and confidentiality → How to measure loss of privacy?
 - Value of data → How to measure it?

Why a Separate Category for Computer Crime Is Needed

- * Law regarding crimes involving computers are less clear
- * New laws needed to address these problems
- * Rules of property
 - Unauthorized access to a computing system is a crime
 - Problem is access by a computer does not involve physical object so may not be punishable crime
- * Rules of Evidence
 - Courts prefer an original source document to a copy
 - Copies may be inaccurate or modified
 - Problem with computer-based evidence in court is being able to demonstrate the authenticity of the evidence

Copyrights

- * Copyright must apply to original work
- * It lasts for few years after which it is considered public domain
- * Copyright object is subject to fair use
- * Product used in a manner for which it was intended and does not interfere with the author's rights, e.g. comment, criticism, teaching, scholarly research
- * Unfair use of copyrighted object is called piracy

Copyrights

- * A U.S. copyright now lasts for 70 years beyond the death of the last surviving author, 95 years after the date of publication for organizations
- * The international standard is 50 years after the death of the last author or 50 years from publication
 - World Intellectual Property Organization treaty of 1996, an international copyright standard to which lot of countries adhere.
- * Duration of copyright in Sir Lanka
 - Lifetime of the author and for a further period of 70 years from the date of his death (p.m.a.); a work of applied art is protected for 25 years from the date of the making of the work.