# COMP90007 Internet Technologies Workshop

## Week 11 – Application Layer & Network Security

Ruixi
huor@student.unimelb.edu.au

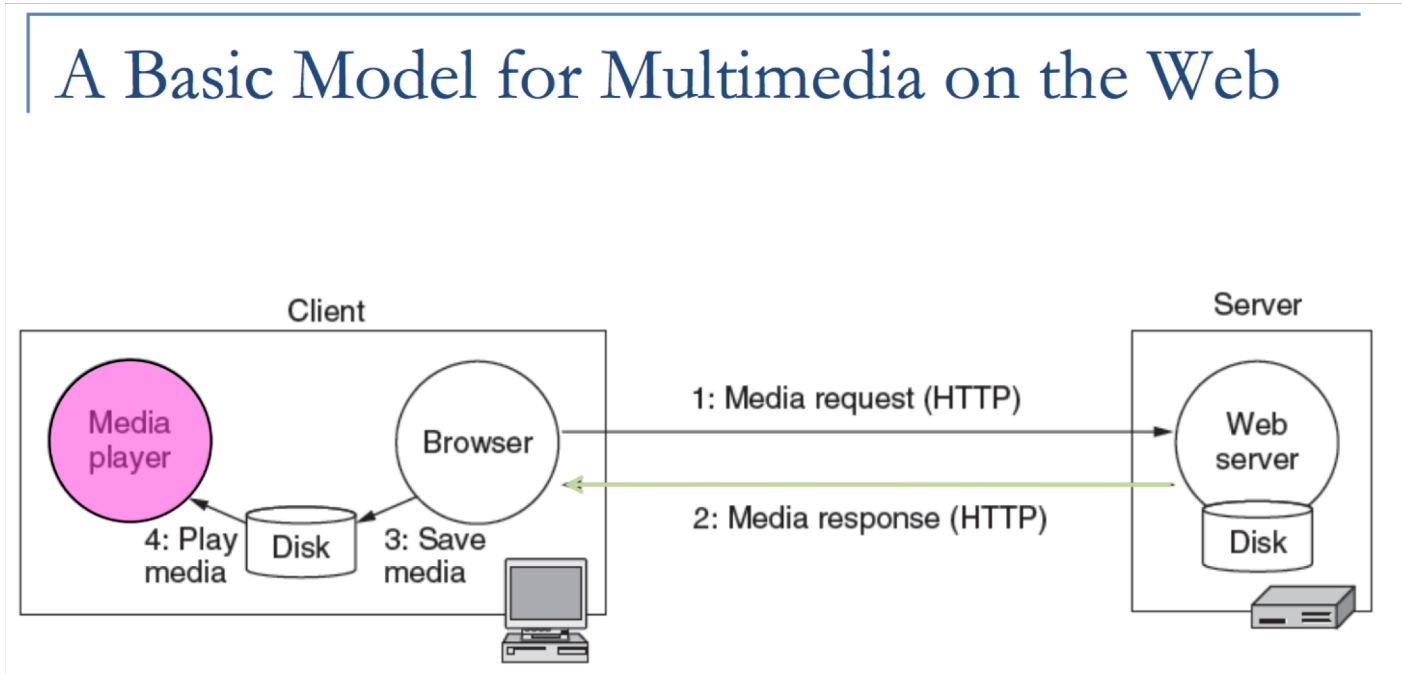
# Multimedia

- Multimedia Networks
  - Audio
  - Video
- Audio and video have become key types of traffic, e.g., voice over IP, and video streaming.

| |
|---|
| Application |
| Transport |
| Network |
| Link |
| Physical |

# Multimedia

## 2 model:

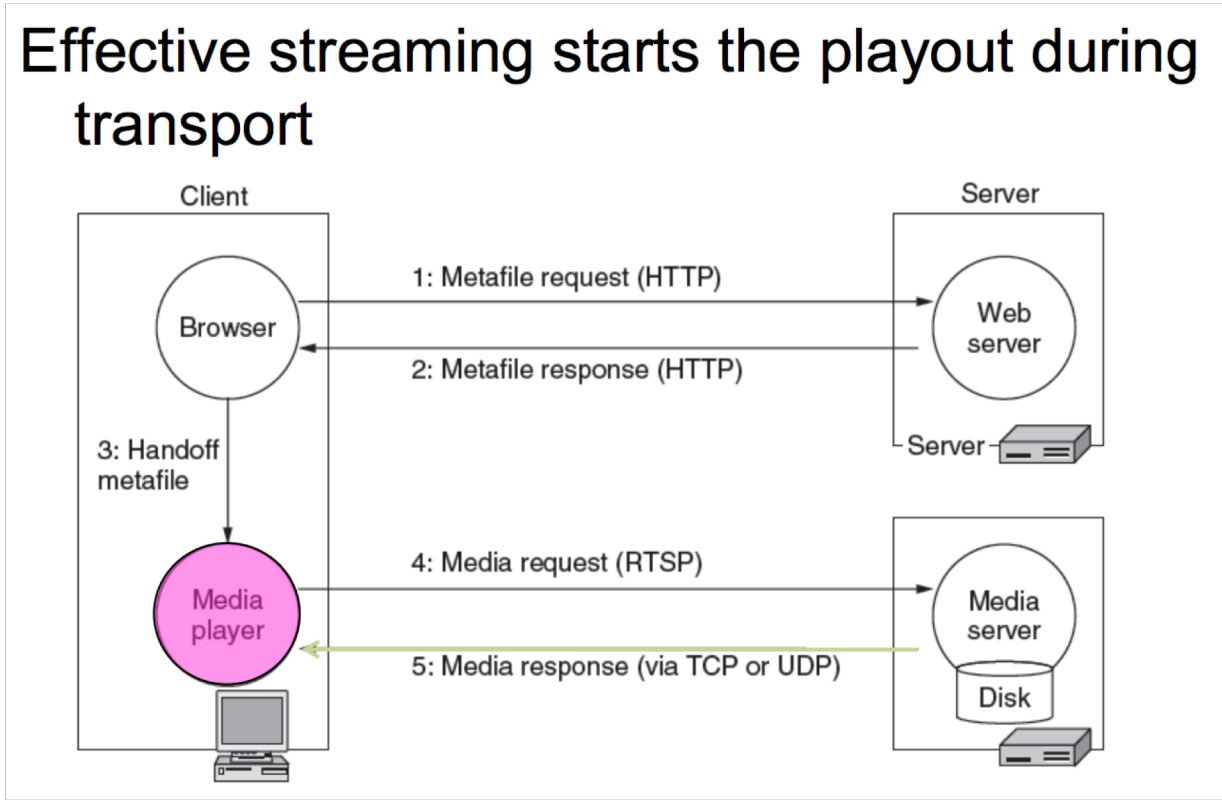



The entire media file must be transmitted over the network before playback starts, causing delay in user experience

# Multimedia

## Streaming Media

**2 model:**



# Opening Question

## What is Bandwidth Delay Product?

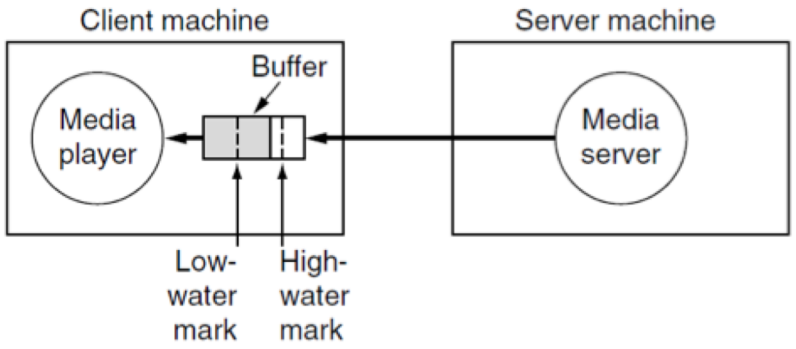
Bandwidth Delay product is an important measure of a link or network path. It is defined as the product of the bandwidth of a link and the delay of the link.

By “bandwidth” we mean the rate of the link or network path.
For “delay”, we will use the propagation delay. With this simplification we have:

$$BD = R \times D$$

# Question 1

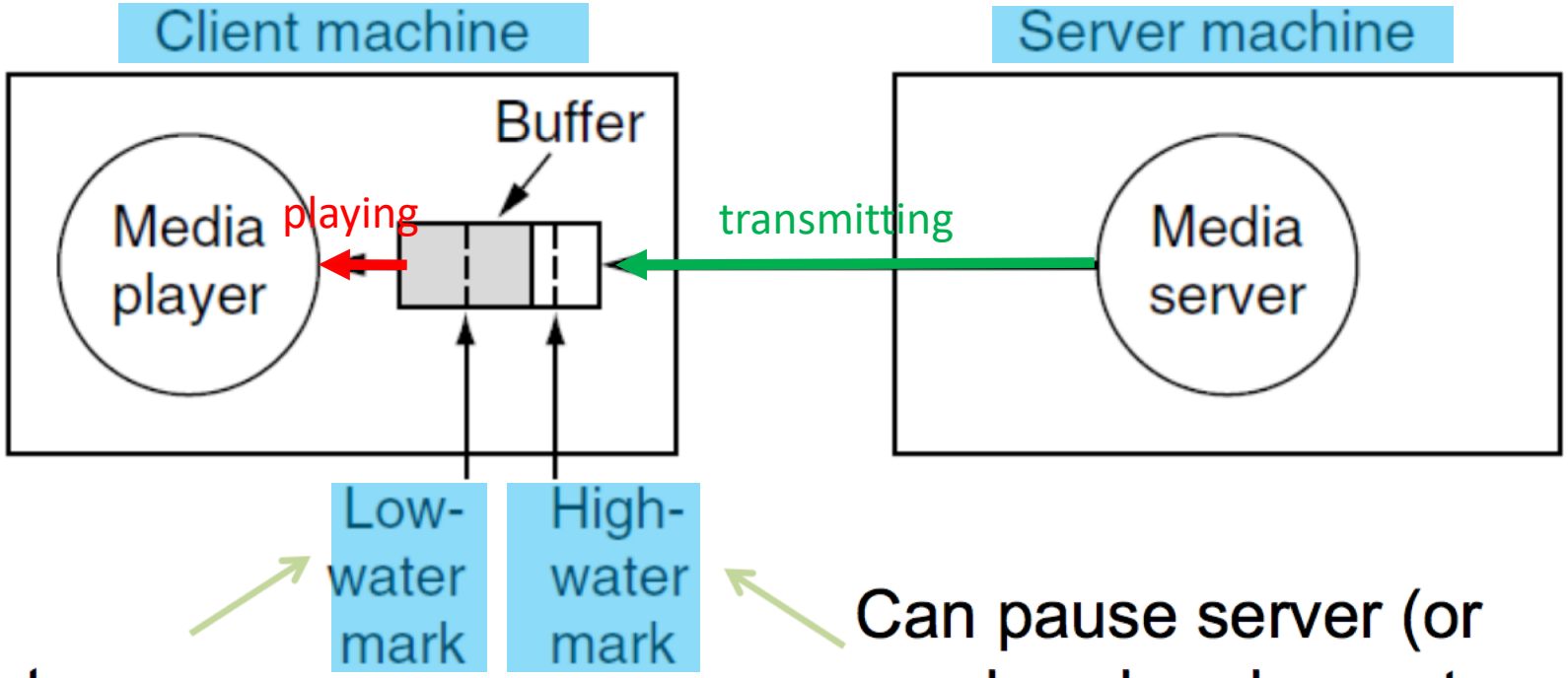
An video streaming server takes 50 ms to communicate with a client media player. If the media player has a 1 MB buffer, what can you say about the position of the low and high-water mark? Assume the media being streamed has a bitrate of 1 Mbps across a sufficiently fast link.




**Why do we need a buffer here?**

# Jitter Management

- Similarly to router buffering, multimedia software buffers streamed media sources prior to transmission
- Buffering is a defensive mechanism to reduce jitter (variance in average packet arrival times)
- Ideally the stream buffer will continue to be filled at the same rate the stream is played back to the user

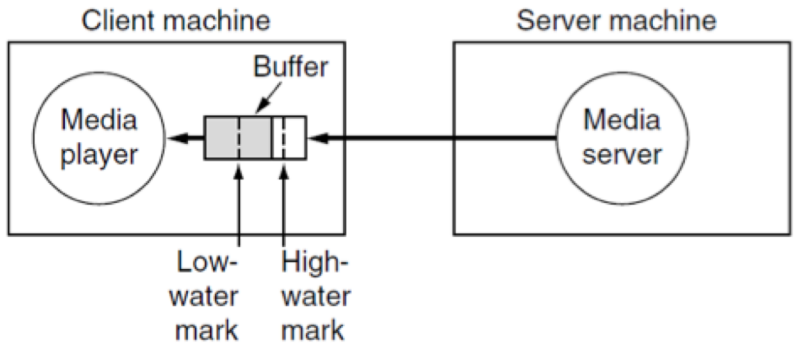



Safety margin, to avoid a stall

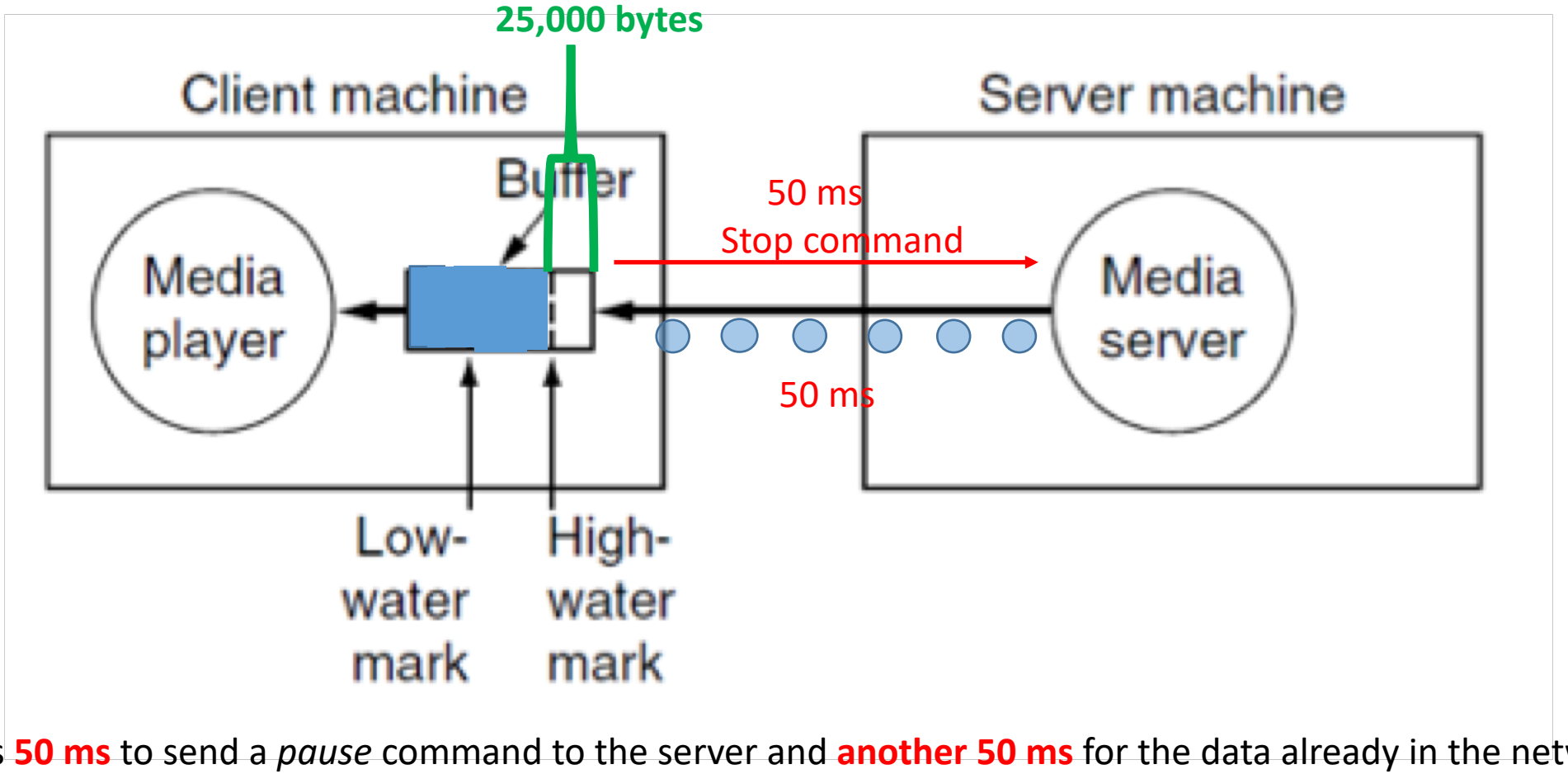
Can pause server (or go ahead and save to disk)

# Question 1

An video streaming server takes 50 ms to communicate with a client media player. If the media player has a 1 MB buffer, what can you say about the position of the low and high-water mark? Assume the media being streamed has a bitrate of 1 Mbps across a sufficiently fast link.



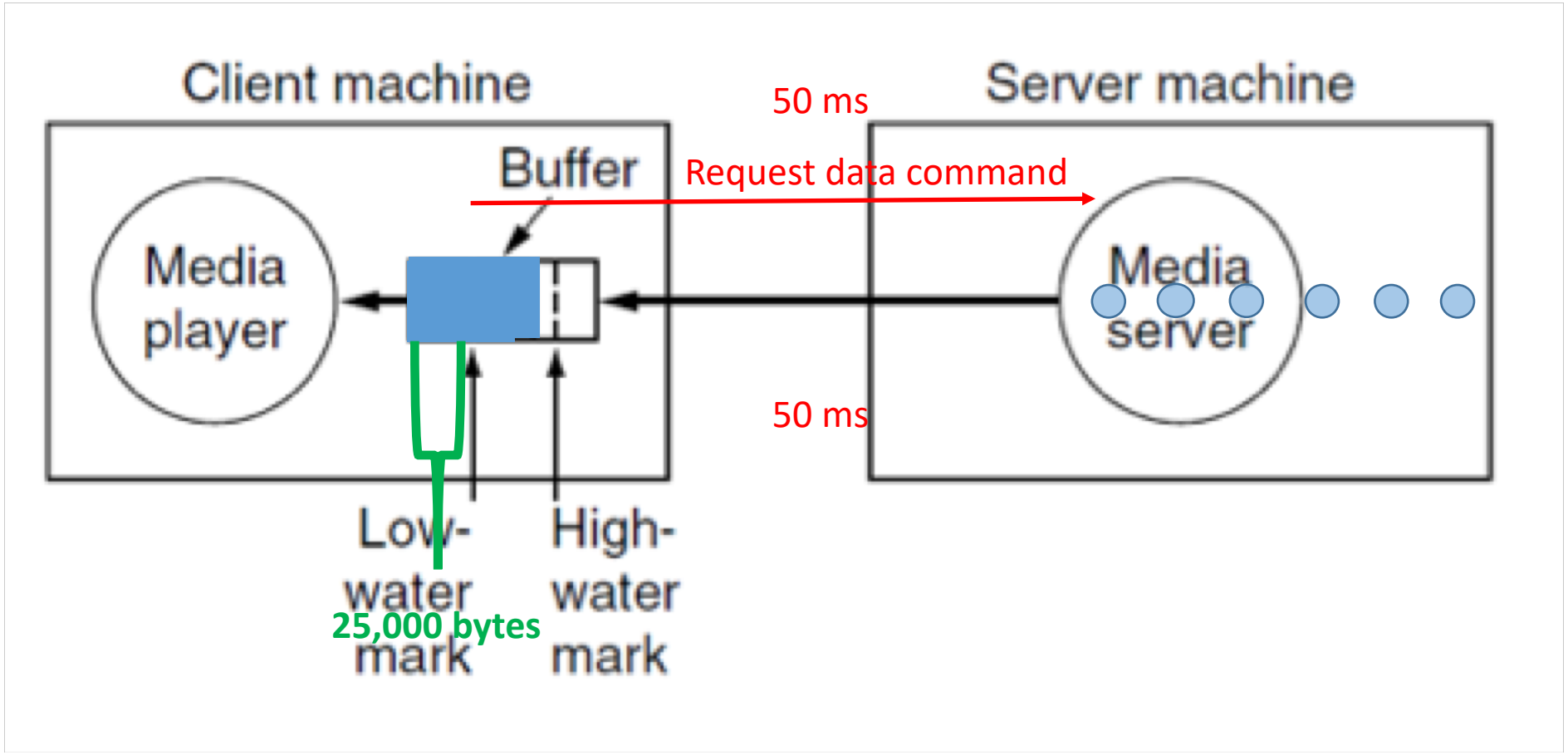
# How to identify the position of high-water mark?




It takes **50 ms** to send a *pause* command to the server and **another 50 ms** for the data already in the network to drain after the server stops sending.
100 ms × 1 Mbps = 12,500 bytes will arrive, so the high-water mark should be at least **12,500 bytes below the top** to avoid buffer overflow.

To be absolutely safe, set the mark to be 25,000 bytes from the top.

# How to identify the position of low-water mark?

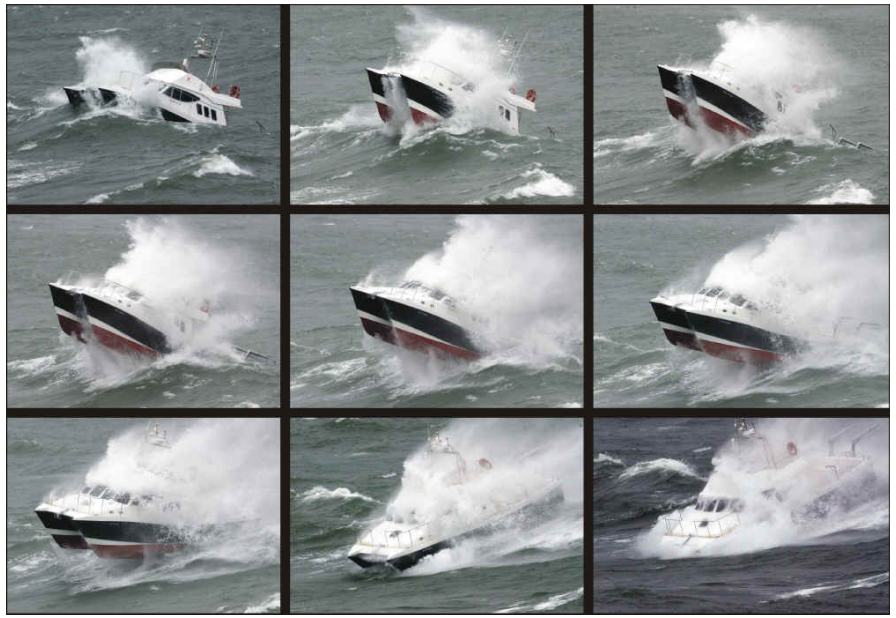



Similarly, it takes **100 ms** for the client to request data from the server and start receiving it. 12,500 bytes will be played during this time.

Hence, the low-water mark should be at least 12,500 bytes and probably 25,000 bytes to be safe.

# Question 2

What is the bitrate required to transmit 1280 × 720 pixel colour frames with 8 bits/pixel at 30 frames/sec? How can such a video be streamed across the internet?



*Answer:*

- 30 frames/sec × 1280×720 pixels/sec
  = 27.648 megapixels/sec
- $27.648 \times 10^6 \times 8$ bits/sec
  = 221.2 Mbps (!!)

Assuming 70% compression ratio of each image:
= 66.36 Mbps (!)

Solution: use video compression techniques –
encode the **differences** from each frame to the next

# Question 3

A compact disk holds about 650 MB of data. Is compression used in the production of audio CDs?

Assume that there is 60 minutes of audio to be stored and that it is sampled at 44.1 kHz using 16 bits each for the left and right stereo channels.

*Answer:*

- Data rate of audio: $44.1 \times 10^3 \times 16 \times 2$
  $= 1.411$ Mbps
  $= 176$ kB/s

- Size required for 60 min of audio: 176 kB/s × 60×60 sec = 634 MB
  ∴ Compression is not used.

# Cryptography Concepts

- Three foundational concepts
  - 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

# Types of Ciphers

| | |
|---|---|
| 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 |

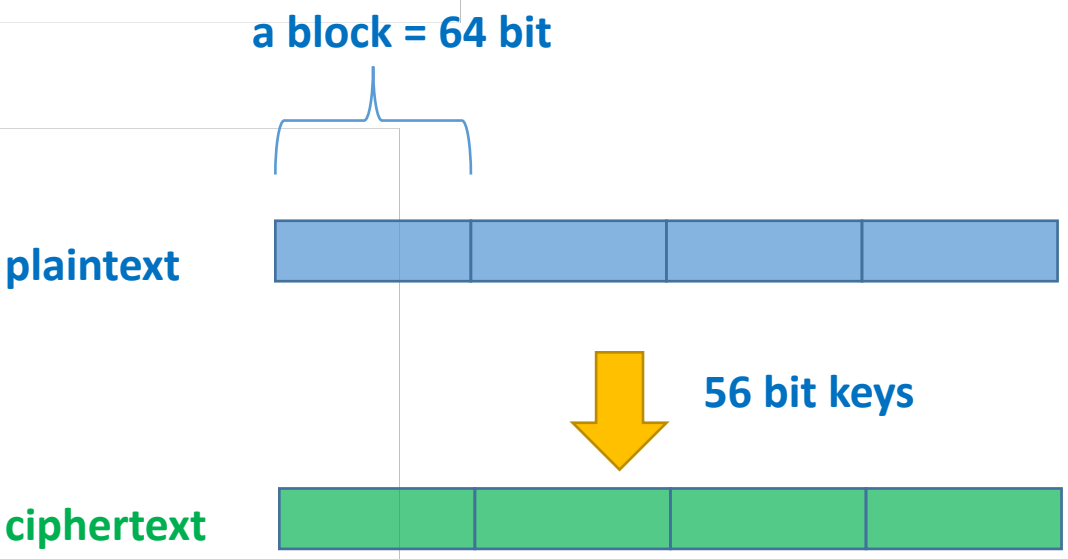
- Substitution cipher
  - Each letter of group of letters is replaced systematically by other letters or groups of letters (breakable with knowledge of the replacement system)
- Transposition cipher
  - All letters are re-ordered without disguising them (breakable with knowledge of re-ordering system)
- 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 (unbreakable because any other plaintext has the same likelihood as the original message)

# Symmetric Key Algorithms

- A symmetric key algorithm uses the same key for both encryption and decryption

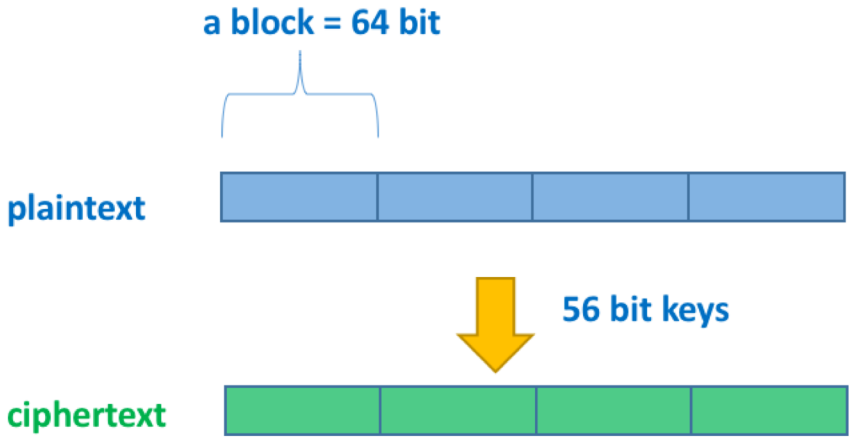
- Symmetric key algorithms can use permutation, substitution and a combination of both to encrypt and decrypt

- 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)
    - Uses 128 bit blocks and 128 bit keys (others available)
    - $2^{128}$ key space



# Question 4

Design an attack on the DES cipher by using the knowledge that the plaintext to be sent contains only upper case ASCII letters and the space character.




*Answer:*

Take the first block, and **try** keys **until** the plaintext only contains legal characters.

Keep trying subsequent blocks while the key produces only legal characters. If the key produces an illegal character, go back to the first block and keep trying more keys.

# Cipher Modes

Cipher Block Chaining Mode

Cipher Feedback Mode

Stream Cipher Mode

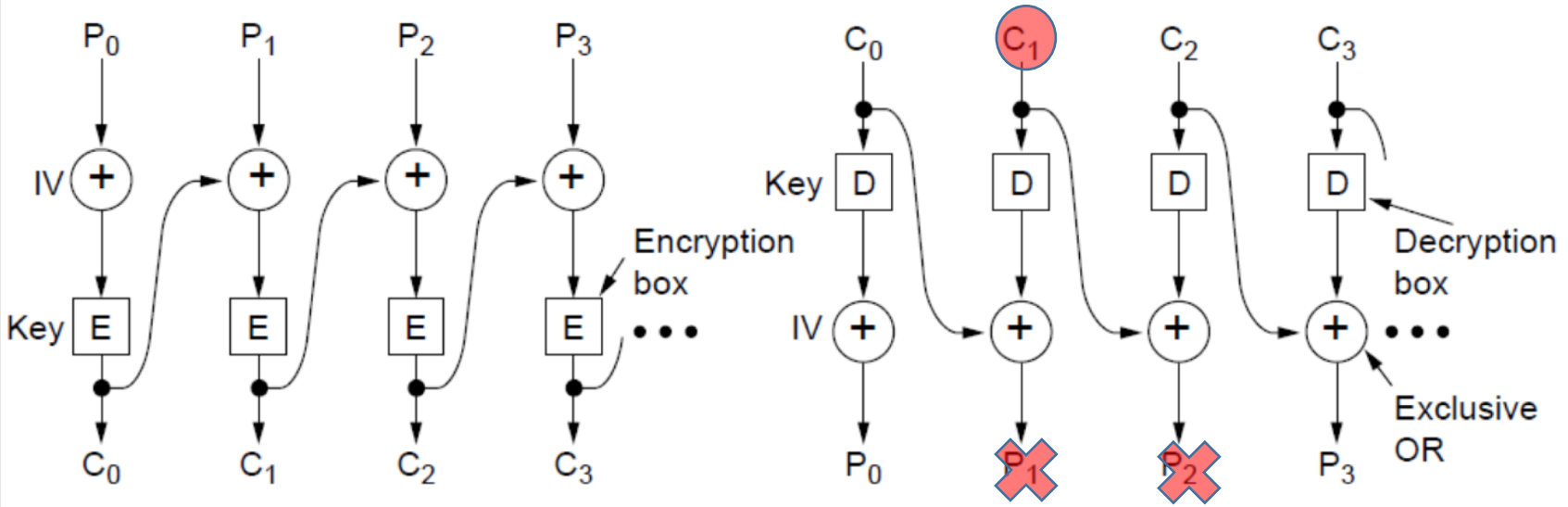
Counter Mode

# Cipher Modes: Block Chaining

- An XOR is an “exclusive or” function.
- A XOR B means A or B, but not both
- XOR is commonly used in cryptography as a comparison mechanism in multiphase encryption and decryption

- In block chaining mode, each plaintext block is XOR'ed with the previous ciphertext block before being encrypted
- (a) encryption, (b) decryption

IV = initialization vector




CBC mode encryption

CBC mode decryption

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

# Question 5

Suppose that a message has been encrypted using DES in ciphertext **block chaining mode**. One bit of the ciphertext in block $C_i$ has been corrupted during transmission. How much of the plaintext will be corrupted as a result?

*Answer:*

If a single bit error occurs in $C_i$ , then all of $P_i$ is likely to be corrupted. $C_i$ is then XOR'ed with D($C_{i+1}$), which will cause a single bit error in $P_{i+1}$.

No other plaintext will then be corrupted. **A single bit error thus only affects two plaintext blocks.**

# Question 6

Now consider ciphertext **block chaining** again. Instead of a single 0 bit being transformed into a 1 bit, an extra 0 bit is inserted into the ciphertext stream after block $C_i$ . How much plaintext will be garbled as a result?

*Answer:*
Unfortunately, **every plaintext block starting at $P_{i+1}$ will be wrong now**, since all the inputs to the XOR boxes will be wrong. A framing error is thus much more serious than an inverted bit.

# Cipher Modes

Cipher Block Chaining Mode

Cipher Feedback Mode

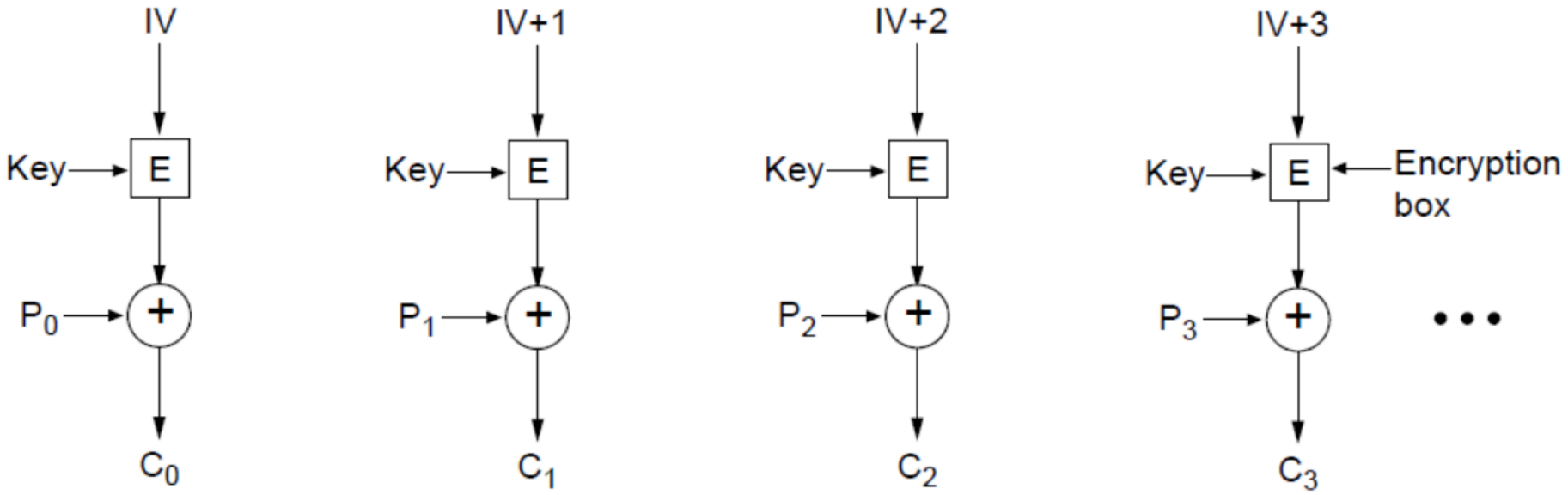
Stream Cipher Mode

Counter Mode

# Cipher Modes: Counter Mode

- 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 to generate new ciphertext

IV = initialization vector




Encryption above; repeat the operation to decrypt

# Question 7

Consider the use of **counter mode**, as shown in Fig. 8-15, but with IV = 0. Does the use of 0 threaten the security of the cipher in general?

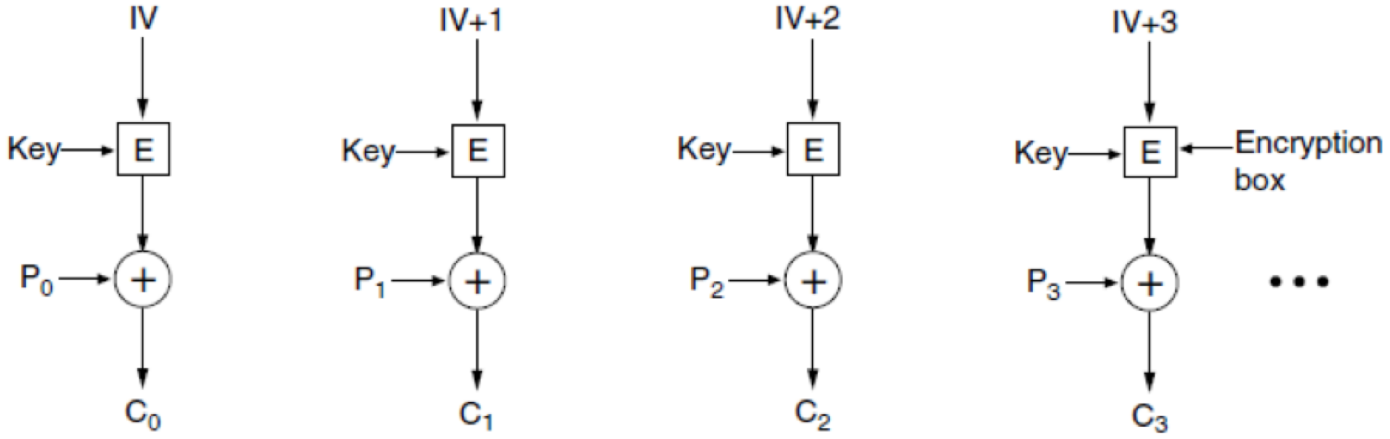



**Figure 8-15.** Encryption using counter mode.

*Answer:*

No. The security is based on having a strong crypto algorithm and a long key. **The IV is not really essential.** The key is what matters.

# Question 8

What are some of the differences between HTTP 1.0 and HTTP 1.1?

| HTTP 1.0 | HTTP 1.1 |
|---|---|
| - Does not officially require a Host Header to be included. | - Requires a Host Header as per its specification |
| - Open a new connection for each request/ response pair, following which the connection would be closed. (leads to efficiency problems – TCP slow start). | - Allows persistent connections, allowing more than one request/ response on the same HTTP connection. |