BIT

1 : Information Security Concepts

IT5306 - Principles of Information Security

Level III - Semester 5

UCSC

eLC

List of sub topics

1.1 Computer Security Concepts: Confidentiality, Integrity, and Availability

1.2. Threats, Attacks, and Assets

1.3. Security Functional Requirements

1.4. Fundamental Security Design Principles

1.5. Attack Surfaces and Attack Trees

1.6. Computer Security Strategy

1.7. Concepts of Encryption, Decryption, Plain Text and Cipher Text

1.8. Stream and Block Ciphers

Introduction

- At one time Bank robbery was common. Now its very rare. What has changed or been implemented to provide this security?
  - Sophisticated alarms
  - Criminal investigation techniques (DNA testing)
  - Change in “assets” (cash was/is inherently insecure)
  - Improvements in communication and transportation

- Risk becomes so high that it is no longer beneficial.

Introduction

- In our case the "valuables" are computer related assets instead of money
  - Though these days money is so electronic that one can argue that the protection of money is a subset of computer asset security

- Information seems to be the currency of the 21st century.

Introduction

- **Size and portability**
  - Banks are large and unportable.
  - Storage of information can be very small and extremely portable. (So small that an entire corporations intellectual property can be stored on something the size of a postage stamp.)Ability to avoid physical contact
  - Banks: physical interaction with the bank and the loot is unavoidable or impossible to circumvent
  - Computers: require no physical contact to either gain access to, copy or remove data.
- **Value of assets:**
  - Bank: generally very high (or why would somebody bother to put it in a bank?)
  - Computers: Variable, from very low (useless) to very high.

Information security
are methods and technologies
for protection, integrity, availability,
authenticity and extended functionality
of computer programs and data

Threats, Attacks and Assert

- **Method:** The skills knowledge and tools that enable the attack
- **Opportunity:** The time, access and circumstances that allow for the attack
- **Motive:** The reason why the perpetrator wants to commit the attack

People

Amateurs . . .
Crackers
Criminals
Regular users

Accidental access
to unauthorized resources
and execution of
unauthorized operations
(no harm to regular users)

People

- Amateurs
- **Crackers . . .**
- Criminals
- Regular users

Active attempts to access sensitive resources and to discover system vulnerabilities
(minor inconveniences
to regular users)

People

Amateurs
Crackers
Criminals . . .
Regular users

Active attempts to utilize weaknesses in protection system in order to steal or destroy resources (serious problems to regular users)

People

Amateurs

Crackers

Criminals

Regular users . . .

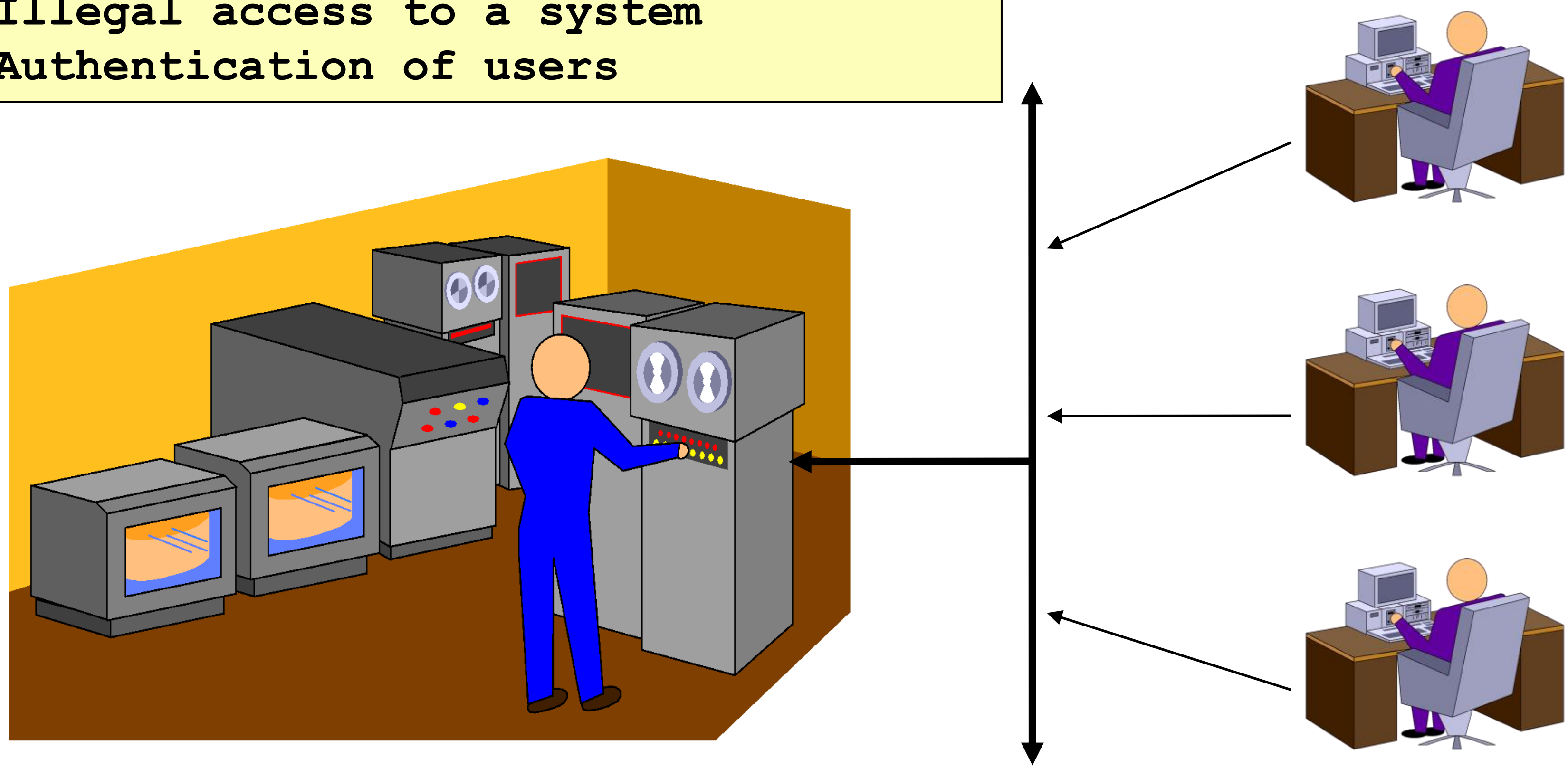
Special requirements: authentication in open networks, authorization, message integrity, non-repudiation, special transactions

Threats, Attacks and Assert

- Vulnerability: A weakness in the security system.
- Attack: A human exploitation of a vulnerability.
- Control: A protective measure. An action, device or measure taken that removes, reduces or neutralizes a vulnerability.
- Problems : Consequences of unintentional accidental errors
- Threat: a set of circumstances that has the potential to cause loss or harm.
- Risks : Probabilities that some threat or problem will occur due to system vulnerabilities

Threats, Attacks and Assert

- Illegal access to a system
- Authentication of users



Attack Types: An Active Attack

- Active attacks are attacks in which the attacker attempts to change or transform the content of messages or information.

- These attacks are a threat to the integrity and availability of the system.

- Due to these attacks, systems get damaged, and information can be altered.

- The prevention of these attacks is difficult due to their high range of physical and software vulnerabilities.

Attack Types: A Passive Attack

- Passive attacks are the ones in which the attacker observes all the messages and copy the content of messages or information.
- They focus on monitoring all the transmission and gaining the data.
- The attacker does not try to change any data or information he gathered.
- Although there is no potential harm to the system due to these attacks, they can be a significant danger to your data’s confidentiality.

Active vs Passive Attacks

- In active attacks, modification of messages is done, but on the other hand, in passive attacks, the information remains unchanged.

- The active attack causes damage to the integrity and availability of the system, but passive attacks cause damage to data confidentiality.

- In active attacks, attention is given to detection, while in the other one, attention is given to prevention.

- The resources can be changed in active attacks, but passive attacks have no impact on the resources.

Active vs Passive Attacks

- The active attack influences the system services, but the information or data is acquired in passive attacks.

- Inactive attacks, information is gathered through passive attacks to attack the system, while passive attacks are achieved by collecting confidential information such as private chats and passwords.

- Active attacks are challenging to be prohibited, but passive attacks are easy to prevent.

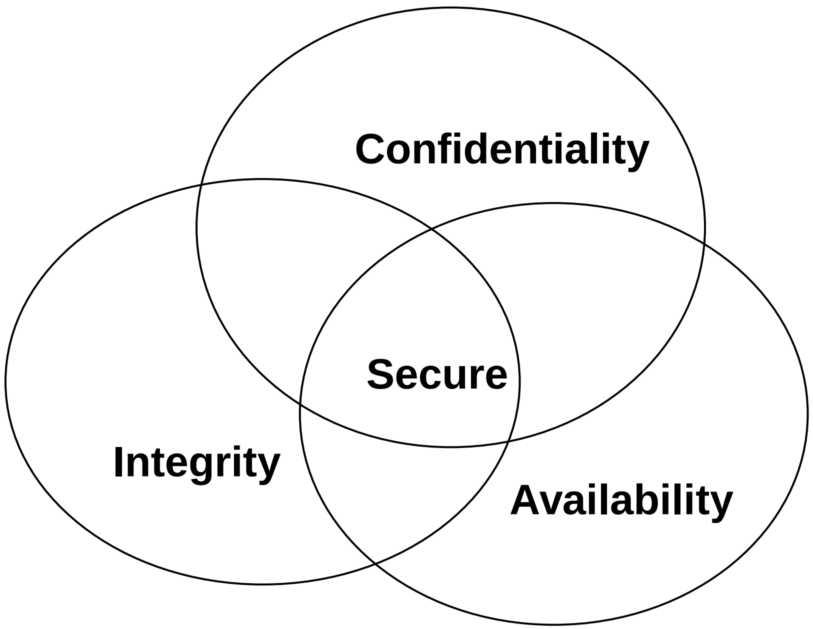
Security Functional Requirements

- What makes a “secure” system?
  - Financial “Security” requirements
  - Home “security”
  - Physical “security”
  - Information “security”

- All these concepts of security have different requirements. We are, of course, interested mostly on computer security; which requires three items:

Security Functional Requirements

- The presence of all three things yields a secure system:



Security Functional Requirements

- Confidentiality:

  Computer related assets are only available to authorized parties. Only those that should have access to something will actually get that access.

  - “Access” isn't limited to reading. But also to viewing, printing or…
  - Simply even knowing that the particular asset exists (steganography)

- Straight forward concept but very hard to implement.

Security Functional Requirements

- Integrity
  Can mean many things: Something has integrity if it is:
  - Precise
  - Accurate
  - Unmodified
  - Consistent
  - Meaningful and usable

Security Functional Requirements

- Three important aspects towards providing computer related integrity:
  - Authorized actions
  - Seperation and protection of resources
  - Error detection and correction.

- Again, rather hard to implement; usually done so through rigorous control of who or what can have access to data and in what ways.

Security Functional Requirements

- Availability
  - There is a timely response to our requests
  - There is a fair allocation of resources (no starvation)
  - Reliability (software and hardware failures lead to graceful cessation of services and not an abrupt crash)
  - Service can be used easily and in the manner it was intended to be used.
  - Controlled concurrency, support for simultaneous access with proper deadlock and access management.

Security Design Principles

Confidentiality . . .
Integrity
Availability
Functionality

Threats to Data and Programs:
illegal read, illegal access, data (files) deletion, illegal users, criminal acts, sabotage, etc.

Security Design Principles

- Confidentiality
- **Integrity . . .**
- Availability
- Functionality

Threats to software and data: technical errors, software errors, processing errors, transmission correctness, etc.

Security Design Principles

Confidentiality
Integrity
Availability . . .
Functionality

Requirements for:
timely response, fair allocation, fault tolerance, usability, controlled concurrency

Security Design Principles

Confidentiality
Integrity
Availability
Functionality . . .

New functions needed for electronic data transactions: authentication, digital signature, confidentiality, and others

Attack Surface

- The entire system:
  - Hardware
  - Software
  - Storage media
  - Data
  - Memory
  - People
  - Organizations
  - Communications

Attack Surface

An attack surface consists of reachable and exploitable vulnerabilities in a system and can be classified into three categories:

Network attack surface refers to vulnerabilities over an enterprise network or the internet. Examples of this include network protocol vulnerabilities, such as those used for a DDoS attack, disruption of communication links and various forms of intruder attacks.

Software attack surface refers to vulnerabilities in application, utility, or operating system code. A particular focus in this category is web server software.

Human attack surface refers to vulnerabilities created by personnel or outsiders, such as social engineering, human error and trusted insiders.

Attack trees

- An attack tree is a tree structure that represents attacks against a system.

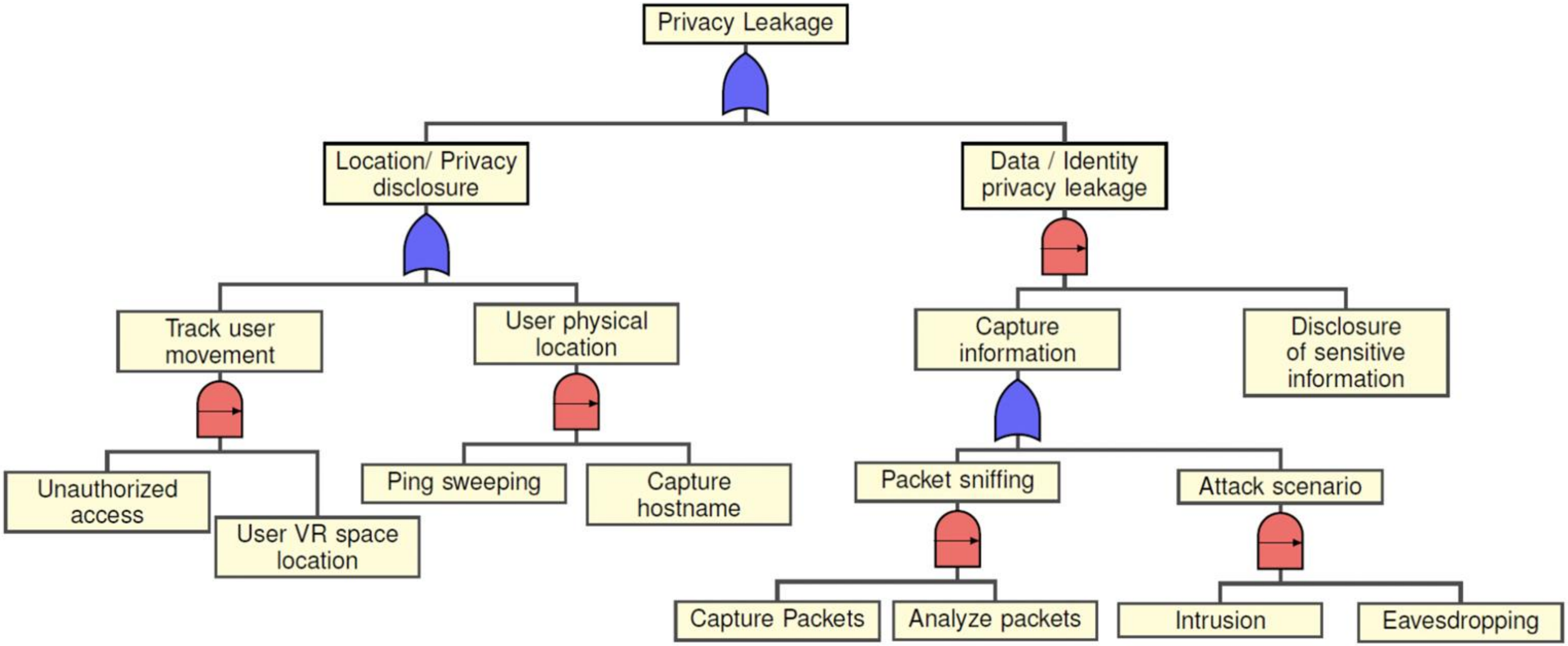
- Each subnode defines a subgoal, and each subgoal may, in turn, have its own set of subgoals, and so on.

- The leaf nodes of the attack tree represent different ways to initiate an attack.

- Each node other than a leaf is either an AND-node or an OR-node.

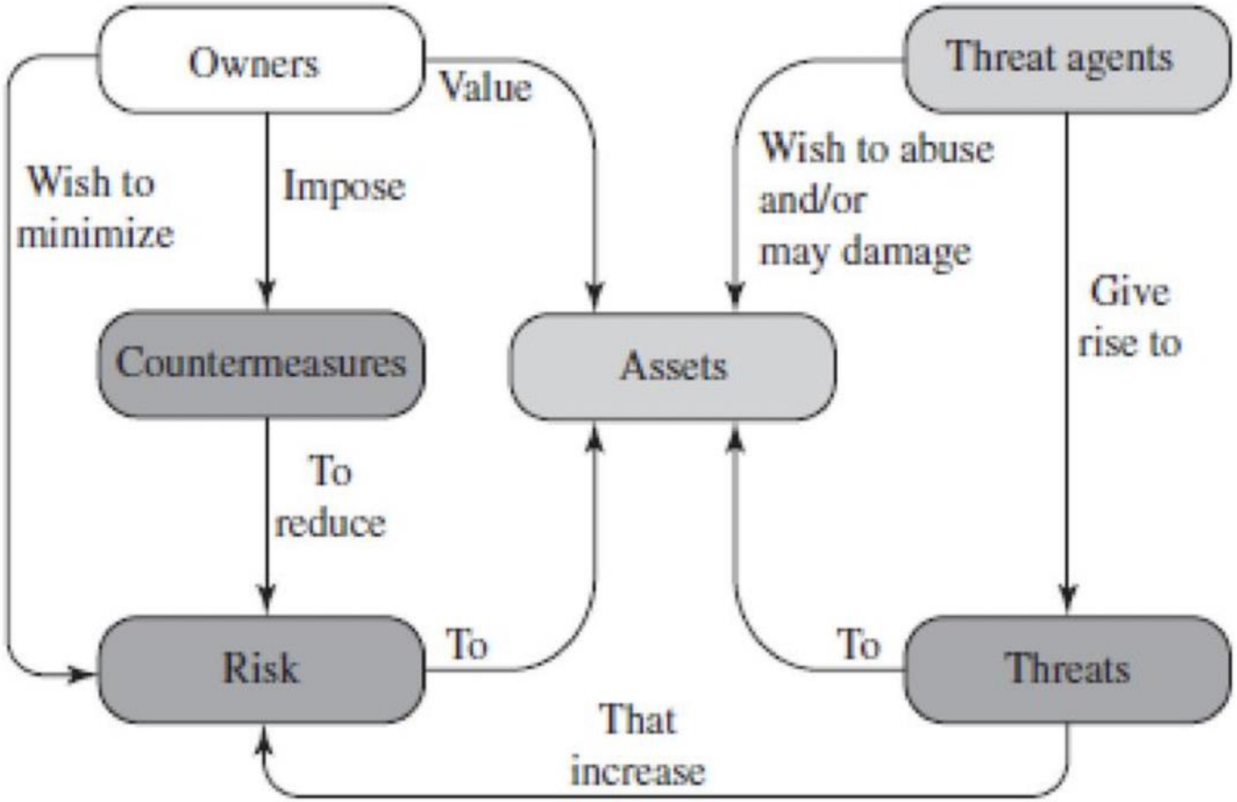
- To achieve the goal represented by an AND-node, all of the child subgoals must be achieved; and for an OR-node, at least one of the child subgoals must be achieved.

Privacy Attack Tree with Threat Scenarios



Computer Security Strategy

A Model for Computer Security



Computer Security Strategy

- Encryption
- SW & HW Controls
- Policies
- Physical controls

Computer Security Strategy

Encryption . . .	**Effective for:** **confidentiality,** **users and messages** **authentication,** **access** **control**
SW & HW Controls	
Policies	
Physical controls	

Computer Security Strategy

Encryption

SW & HW Controls

Policies

Physical controls

Available methods: software and hardware controls (internal SW, OS controls, development controls, special HW devices)

Computer Security Strategy

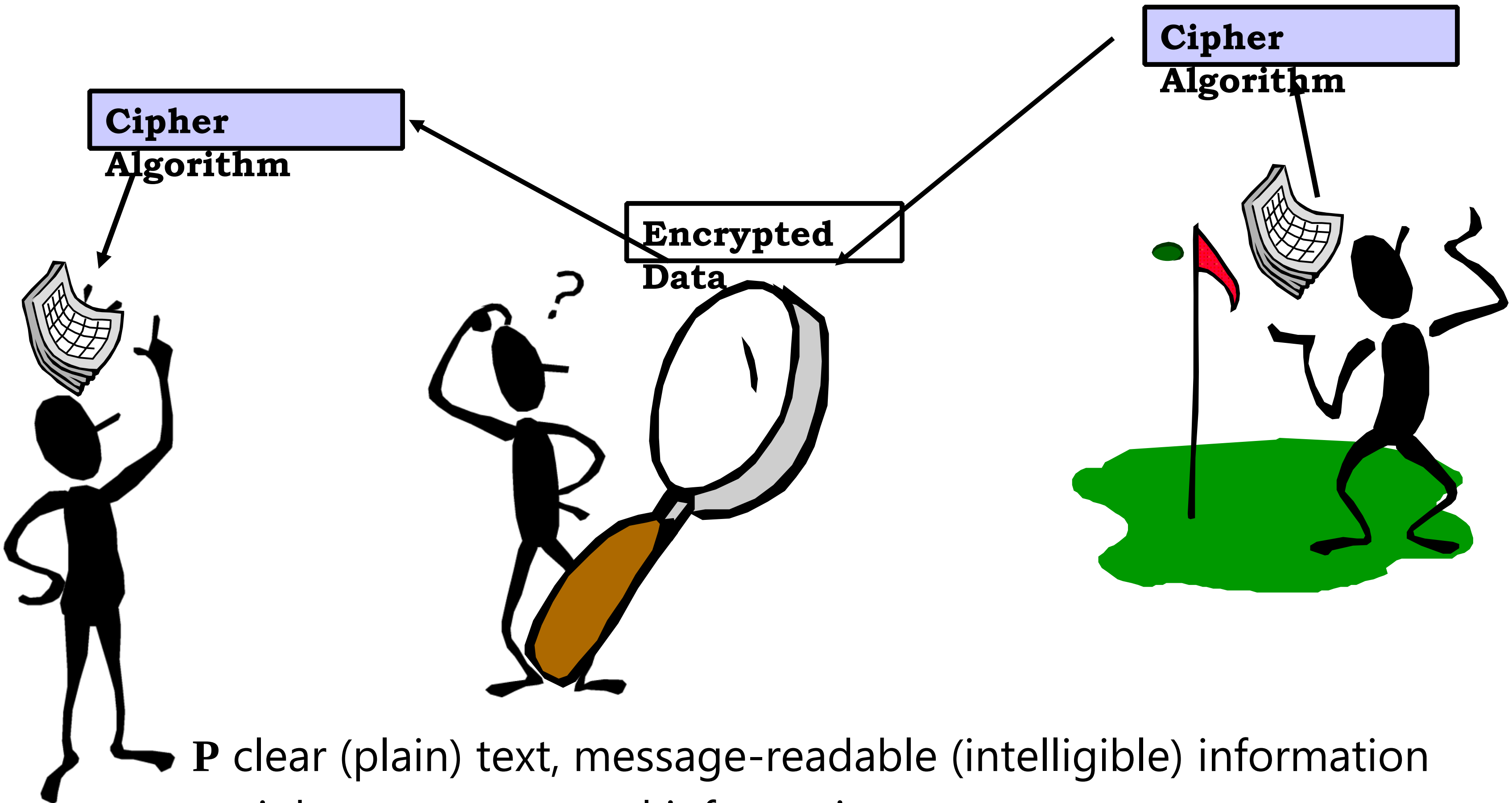
Encryption	**Precise specifications:** **special procedures,** **security methods,** **security parameters,** **organizational issues**
SW & HW Controls	
Policies . . .	
Physical controls	

Computer Security Strategy

Encryption
SW & HW Controls
Policies
Physical controls

Measures for:
isolation of equipment,
access to equipment,
authorization for personnel,
backup and archiving

Concept of Cryptography



P clear (plain) text, message-readable (intelligible) information

C ciphertext-encrypted information

E encryption (enciphering)-transforming clear text into ciphertext

D decryption (deciphering)-transforming ciphertext back into plaintext

Concept of Cryptography

- **Why Encrypt?**
  - Protect stored information
  - Protect information in transmission

- Cryptography originally used for secrecy

- **Encryption** - process by which **plaintext** is converted to **ciphertext** using a **key**

- **Decryption** - process by which ciphertext is converted to plaintext (with the appropriate key)

- **plaintext** (cleartext)- intelligible data

Concept of Cryptography

- **Historic examples…**
  - Earliest cryptography: an Egyptian scribe using non-standard hieroglyphics
  - Julius Caesar ("Caesar Cipher")
    Each plaintext letter is replaced by a letter some fixed number of positions further down the alphabet (e.g. Belgica (3 positions) → ehojlfd)
  - The Kama Sutra recommends cryptography as 44^{th} and 45^{th} art
    (of 64) men and women should know

Concept of Cryptography

- ENIGMA Used by the Germans in WW2 – and the subsequent
  code-breaking activities at Bletchley park
  (still a popular subject of books and movies)

- 1976: Public Key Cryptography concept
  (Whitfield Diffie & Martin Hellman)

- 1977: first (*published*) practical PKC cryptosystem invented
  (RSA - Rivest, Shamir, Adleman)

- October 2000 Rijndael is chosen as AES
  (Advanced Encryption Standard)

The Caesar Cipher

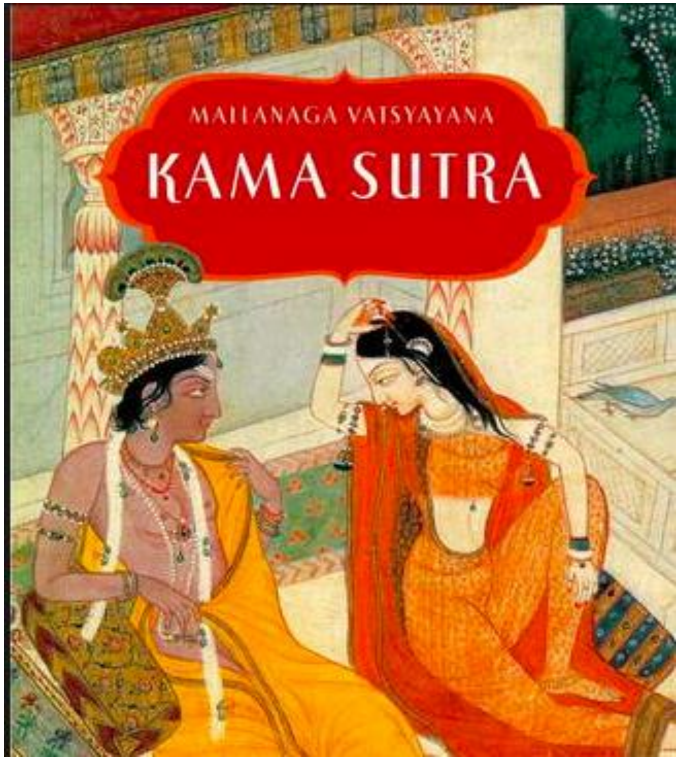
Plain Text : A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Cipher Text : 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

$$C_i = E(P_i) = P_i + 3$$

Kamasutra

One of the earliest descriptions of encryption by substitution appears in the Kama-sutra, a text written in the 4th century AD by the Brahmin scholar Vatsyayana, but based on manuscripts dating back to the 4th century BC.



How it work

The kamasutra generate list of 26 alphabet with no duplicate. Then divide by 2 row. Find for each letter of message text in table and choose the opposite of the letter

Kamasutra

for example:
Key = G H A J R I O B E S Q C L F V Z T Y K M X W N U D P

divide by 2 rows
G H A J R I O B E S Q C L
F V Z T Y K M X W N U D P

Given String = KAMASUTRA
K is at 2nd row and 5th column. Get the opposite of K that is I.
Do each letter until the end

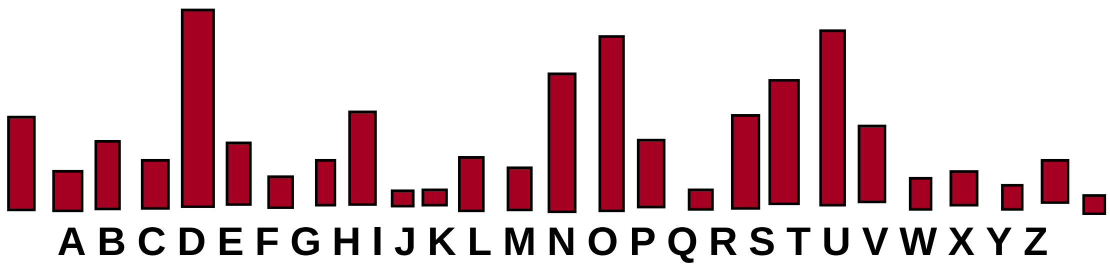
Cipher : IZOZNQJYZ

Polyalalphaberic Substitutions

Plain Text : 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

Cipher Text : K E Y G H I J K L M N O P Q R S T U V W X Y Z A B C

<u>Letter Frequency</u>



Polyalalphaberic Substitutions

Table for Odd Positions

Plain Text : 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
Cipher Text : A D G J N O S V Y B E H K N Q T W Z C F I L O R U X

Table for Even Positions

Plain Text : 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
Cipher Text : N S X C H M R W B G I Q V A F K P U Z E J O T Y D I

Plain Text : SSIBL
Cipher Text : czysh

Transposition (Permutation) Substitutions

Columnar Transposition

c1	c2	c3	c4	c5
c6	c7	c8	c9	c10
c11	c12	etc.		

Cipher text formed by → **c1 c6 c11 c2 c7 c12 c3 c8 ...**

c1	c2	c3	c4	c5
c6	c7	c8	c9	c10
c11	c12	etc.		

The Vernam Cipher

Plain Text	: V E R N A M C I P H E R
Numeric Equivalent	: 21 4 17 13 0 12 2 8 15 7 4 17
+Random Number	: 76 48 16 82 44 3 58 11 60 5 48 88
= Sum	: 97 52 33 95 44 15 60 19 75 12 52 105
=Mod 26	: 19 0 7 17 18 15 8 19 23 12 0 1
Cipher text	: t a h r s p I t x m a b

Binary Vernam Cipher

Plain Text	: 1 0 1 0 0 0 1 1 1 0 0 1 1 0 1
$\oplus$ Random Stream	: 0 1 0 1 1 0 1 0 1 1 1 0 1 0 1
Cipher text	: 1 1 1 1 1 0 0 1 0 1 1 1 0 0 0

The One-Time Pad

- If a truly random key as long as the message is used, the cipher will be secure
- Called a **One-Time pad**
- Has unconditional security:
- ciphertext bears no statistical relationship to the plaintext since for **any plaintext** & **any ciphertext** there exists a key mapping one to other
- Can only use the key **once**
- Have problem of safe distribution of key

Stream and Block Ciphers

- Stream Ciphers - Message broken into characters or bits and enciphered with a “key stream”
  - key stream - should be random and generated independently of the message stream

- Block ciphers process messages in blocks, each of which is then en/decrypted

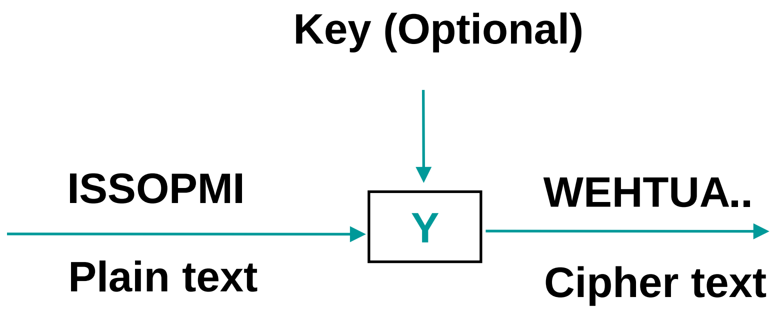
Stream and Block Ciphers

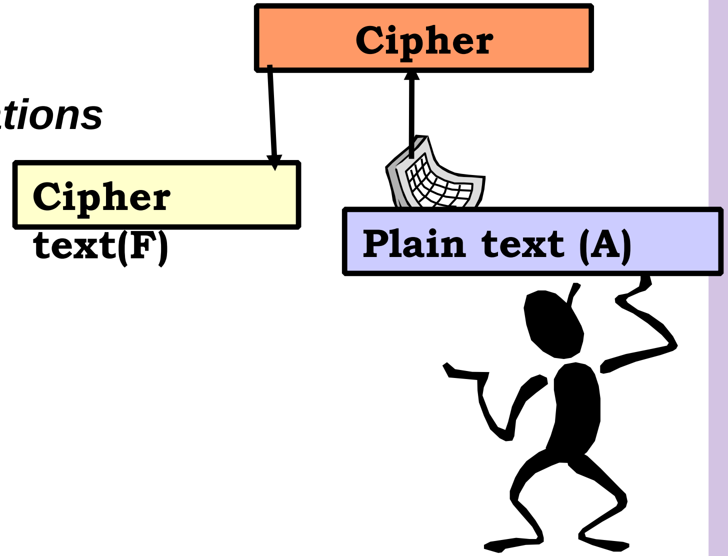
<u>**Advantage**</u>

- ***Speed of transformation***
- ***Low error propagation***

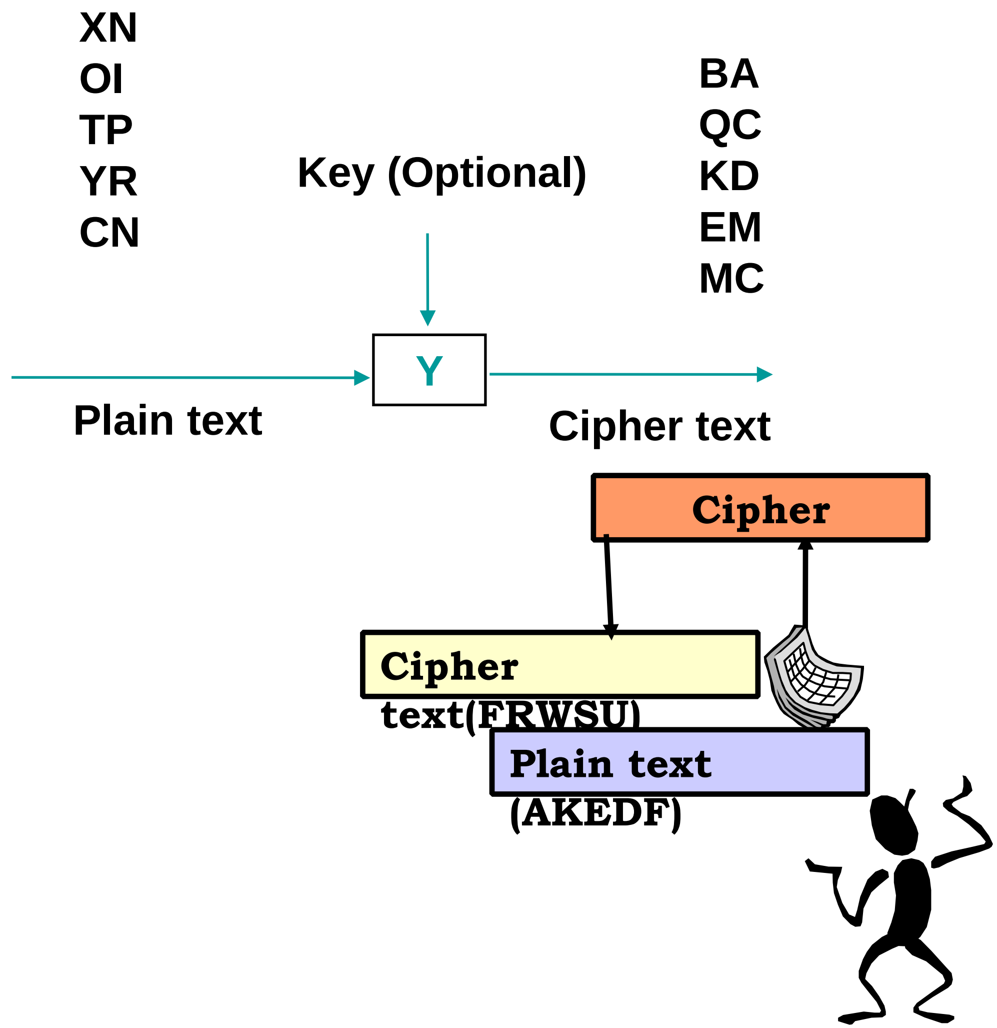
<u>**Disadvantage**</u>

- ***Low diffusion***
- ***Susceptibility to malicious insertion and modifications***





Stream and Block Ciphers



Disadvantage

- *Slowness of encryption*
- *Error propagation*

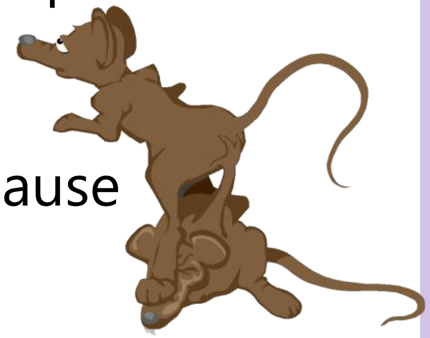
Advantage

- *Diffusion*
- *Immunity to insertion*

Characteristic of "GOOD" Cipher

<u>Shannon Characteristics - 1949</u>

- The amount of secrecy needed should determine the amount of labor appropriate for encryption and decryption
- The set of keys and the encryption algorithm should be free from complexity
- The implementation of the process should be as simple as possible
- Errors in the ciphering should not propagate and cause corruption of further information in the message
- The size of enciphered text should be no larger than the text of the original message



Kerckhoff’s Principle

The security of the encryption scheme must depend only on *the secrecy of the key and not on the secrecy of the algorithms.*

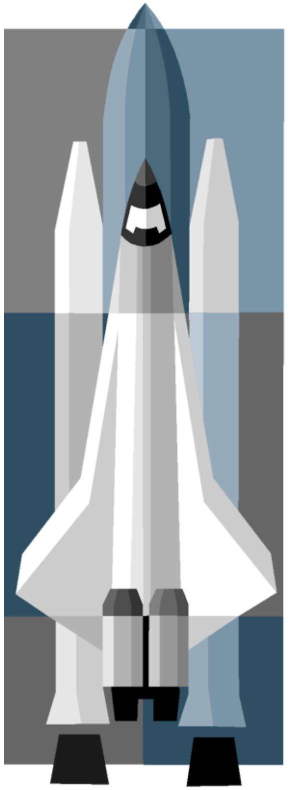
Reasons:

- Algorithms are difficult to change
- Cannot design an algorithm for every pair of users
- Expert review
- No security through obscurity!

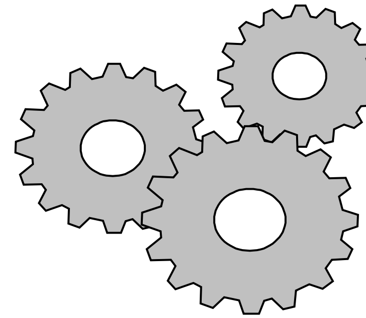
Brute Force Search

- **Always possible to simply try every key**
- **Most basic attack, proportional to key size**
- **Assume either know/recognize plaintext**

Key Size (bits)	Number of Alternative Keys	Time required at 10^6 Decryption/µs
32	$2^{32} = 4.3 \times 10^9$	2.15 milliseconds
56	$2^{56} = 7.2 \times 10^{16}$	10 hours
128	$2^{128} = 3.4 \times 10^{38}$	5.4×10^{18} years
168	$2^{168} = 3.7 \times 10^{50}$	5.9×10^{30} years



Unconditional/Computational Security



Unconditional security

no matter how much computer power is available, the cipher cannot be broken since the ciphertext provides insufficient information to uniquely determine the corresponding plaintext

Computational security

given limited computing resources (e.g. time needed for calculations is greater than age of universe), the cipher cannot be broken

Sec_rity is not Complete without U

You, as a Computer User, have to make your contribution to computer security: **You are responsible for the security and protection** of your computers, the operating systems you run, the application you install, the software you program, the data you own - and the services and systems you manage.

Thank You