# Week 10: Network Security

## Internet Technologies COMP90007

Lecturer: Muhammad Usman

Semester 2, 2020

# What is Network Security?

- Network security is a combo of 4 related areas:
  - **Secrecy** (Keeping information hidden from a general audience)
  - **Authentication** (Ensuring the user you are giving content to has valid credentials)
  - **Non-repudiation** (Prove a content was created by a named user)
  - **Integrity control** (Ensure that a content has not been tampered with)
- All of these are **equally valid** and has been around for all systems for some time, but have different and sometimes more challenging implications in a networked environment
- Aspects of security can be found at all layers of a protocol stack, **there is no way to secure a network by building security into one layer only**
- Most security implementations are **based on common cryptographic principles** and appear on almost all layers

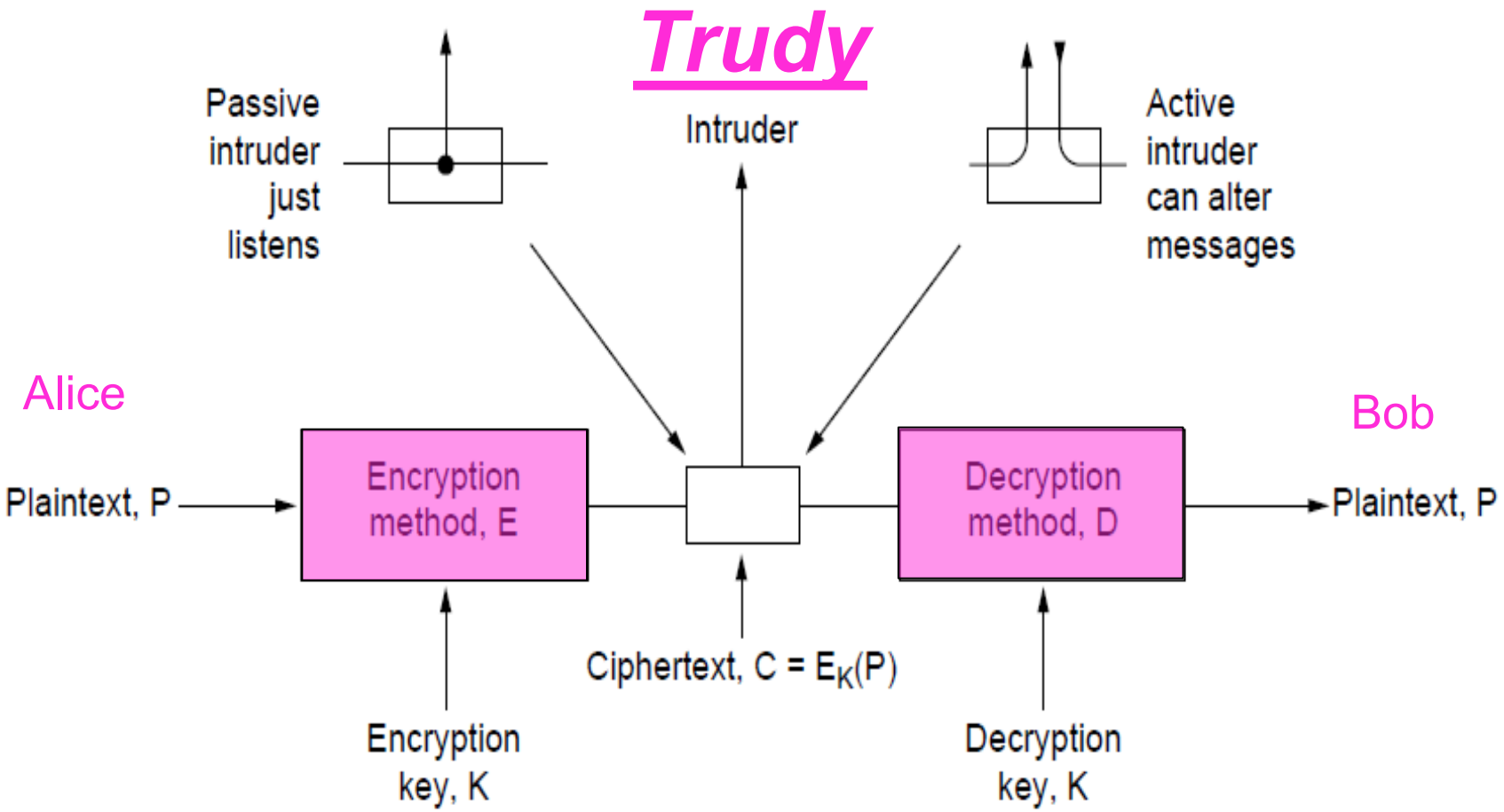
# Some Well-Known Characters..

| Adversary | Goal |
|---|---|
| Student | To have fun snooping on people's email |
| Cracker | To test out someone's security system; steal data |
| Sales rep | To claim to represent all of Europe, not just Andorra |
| Businessman | To discover a competitor's strategic marketing plan |
| Ex-employee | To get revenge for being fired |
| Accountant | To embezzle money from a company |
| Stockbroker | To deny a promise made to a customer by email |
| Con man | To steal credit card numbers for sale |
| Spy | To learn an enemy's military or industrial secrets |
| Terrorist | To steal germ warfare secrets |

# Cryptography

- A key area/set of **algorithms** for creating secrets, authenticating users, making sure messages are not tampered with, and edits are not denied by the original author….

# Encryption Model



# Key Cryptography Concepts

- Three foundations:
  - Plaintext
  - Keys
  - Ciphertext
- Plaintext messages to be encrypted can be transformed (encrypted/decrypted) by a function that is parameterized by a key, the output of the transformation process is ciphertext
- Kerckhoff's principle: Cryptographic Algorithms and related functions (E, D) are public; only the keys (K) are secret

# A Simple Example

- Key/Method for Encryption: this function is a very simple one, transform every character to the next one in the alphabet for a given plaintext (and “z” becomes “a” to circle around the end)
- Input plaintext
  - “where”
- Output ciphertext
  - “xifsf”
- Decryption is the simple method to go back in the alphabet to reverse the effect of encryption

# The Notation

- C = ciphertext, P = plaintext, E = encryption, D =decryption, K = key
- $C = E_K(P)$
- $P = D_K(C)$
- $D_K(E_K(P)) = P$
- In fact what we want in simple crypto-based network security is efficient methods where

  $D_{K1}(E_{K2}(P)) = P$ **<u>if and only if</u>** K1=K2.

# Keys Plays an Important Role

- A key is a **string that allows the selection** of one of many potential encryptions
- The **key can be changed** as often as required
- **Algorithms are more likely to be at the hands of attackers** anyhow, not changed as frequently
- **Cipher is a term** commonly used as the term for algorithm here
- The size of the overall key space is determined by the number of bits in the key string
- The **longer the key, the more effort is required to break a given encryption**

# A common function used in ciphers: Recall XOR

- An XOR is an “exclusive or” function used regularly.
- A XOR B means A or B, but not both
- XOR is commonly used in cryptography

| A | B | A XOR B |
|---|---|---|
| F | F | F |
| F | T | T |
| T | F | T |
| T | T | F |

| Truth values | Binary Equivalents |
|---|---|
| T | 1 |
| F | 0 |

# Some Main Types of Ciphers

- Substitution cipher
  - Each letter of group of letters is replaced systematically by other letters or groups of letters (our example)
- Transposition cipher
  - All letters are re-ordered without disguising them
- One-time pad
  - Uses a random bit string as the key: convert the plaintext into a bit string, then XOR the two strings bit by bit
  - Harder to break
  - How to share the random key and its size are factors

# Example revisited: Substitution cipher example

- Substitution ciphers replace each group of letters in the message with another group of letters based on a key with an intension to disguise the message.

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

- If “were” was the ciphertext received then it becomes

“bcdc”

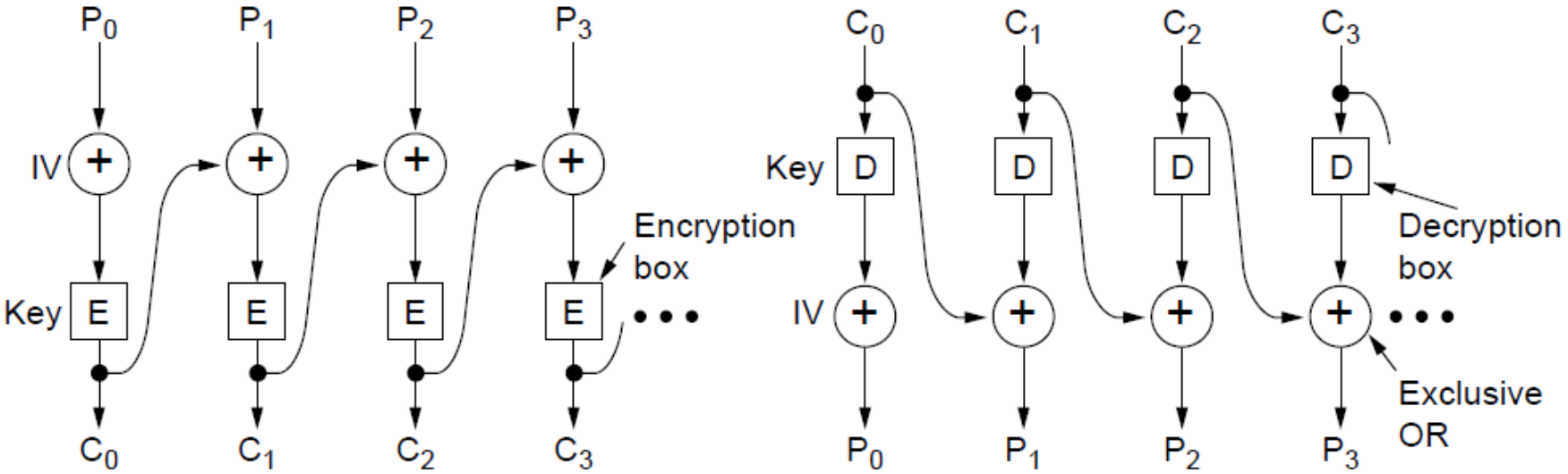
when decrypted

# Modern Key-based Algorithms

- **Two main categories**
- **Symmetric key algorithms** use the same key for both encryption and decryption
- Symmetric key algorithms can use permutation, substitution and a combination of both to encrypt and decrypt
- **2 Symmetric Key Algorithms**
  - Data Encryption Standard (DES)
    - Uses 64 bit blocks and 56 bit keys
    - $2^{56}$ key space
    - Triple DES has a 3 x $2^{56}$ key space
  - Advanced Encryption Standard (AES) in use since 2000s
    - Uses 128 bit blocks and 128 bit keys
    - $2^{128}$ key space
    - Still substitution and permutation based with multiple rounds

# Cipher Block Chaining Mode

- **<u>Same text leads to same ciphertext unless something else is done, thus:</u>**
- In block chaining mode, each plaintext block is XOR'ed with the previous ciphertext block before being encrypted
- (a) encryption, (b) decryption




CBC mode encryption

CBC mode decryption

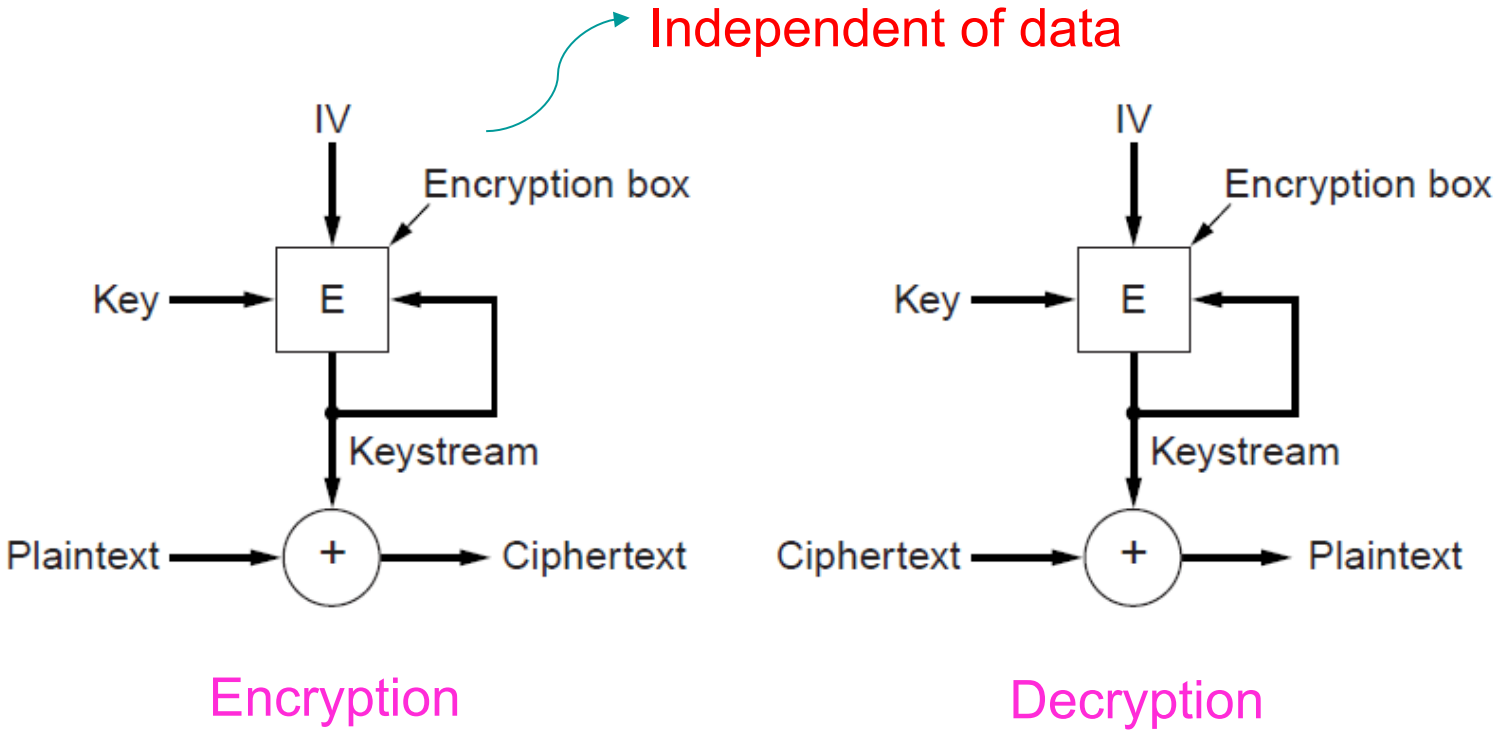
# Cipher Feedback Mode

- In cipher feedback mode, **byte-by-byte encryption** is used rather than block-by-block encryption

- Good for things like encrypting someones key strokes on a keyboard **where a lot of data is not immediately available**

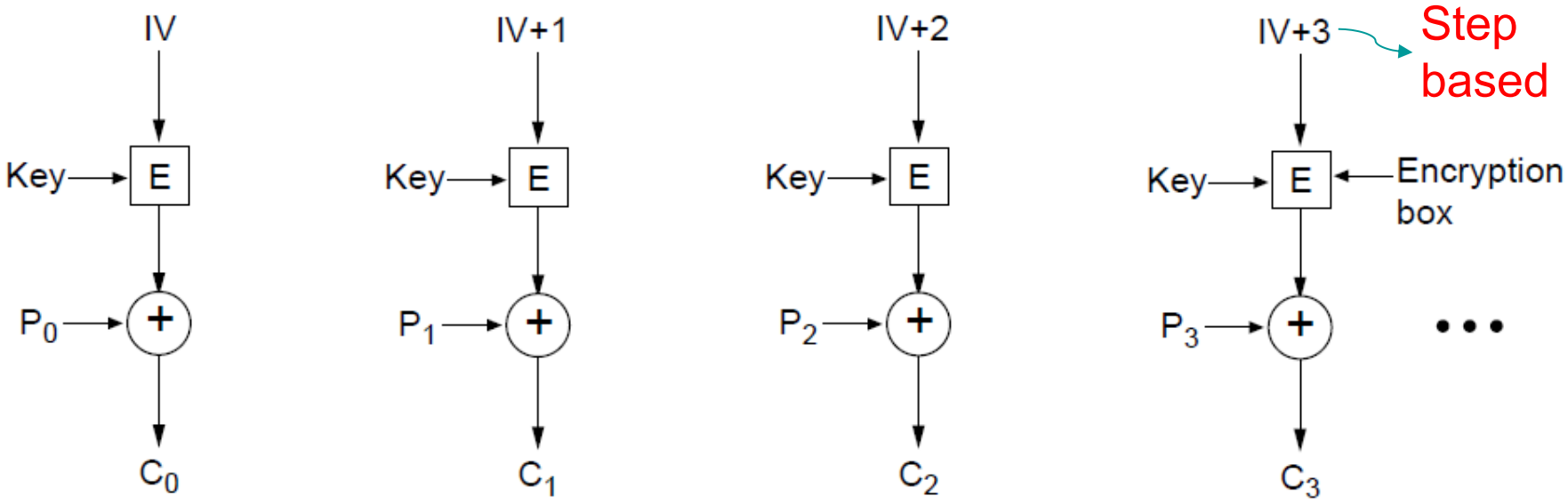
# Stream Cipher Mode

- What if **data transmission errors occur**? In stream cipher mode, recursive sequential block encryption is used as a **one-time pad**, and XOR'ed with plaintext to generate ciphertext



# Counter Mode

- But **how about random access to data after encryption**?
- In counter mode, plaintext is not directly encrypted, but an initialisation parameter plus an arbitrary constant is encrypted, and the resulting ciphertext is XOR'ed with plaintext



# Many Symmetric Key Algorithms Exist

| Cipher | Author | Key length | Comments |
|---|---|---|---|
| Blowfish | Bruce Schneier | 1–448 bits | Old and slow |
| DES | IBM | 56 bits | Too weak to use now |
| IDEA | Massey and Xuejia | 128 bits | Good, but patented |
| RC4 | Ronald Rivest | 1–2048 bits | Caution: some keys are weak |
| RC5 | Ronald Rivest | 128–256 bits | Good, but patented |
| Rijndael | Daemen and Rijmen | 128–256 bits | Best choice |
| Serpent | Anderson, Biham, Knudsen | 128–256 bits | Very strong |
| Triple DES | IBM | 168 bits | Second best choice |
| Twofish | Bruce Schneier | 128–256 bits | Very strong; widely used |