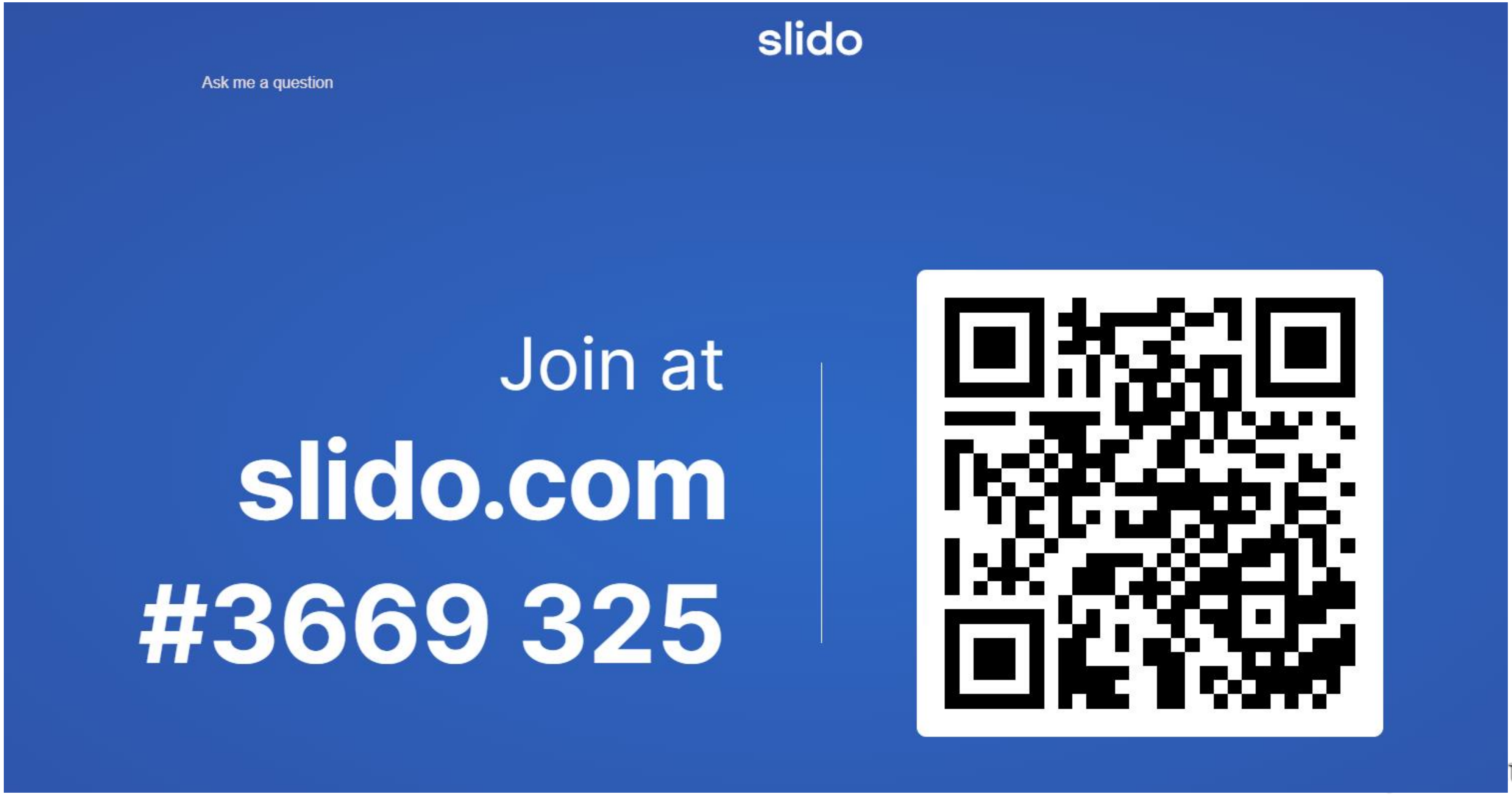
Lesson 7: Ethernet/LAN

Van-Linh Nguyen

Fall 2024

國立中正大學 Cybersecurity Lab

Ask me a question without revealing your name



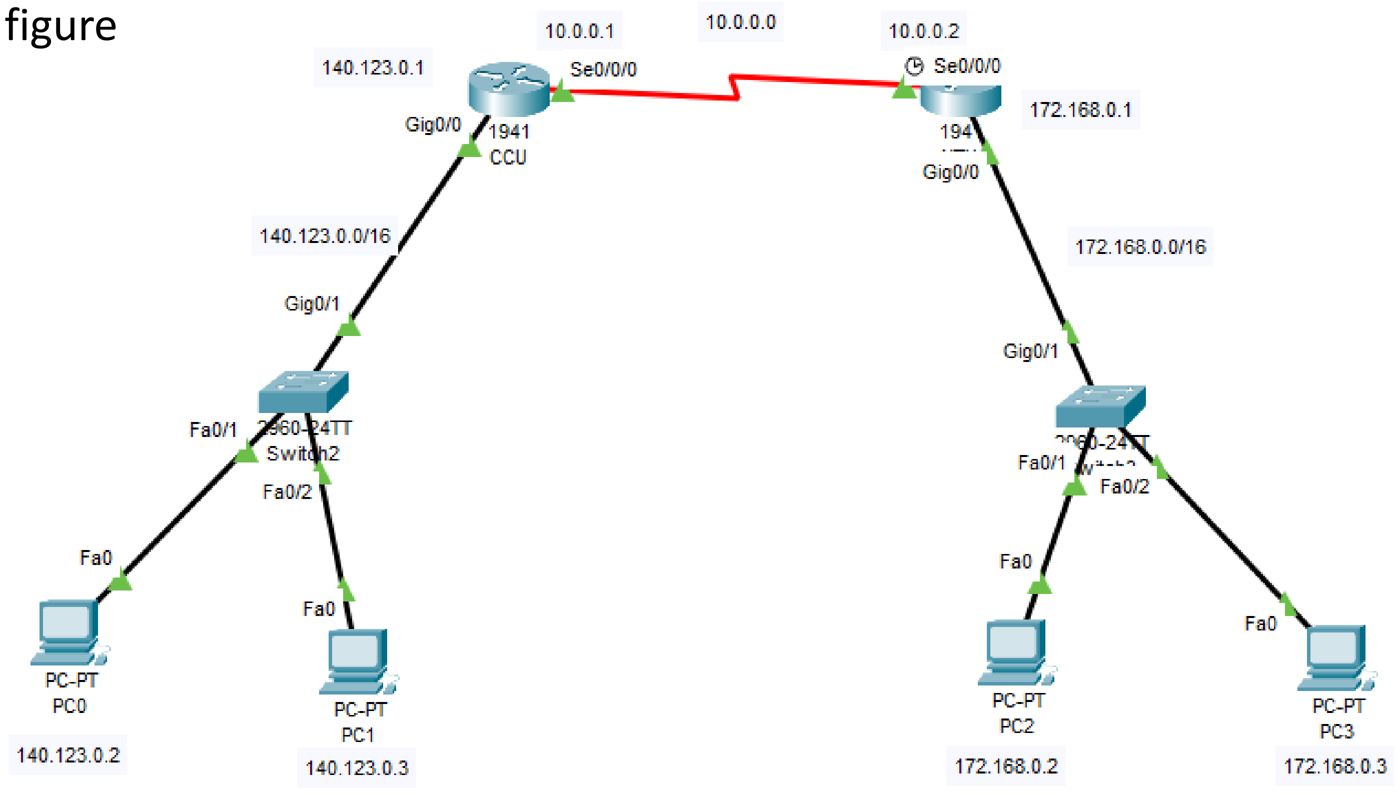
Outline

- Ethernet/LAN
  1. Encoding
  2. Framing
  3. Error Detection
  4. Reliable Transmission
  5. Ethernets/LAN

Configure OSPF [small]

QR Code For Q&A

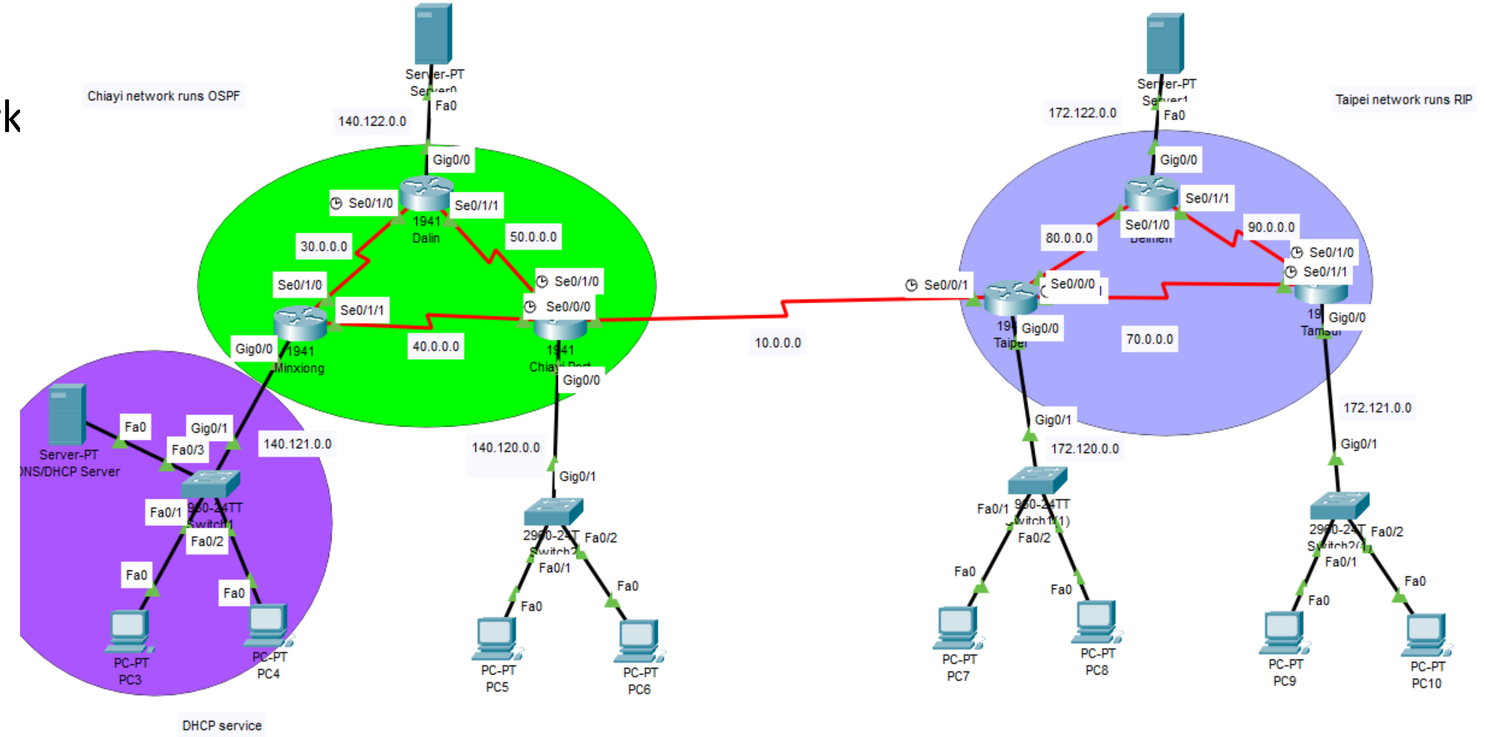
Configure routing for a network with topology as the figure



Cybersecurity Lab

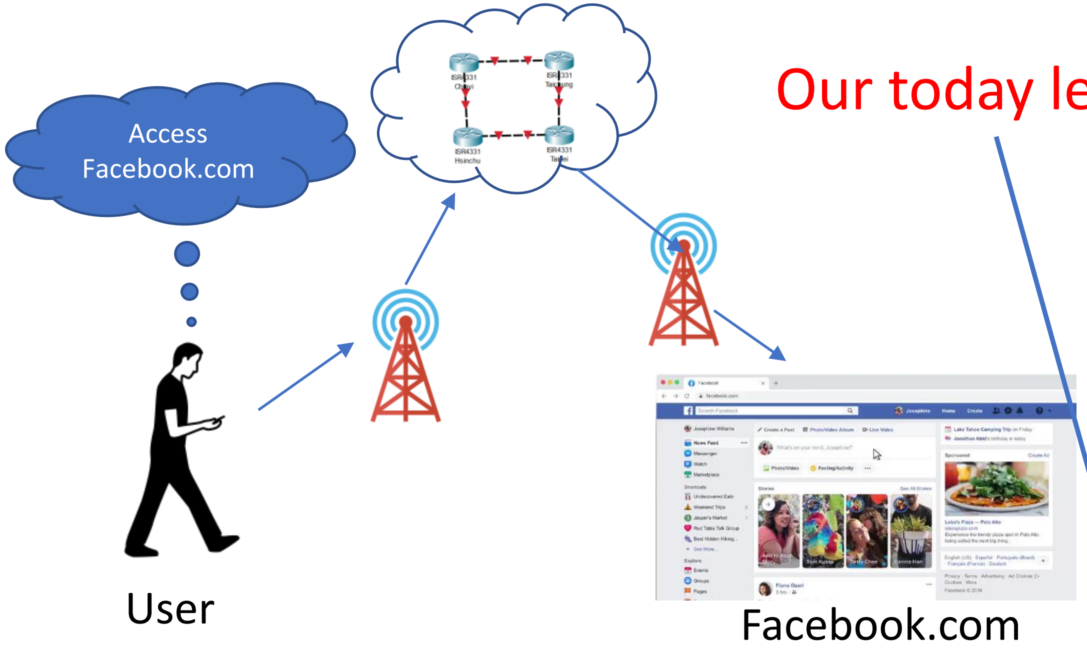
Configure OSPF –RIP – BGP [big]

1. Configure routing for a network with topology as the figure
2. Configure DHCP for the local network

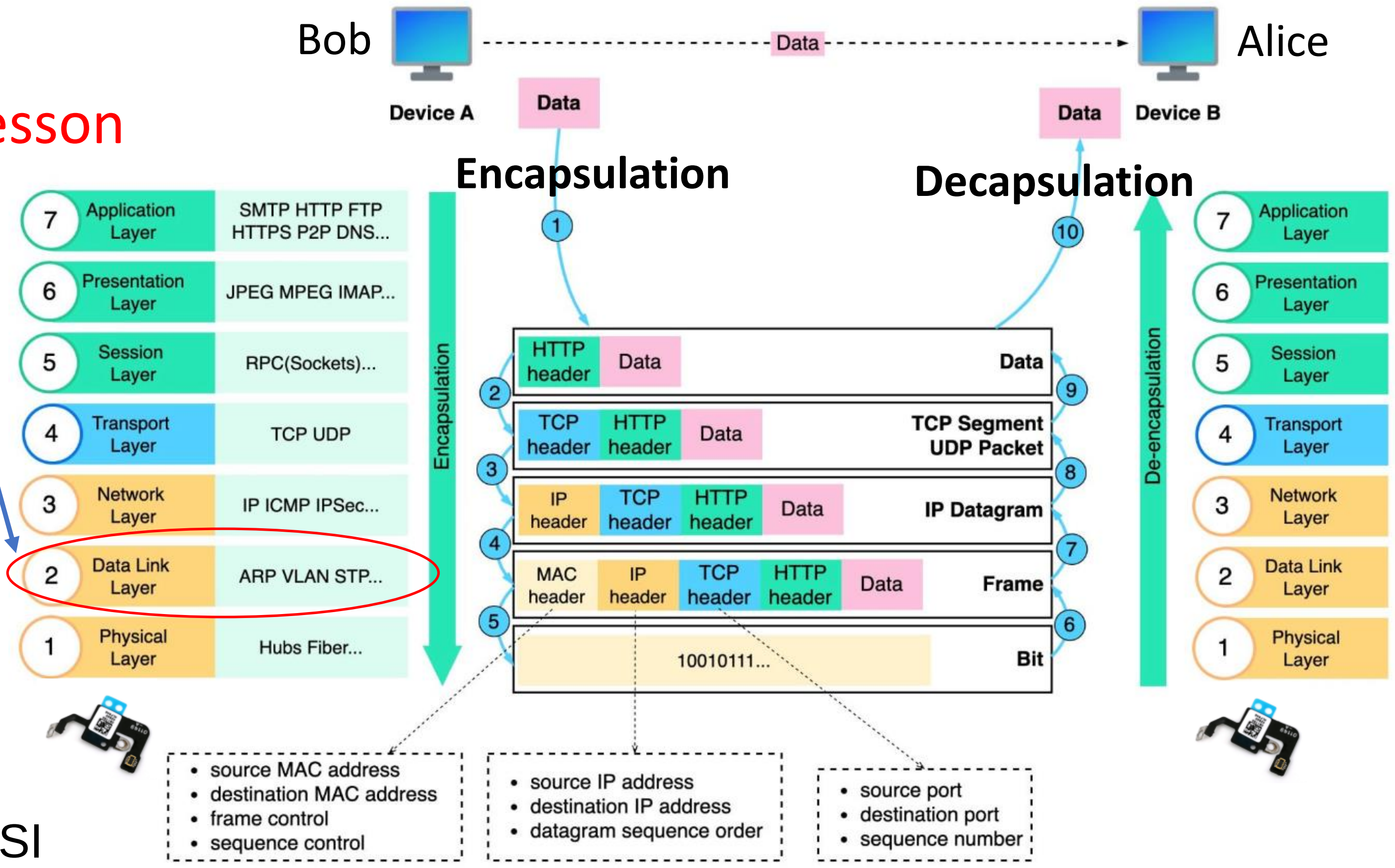


Today lesson

Encapsulation/Decapsulation



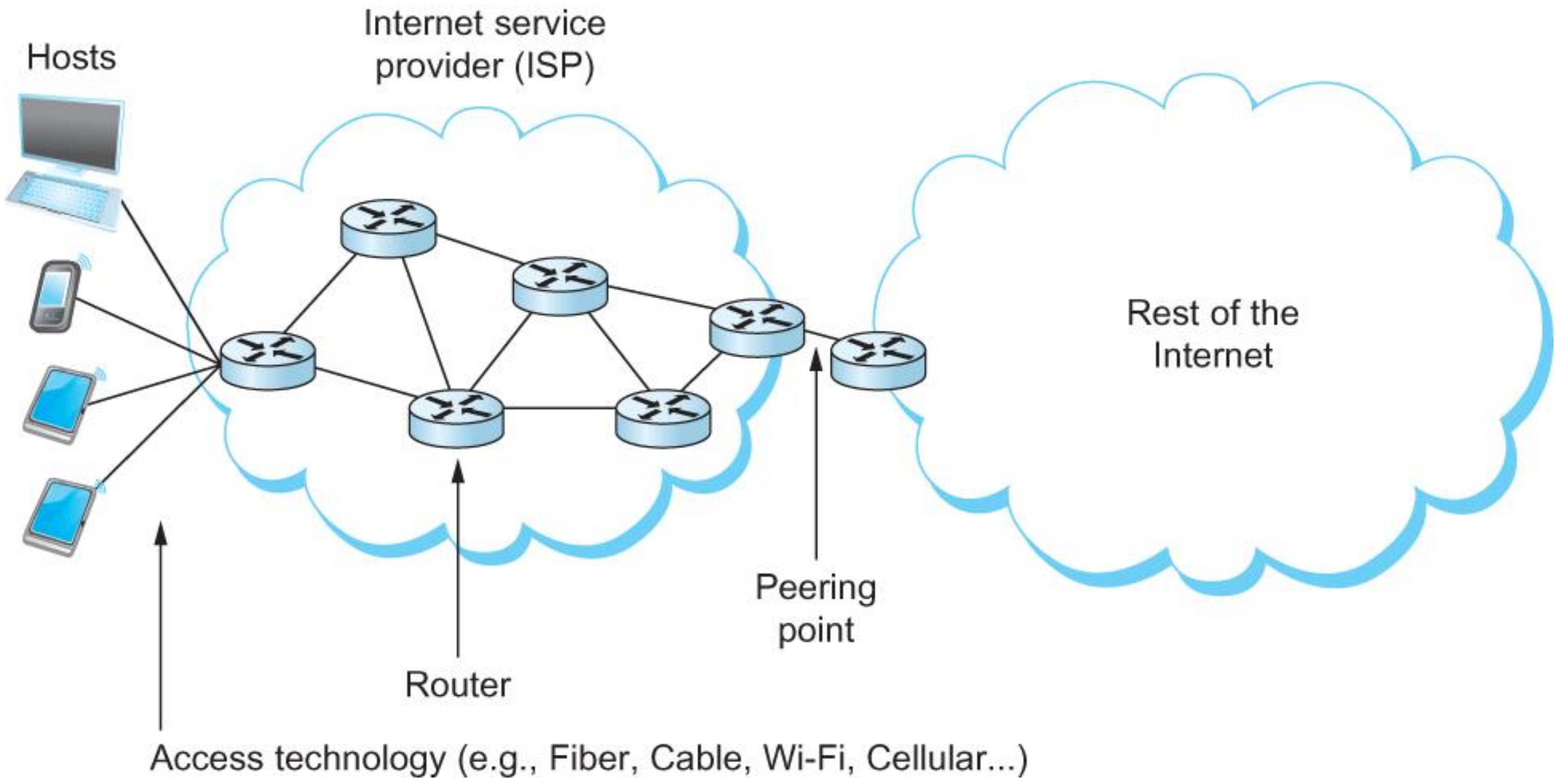
Our today lesson



https://blog.bytebytego.com/

TCP/IP Model is more popular in industry than OSI
OSI is for academic research

Perspectives on Connecting



An end-user's view of the Internet

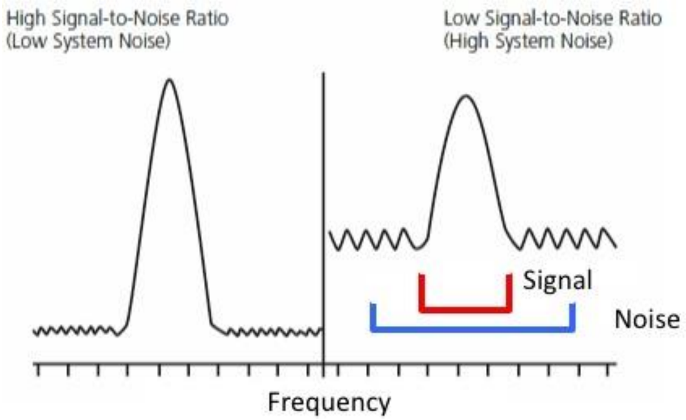
Shannon formula – Link capacity

QR Code For Q&A

- How much bit per second we get from every Hz?

$$C = W * n * Log_2\left(1 + \frac{S}{N}\right)$$

- C: Channel Capacity or Data rate (bits/sec)
- W: Channel bandwidth (Hz)
- n: Number of antennas
- S: Signal power
- $\frac{S}{N}$: Signal to noise ratio (SNR) in dB



* Signal power S is decreased by path loss = $\left(\frac{4\,\pi\,d\,c}{f}\right)^2$

- d: Distance
- c: Speed of light
- f: Frequency

國立中正大學 Cybersecurity Lab

QR Code For Q&A

Other Shannon variant

- How much bit per second we get from every Hz?

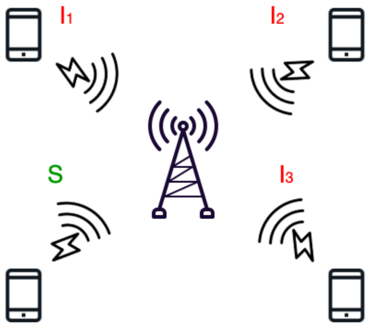
$$C = W * n * Log_2 \left(1 + \frac{S}{I + N}\right)$$

Channel Capacity or Data rate (bits/sec)

Channel bandwidth (Hz)

Number Of antennas

Signal-to-interference-plus-noise (SINR)



* Interference is influenced by surrounding signal sources = $\sum_i^N \mathrm{I}_i$

Cybersecurity Lab

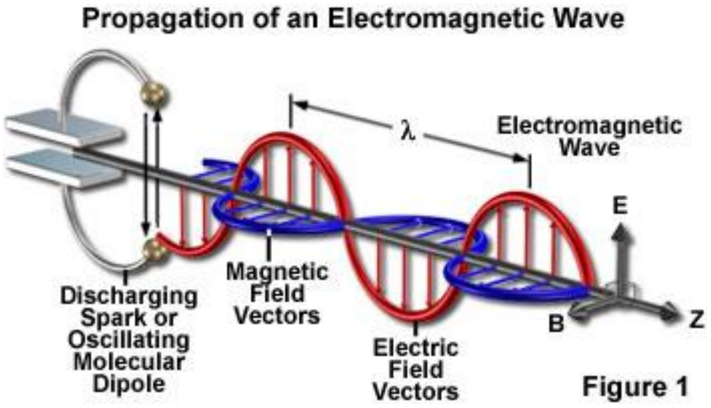
Link Capacity

- $C = W * n * Log_2(1 + \frac{S}{N})$
- W = 3300 – 300 = 3000Hz, S is the signal power, N the average noise.
- The signal to noise ratio (S/N) is measured in decibels is related to dB = 10 x $\log_{10}$(S/N).
- If there is 30dB of noise then S/N = 1000.
- Now C = 3000 x $\log_2$(1001) = 30kbps.
- How can we get 56kbps?

Links

QR Code For Q&A

- All practical links rely on some sort of electromagnetic radiation propagating through a medium or, in some cases, through free space



- One way to characterize links, then, is by the medium they use
  - Typically copper wire in some form (as in Digital Subscriber Line (DSL) and coaxial cable),
  - Optical fiber (as in both commercial fiber-to-the home services and many long-distance links in the Internet's backbone)
  - Air/free space (for wireless links)

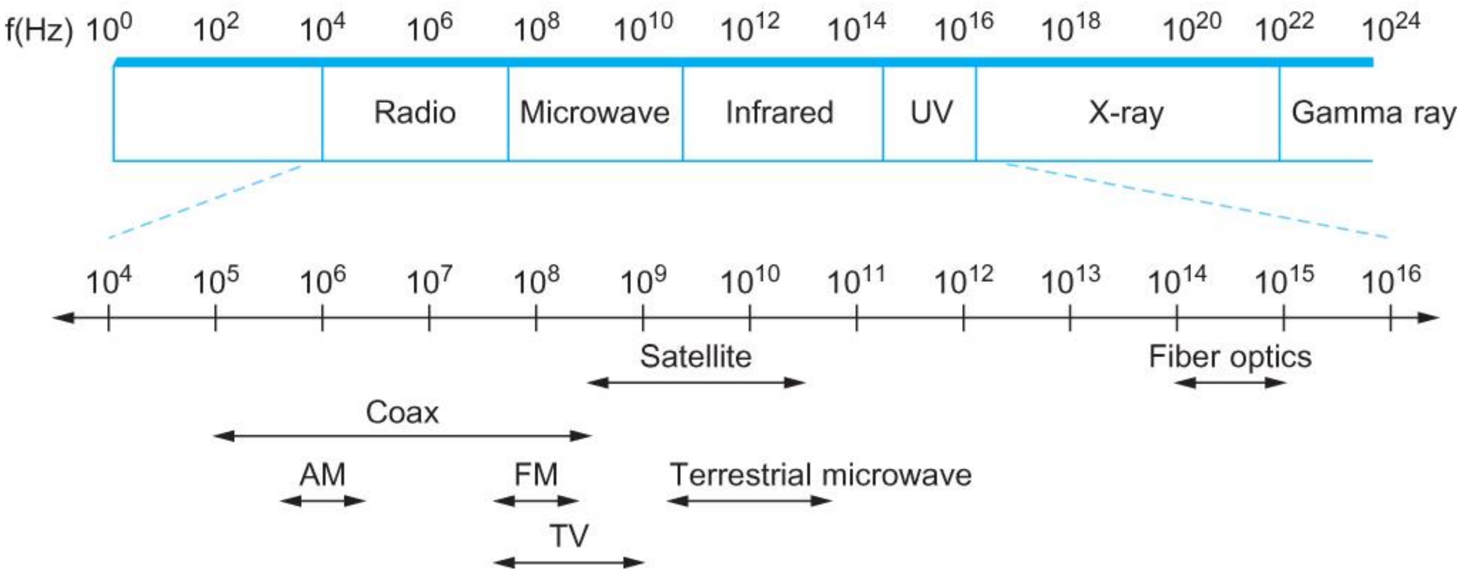
國立中正大學 Cybersecurity Lab

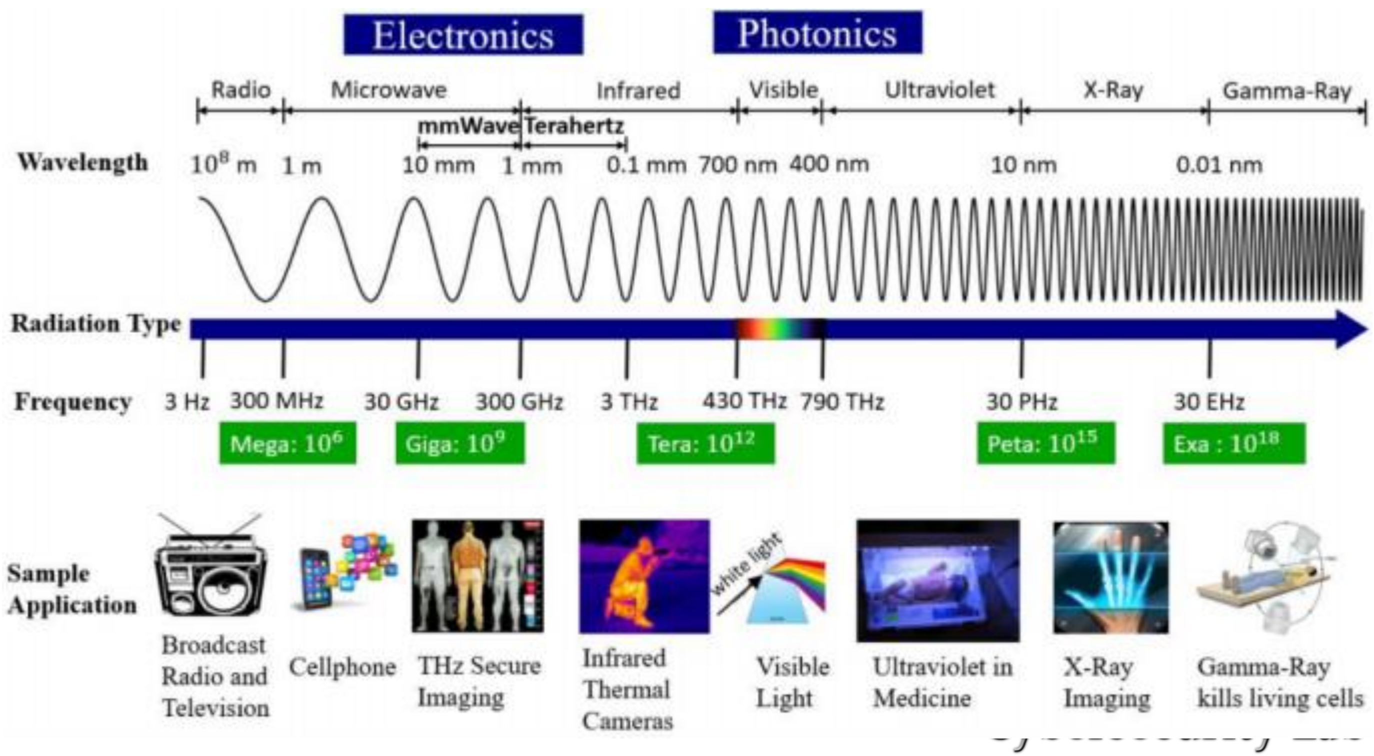
Links

- Another important link characteristic is the *frequency*
  - Measured in hertz, with which the electromagnetic waves oscillate
- Distance between the adjacent pair of maxima or minima of a wave measured in meters is called *wavelength*
  - Speed of light divided by frequency gives the wavelength.
  - Frequency on a copper cable range from 300Hz to 3300Hz; Wavelength for 300Hz wave through copper is speed of light on a copper / frequency
  - $2/3 \times 3 \times 10^8 / 300 = 667 \times 10^3$ meters.
- Placing binary data on a signal is called *encoding*.
- Modulation involves modifying the signals in terms of their frequency, amplitude, and phase.

Links

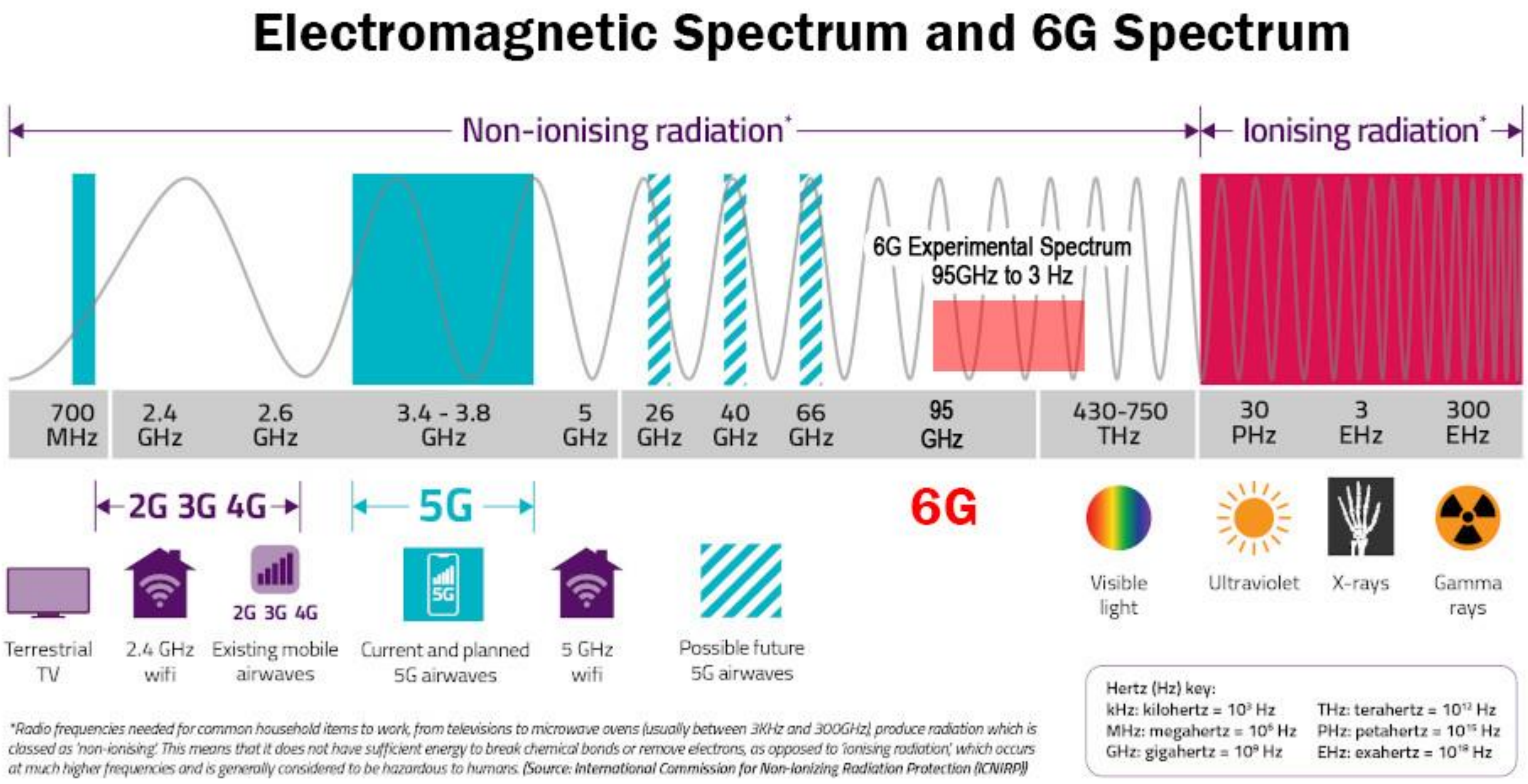
Wireless communications





QR Code For Q&A

Links



https://precisionmmw.com/what-is-6g/

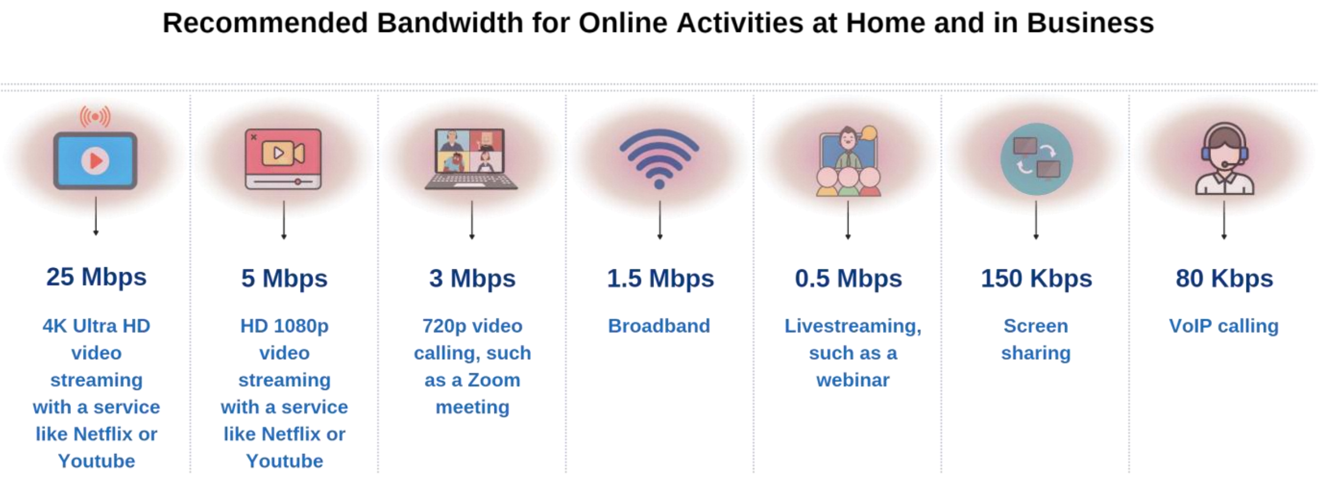
國立中正大學 Cybersecurity Lab

QR Code For Q&A

Links

- Some common services and typical bandwidth

Service	Bandwidth (typical)
Dial-up	28–56 kbps
ISDN	64–128 kbps
DSL	128 kbps–100 Mbps
CATV (cable TV)	1–40 Mbps
FTTH (fibre to the home)	50 Mbps–1 Gbps



#broadbandsearch

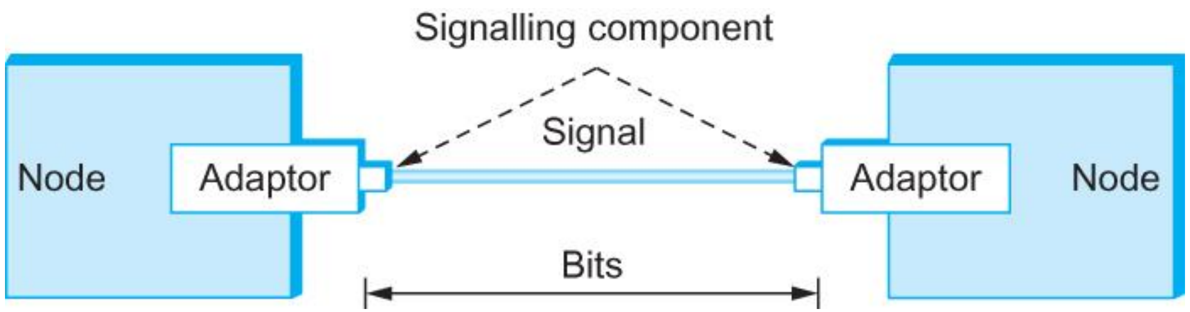
QR Code For Q&A

Encoding

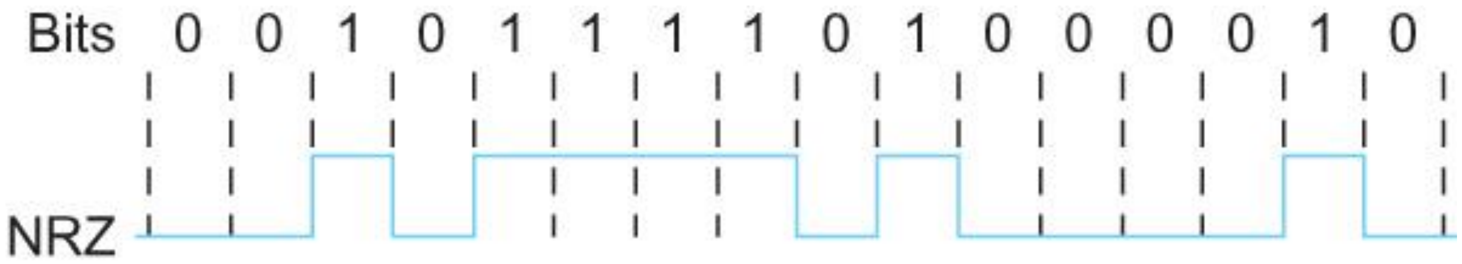
Binary code is represented:

1: a positive voltage

0: the rest conditions



Signals travel between signaling components; bits flow between adaptors

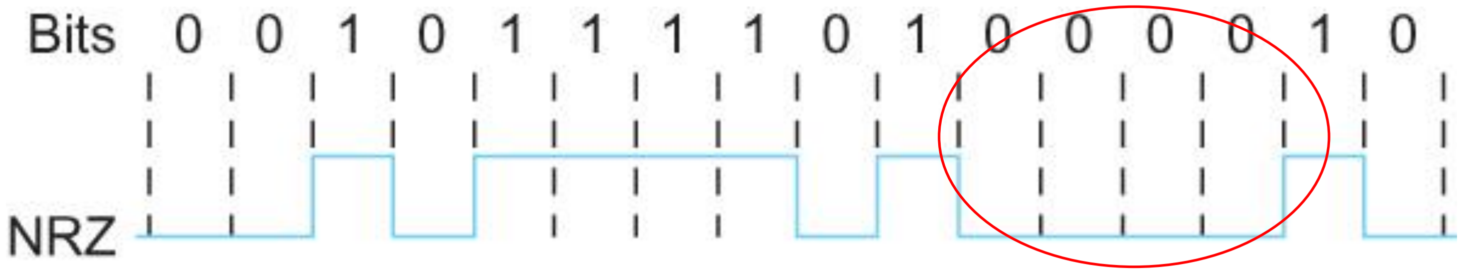


NRZ encoding of a bit stream

QR Code For Q&A

Encoding

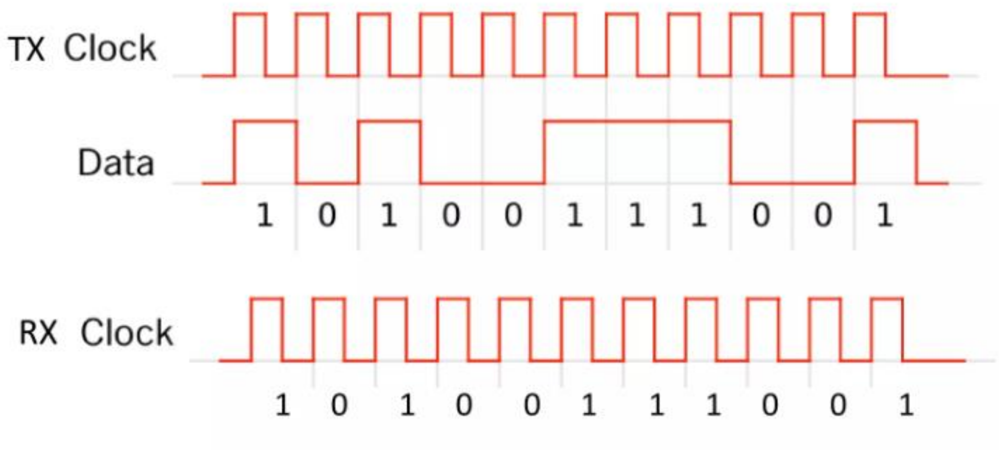
- Problem with Non-return-to-zero (NRZ)
  - Baseline wander
    - The receiver keeps an average of the signals it has seen so far
    - Uses the average to distinguish between low and high signal
    - When a signal is significantly low than the average, it is 0, else it is 1
    - Too many consecutive 0’s and 1’s cause this average to change, making it difficult to detect



QR Code For Q&A

Encoding

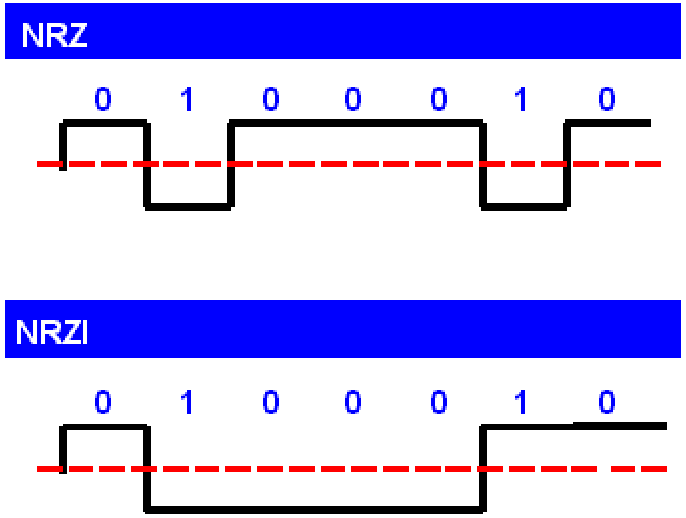
- Problem with NRZ
  - Clock recovery
    - Frequent transition from high to low or vice versa are necessary to enable clock recovery
    - Both the sending and decoding process is driven by a clock
    - Every clock cycle, the sender transmits a bit and the receiver recovers a bit
    - The sender and receiver have to be precisely synchronized



QR Code For Q&A

Encoding

- NRZI
  - Non Return to Zero Inverted
  - Sender makes a transition from the current signal to encode 1 and stay at the current signal to encode 0
  - Solves for consecutive 1's

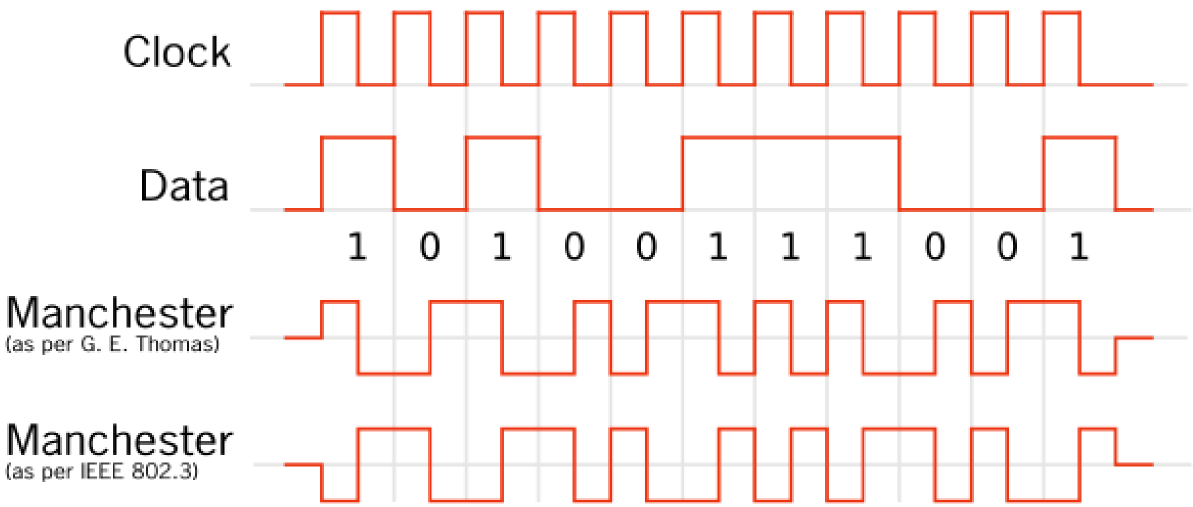


國立中正大學 Cybersecurity Lab

QR Code For Q&A

Encoding

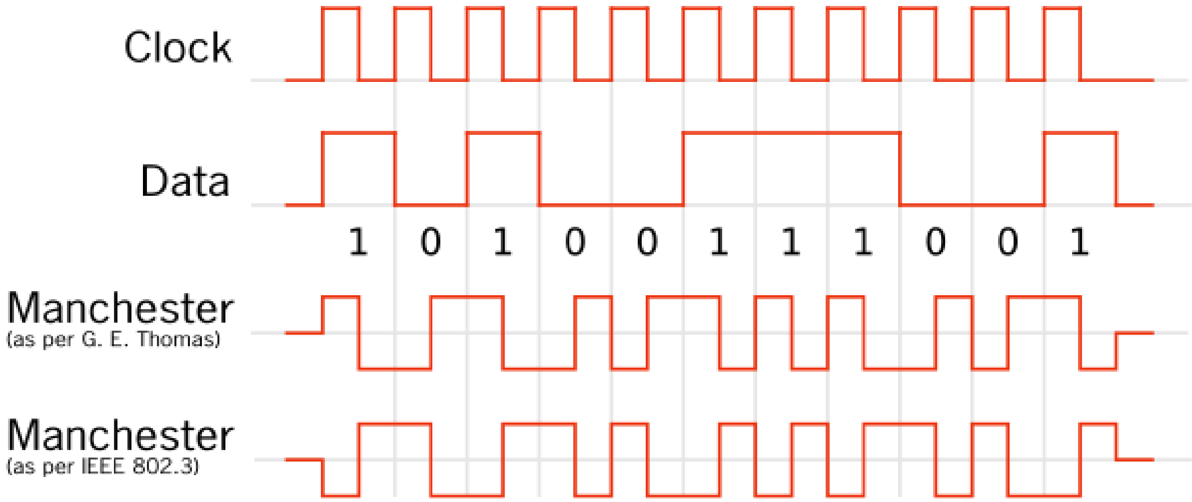
- Manchester encoding
  - Merging the clock with signal by transmitting Ex-OR of the NRZ encoded data and the clock
  - Clock is an internal signal that alternates from low to high, a low/high pair is considered as one clock cycle
  - In Manchester encoding
    - 0: low→ high transition
    - 1: high→ low transition



QR Code For Q&A

Encoding

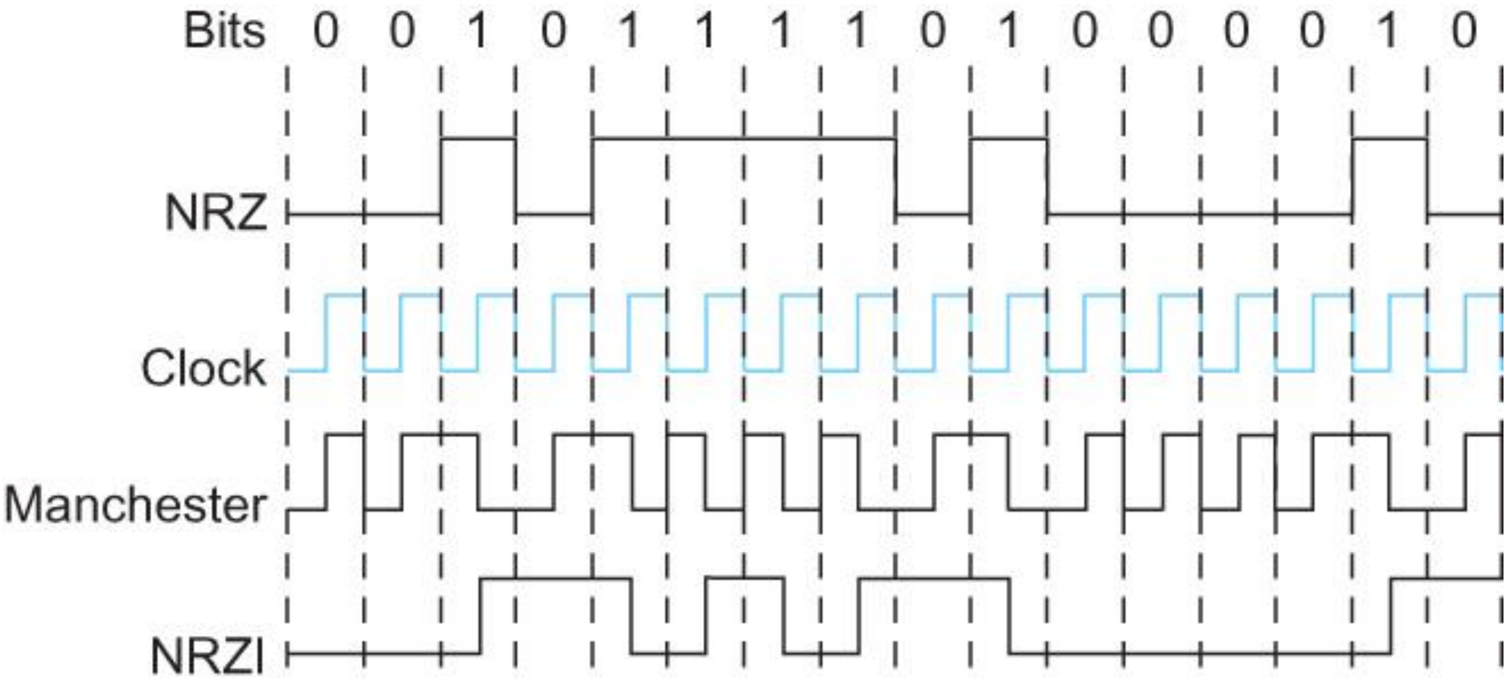
- Problem with Manchester encoding
  - Doubles the rate at which the signal transitions are made on the link
    - Which means the receiver has half of the time to detect each pulse of the signal
  - The rate at which the signal changes is called the link's baud rate
  - In Manchester the bit rate is half the baud rate



bersecurity Lab

QR Code For Q&A

Encoding summary



Different encoding strategies

Funny game!

- Draw NRZ, NRZI, Manchester Code for the following bit string

100011101

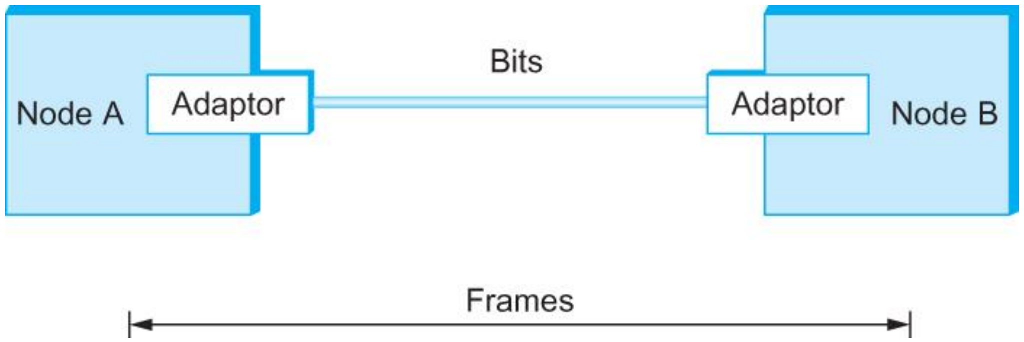
Funny game!

- Draw NRZ, NRZI, Manchester Code for the following bit string

100111001

Framing

- We are focusing on packet-switched networks, which means that blocks of data (called *frames* at this level), not bit streams, are exchanged between nodes.
- It is the network adaptor that enables the nodes to exchange frames.

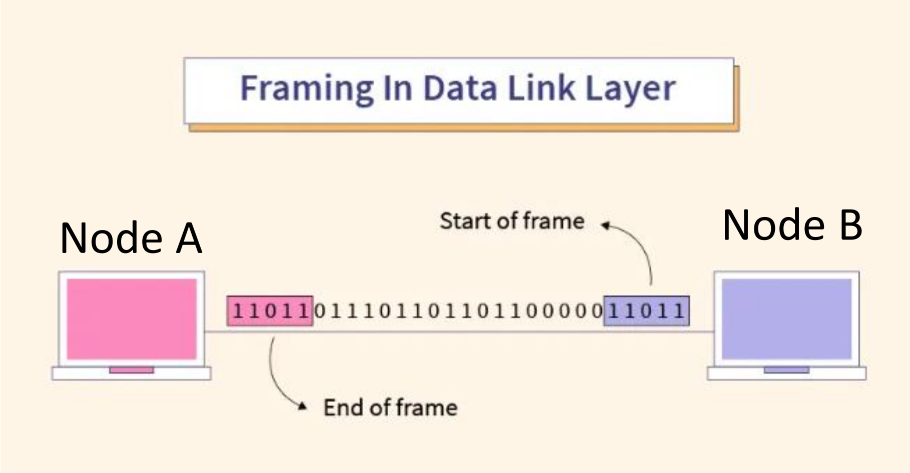


Bits flow between adaptors, frames between hosts

QR Code For Q&A

Framing [key]

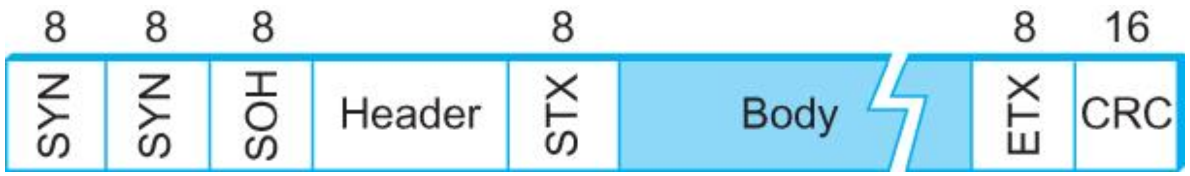
- When node A wishes to transmit a frame to node B, it tells its adaptor to transmit a frame from the node's memory. This results in a sequence of bits being sent over the link.
- The adaptor on node B then collects together the sequence of bits arriving on the link and deposits the corresponding frame in B's memory.
- Recognizing exactly what set of bits constitute a frame—that is, determining where the frame begins and ends—is the central challenge faced by the adaptor



QR Code For Q&A

Framing

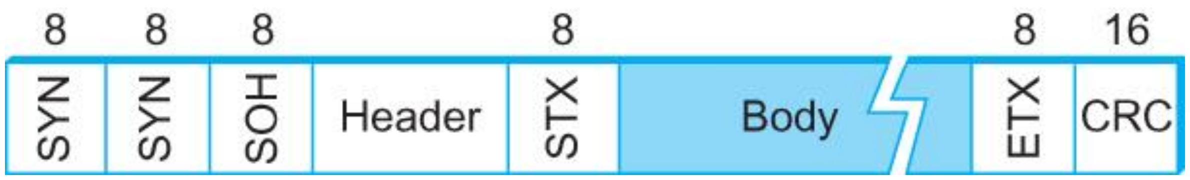
- Byte-oriented Protocols
  - To view each frame as a collection of bytes (characters) rather than bits
  - BISYNC (Binary Synchronous Communication) Protocol
    - Developed by IBM (late 1960)
  - DDCMP (Digital Data Communication Protocol)
    - Used in DECNet



BISYNC Frame Format

Framing

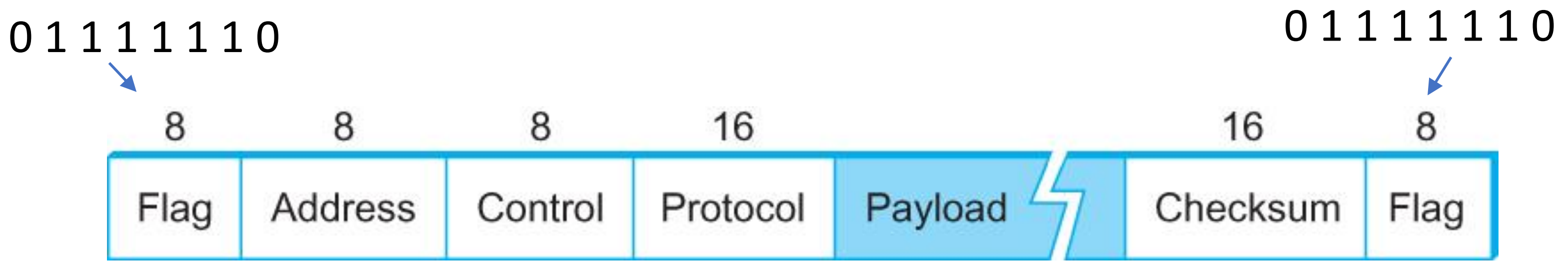
- BISYNC – sentinel approach
  - Frames transmitted beginning with leftmost field
  - Beginning of a frame is denoted by sending a special SYN (synchronize) character
  - Data portion of the frame is contained between special sentinel character STX (start of text) and ETX (end of text)
  - SOH : Start of Header
  - DLE : Data Link Escape
  - CRC: Cyclic Redundancy Check

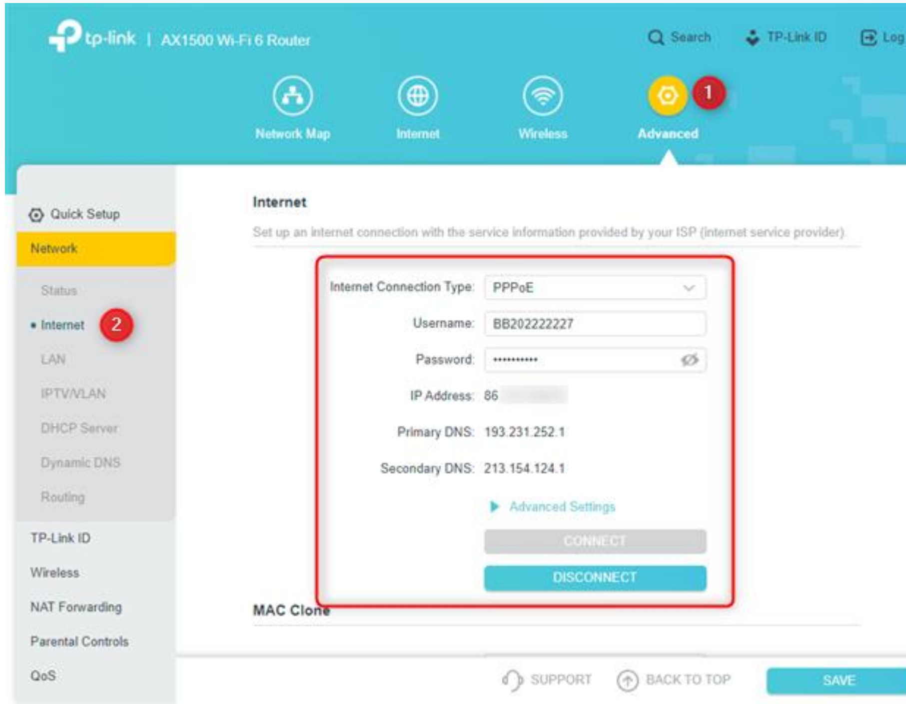


BISYNC Frame Format

Framing [key]

- PPP is commonly run over Internet links uses sentinel approach
  - Special start of text character denoted as Flag
    - 0 1 1 1 1 1 1 0
  - Address, control : default numbers
  - Protocol for demux : IP / IPX
  - Payload : negotiated (1500 bytes)
  - Checksum : for error detection





PPP Frame Format

國立中正大學 Cybersecurity Lab

QR Code For Q&A

Framing [key]

- Bit-oriented Protocol
  - HDLC : High Level Data Link Control
    - Beginning and Ending Sequences

      0 1 1 1 1 1 1 0

HDLC Frame Format

- HDLC Protocol
  - On the sending side, any time five consecutive 1's have been transmitted from the body of the message (i.e. excluding when the sender is trying to send the distinguished 01111110 sequence)
    - The sender inserts 0 before transmitting the next bit

QR Code For Q&A

Framing

- HDLC Protocol
  - On the receiving side
    - 5 consecutive 1's
      - Next bit 0 : Stuffed, so discard it

        1 : Either End of the frame marker

        Or Error has been introduced in the bitstream

        Look at the next bit

        If 0 (01111110) → End of the frame marker

        If 1 (01111111) → Error, discard the whole frame

        The receiver needs to wait for next 01111110 before it can start receiving again

Error Detection

- Bit errors are introduced into frames
  - Because of electrical interference and thermal noises
- Detecting Error
- Correction Error
- Two approaches when the recipient detects an error
  - Notify the sender that the message was corrupted, so the sender can send again.
    - If the error is rare, then the retransmitted message will be error-free
  - Using some error correct detection and correction algorithm, the receiver reconstructs the message

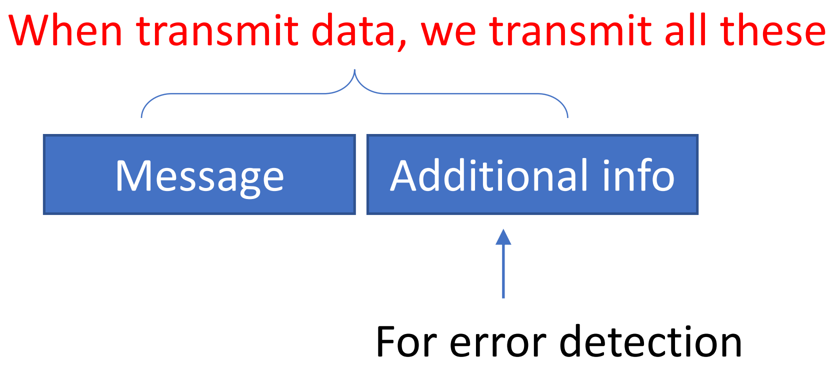
Error Detection

- Common technique for detecting transmission error
  - **CRC** (Cyclic Redundancy Check)
    - Used in HDLC, DDCMP, CSMA/CD, Token Ring
  - Other approaches
    - **Two Dimensional Parity** (BISYNC)
    - **Checksum** (used in IP packet)

QR Code For Q&A

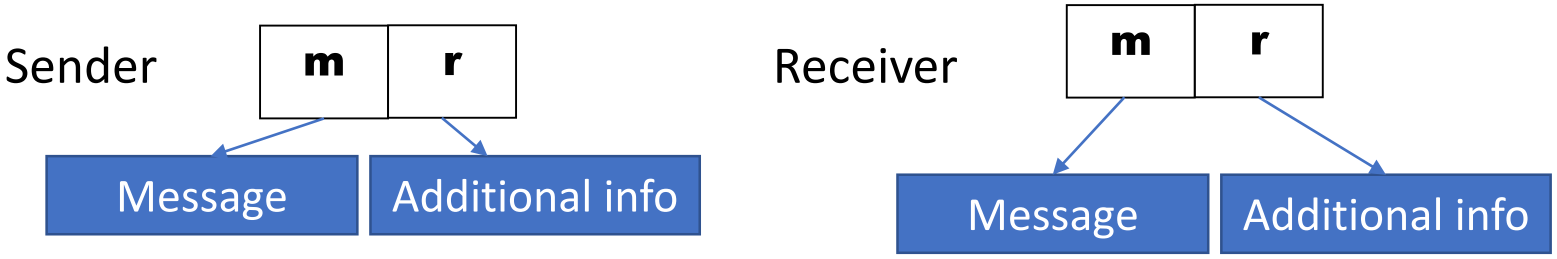
Error Detection

- Basic Idea of Error Detection
  - To add redundant information to a frame that can be used to determine if errors have been introduced
  - Imagine (Extreme Case)
    - Transmitting two complete copies of data
      - Identical → No error
      - Differ → Error
      - Poor Scheme ???
        - n bit message, n bit redundant information
        - Error can go undetected
    - In general, we can provide strong error detection technique
      - k redundant bits, n bits message, k << n
      - In Ethernet, a frame carrying up to 12,000 bits of data requires only 32-bit CRC



Error Detection

- Extra bits are redundant
  - They add no new information to the message
  - Derived from the original message using some algorithm
  - Both the sender and receiver know the algorithm



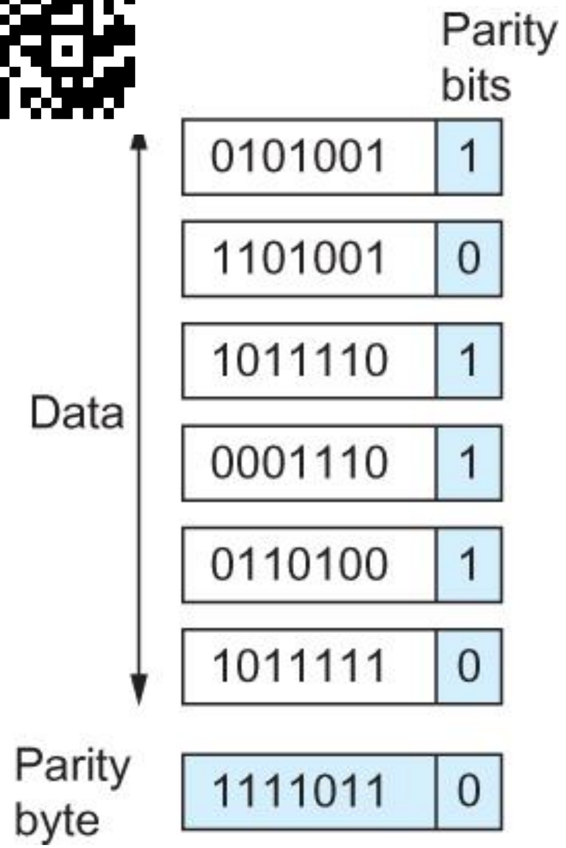
Receiver computes *r* using *m*

If they match, no error

QR Code For Q&A

Two-dimensional parity

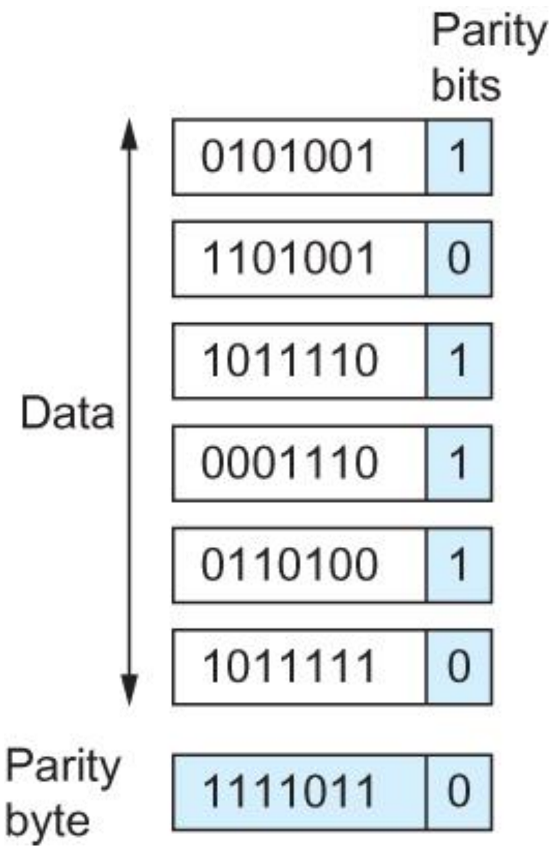
- Two-dimensional parity is exactly what the name suggests
- It is based on “simple” (one-dimensional) parity, which usually involves adding one extra bit to a 7-bit code to balance the number of 1s in the byte. For example,
  - Odd parity sets the eighth bit to 1 if needed to give an odd number of 1s in the byte, and
  - Even parity sets the eighth bit to 1 if needed to give an even number of 1s in the byte



Two Dimensional Parity

Two-dimensional parity

- Two-dimensional parity does a similar calculation for each bit position across each of the bytes contained in the frame
- This results in an extra parity byte for the entire frame, in addition to a parity bit for each byte
- Two-dimensional parity catches all 1-, 2-, and 3-bit errors and most 4-bit errors

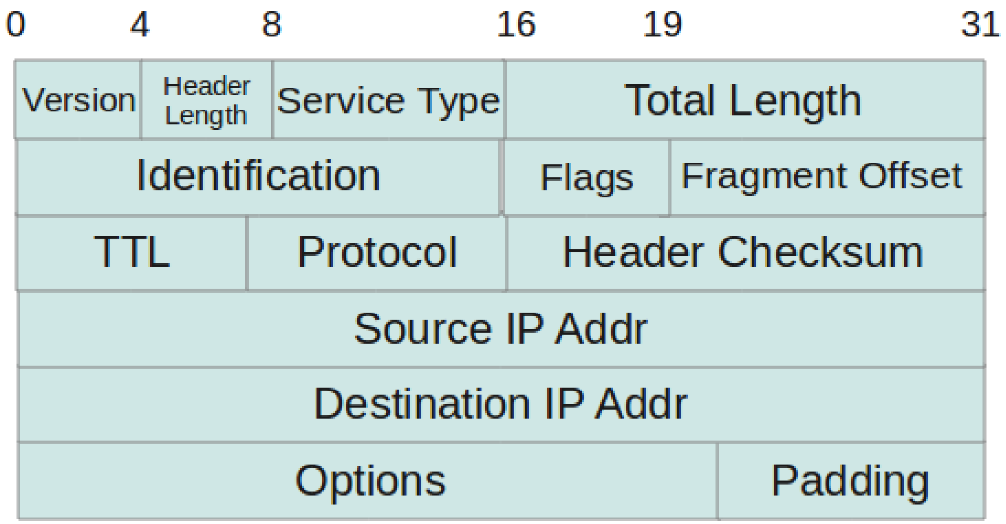


Two Dimensional Parity

QR Code For Q&A

Internet Checksum Algorithm

- The sender sends data + its sum (checksum)
- The receiver performs the same calculation on the received data and compares the result with the received checksum
- If any transmitted data, including the checksum itself, is corrupted, then the results will not match, so the receiver knows that an error occurred



Cybersecurity Lab

QR Code For Q&A

IP Checksum in Wireshark

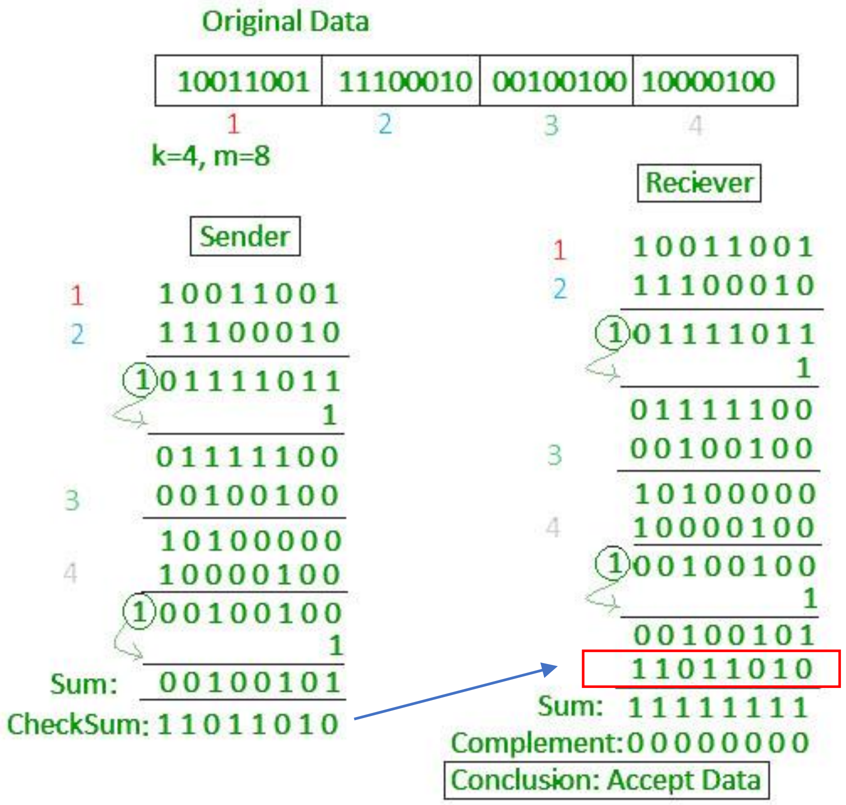
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	145.254.160.237	65.208.228.223	TCP	62	3372 → 80 [SYN] Seq=0 Win=8760 Len=0 MSS=1460 SACK_PERM=1
2	0.911310	65.208.228.223	145.254.160.237	TCP	62	80 → 3372 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1380 SACK_PERM=1
3	0.911310	145.254.160.237	65.208.228.223	TCP	54	3372 → 80 [ACK] Seq=1 Ack=1 Win=9660 Len=0
4	0.911310	145.254.160.237	65.208.228.223	HTTP	533	GET /download.html HTTP/1.1
5	1.472116	65.208.228.223	145.254.160.237	TCP	54	80 → 3372 [ACK] Seq=1 Ack=480 Win=6432 Len=0
6	1.682419	65.208.228.223	145.254.160.237	TCP	1434	80 → 3372 [ACK] Seq=1 Ack=480 Win=6432 Len=1380 [TCP segment of a reassembled PDU]
7	1.812606	145.254.160.237	65.208.228.223	TCP	54	3372 → 80 [ACK] Seq=480 Ack=1381 Win=9660 Len=0
8	1.812606	65.208.228.223	145.254.160.237	TCP	1434	80 → 3372 [ACK] Seq=1381 Ack=480 Win=6432 Len=1380 [TCP segment of a reassembled PDU]
9	2.012894	145.254.160.237	65.208.228.223	TCP	54	3372 → 80 [ACK] Seq=480 Ack=2761 Win=9660 Len=0
10	2.443513	65.208.228.223	145.254.160.237	TCP	1434	80 → 3372 [ACK] Seq=2761 Ack=480 Win=6432 Len=1380 [TCP segment of a reassembled PDU]
11	2.553672	65.208.228.223	145.254.160.237	TCP	1434	80 → 3372 [PSH, ACK] Seq=4141 Ack=480 Win=6432 Len=1380 [TCP segment of a reassembled PDU]
12	2.553672	145.254.160.237	65.208.228.223	TCP	54	3372 → 80 [ACK] Seq=480 Ack=5521 Win=9660 Len=0
13	2.553672	145.254.160.237	145.253.2.203	DNS	89	Standard query 0x0023 A pagead2.googlesyndication.com
14	2.633787	65.208.228.223	145.254.160.237	TCP	1434	80 → 3372 [ACK] Seq=5521 Ack=480 Win=6432 Len=1380 [TCP segment of a reassembled PDU]
15	2.814046	145.254.160.237	65.208.228.223	TCP	54	3372 → 80 [ACK] Seq=480 Ack=6901 Win=9660 Len=0

```
> Frame 14: 1434 bytes on wire (11472 bits), 1434 bytes captured (11472 bits)
> Ethernet II, Src: fe:ff:20:00:01:00 (fe:ff:20:00:01:00), Dst: Xerox_00:00:00 (00:00:01:00:00:00)
v Internet Protocol Version 4, Src: 65.208.228.223, Dst: 145.254.160.237
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 1420
    Identification: 0xc0a3 (49315)
  > Flags: 0x40, Don't fragment
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 47
    Protocol: TCP (6)
    Header Checksum: 0x2c2d [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 65.208.228.223
    Destination Address: 145.254.160.237
> Transmission Control Protocol, Src Port: 80, Dst Port: 3372, Seq: 5521, Ack: 480, Len: 1380
```

QR Code For Q&A

Internet Checksum Algorithm

- **Step 1:** Convert the data segment into a series of 16-bit integers;
- **Step 2:** Calculate the sum of all 16-bit integers, allowing for the carry bit wrap-around;
- **Step 3:** Add the checksum to the final sum total;
- **Step 4:** If the final total is all 1’s the data is validated;
- **Step 5:** If any 0’s are detected the data has been corrupted.



Cyclic Redundancy Check (CRC)

- A common method of detecting accidental changes/errors in the communication channel.
- CRC uses **Generator Polynomial** which is available on both sender and receiver side
- An example generator polynomial is of the form like $x^3 + x + 1$. This generator polynomial represents key 1011.

```
n : Number of bits in data to be sent from sender side.

k : Number of bits in the key obtained from generator
polynomial.
```

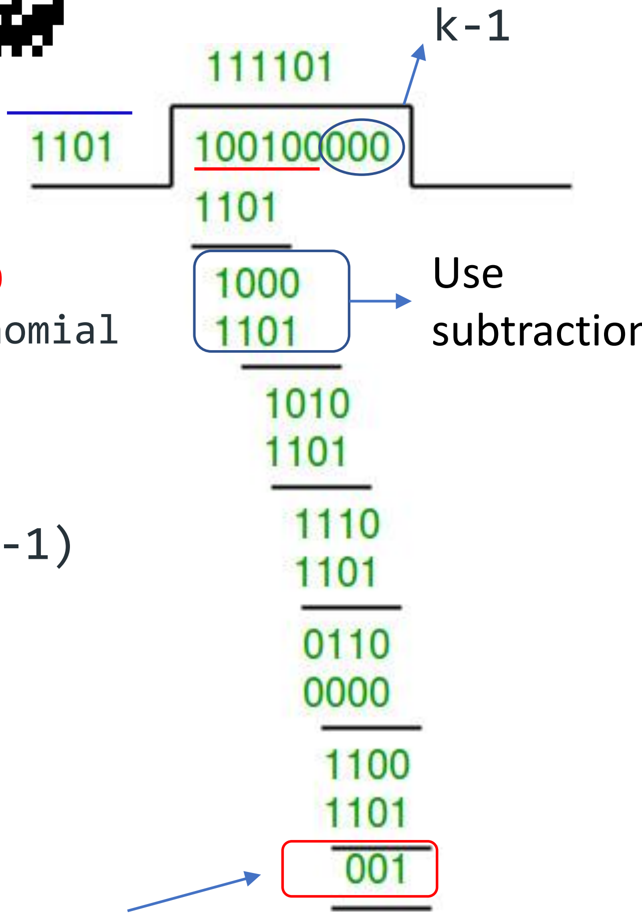
QR Code For Q&A

Step to calculate CRC

- **Step 1:** The binary data is first augmented by adding k-1 zeros in the end of the data
- **Step 2:** Use ***modulo-2 binary division*** to divide binary data by the key and store remainder of division.
- **Step 3:** Append the remainder at the end of the data to form the encoded data and send the same

Data word to be sent - 100100
Key - 1101 Or generator polynomial
$x^3 + x^2 + 1$]
Sender Side:

n : 6
k : 3 (=4-1)



Remainder

n : Number of bits in data to be sent from sender side.
k : Number of bits in the key obtained from generator polynomial.

the remainder is 001 and hence the encoded data sent is 100100001.

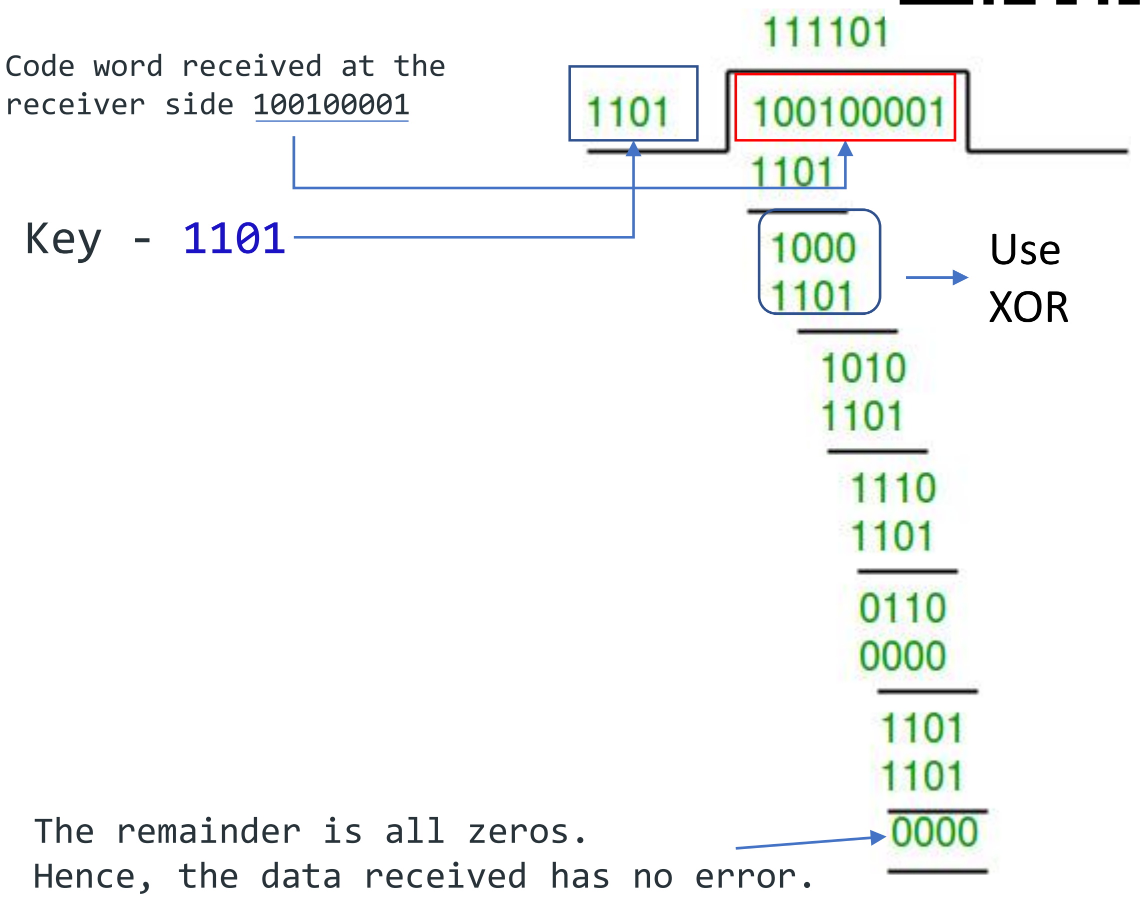
國立中正大學 Cybersecurity Lab

QR Code For Q&A

At the receiver side

- **Step 1:** The binary data is first augmented by adding k-1 zeros in the end of the data
- **Step 2:** Use ***modulo-2 binary division*** (XOR, instead of subtraction) to divide binary data by the key and store remainder of division.
- **Step 3:** If the remainder is all zeros, there is no error. Otherwise, there is an

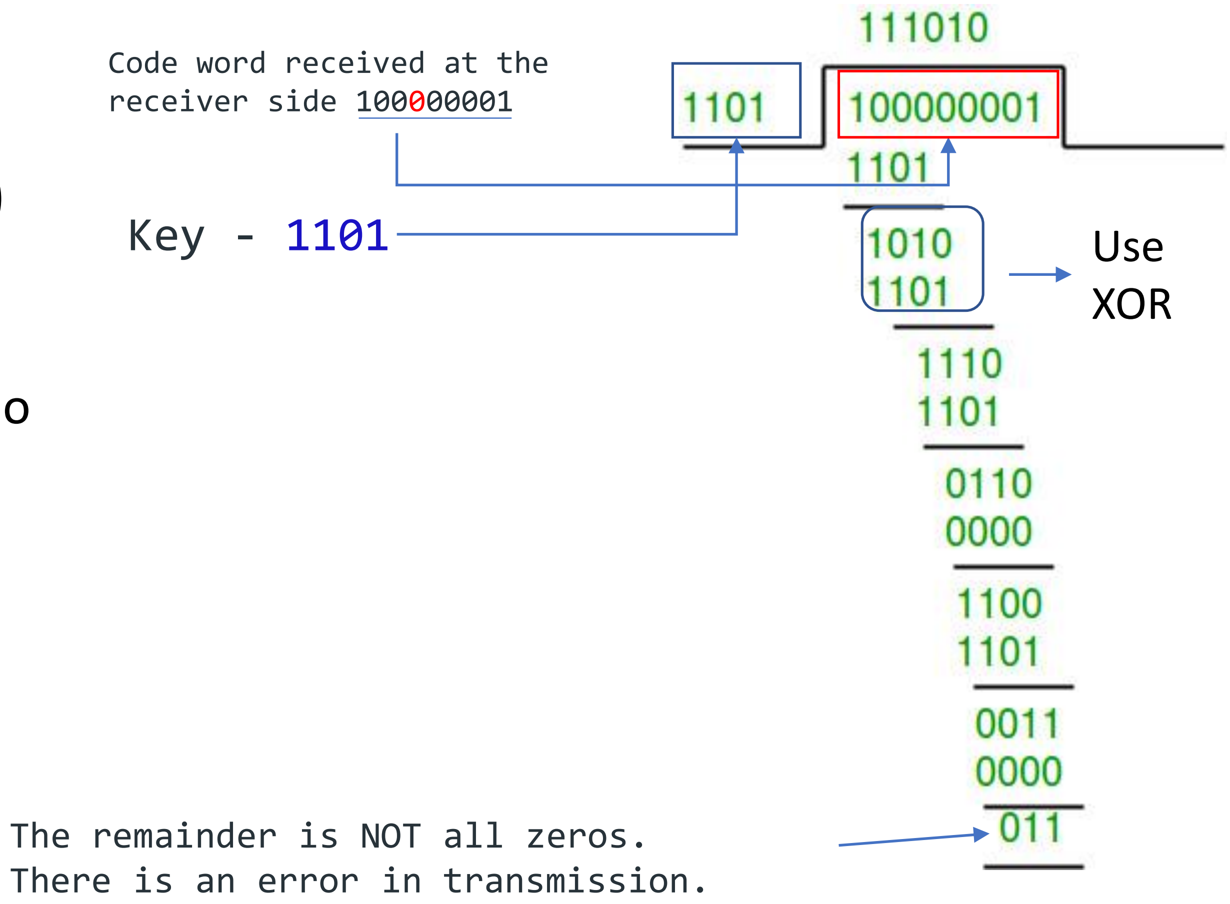
[no error]



At the receiver side

- **Step 1:** The binary data is first augmented by adding k-1 zeros in the end of the data
- **Step 2:** Use ***modulo-2 binary division*** (XOR) to divide binary data by the key and store remainder of division.
- **Step 3:** If the remainder is all zeros, there is no error. Otherwise, there is an error

[With error]



Another example

The sender side

Original data bit to send 10011010

```
                          11111001
Generator  →  1101 )10011010000  ←  Message
                    1101
                     1001
                     1101
                      1000
                      1101
                       1011
                       1101
                        1100
                        1101
                           1000
                           1101
                            101  ←  Remainder
```

New data bit to send [10011010101]

CRC Calculation using Polynomial Long Division

Cyclic Redundancy Check (CRC)

- Six generator polynomials that have become international standards are:
  - CRC-8 = x^8+x^2+x+1
  - CRC-10 = $x^{10}+x^9+x^5+x^4+x+1$
  - CRC-12 = $x^{12}+x^{11}+x^3+x^2+x+1$
  - CRC-16 = $x^{16}+x^{15}+x^2+1$
  - CRC-CCITT = $x^{16}+x^{12}+x^5+1$
  - CRC-32 = $x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1$

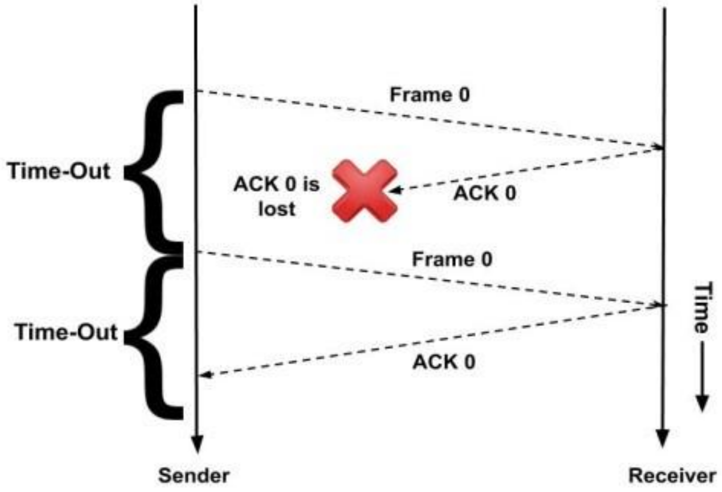
QR Code For Q&A

Reliable Transmission

- CRC is used to detect errors.
- Some error codes are strong enough to correct errors.
- The overhead is typically too high.
- Corrupt frames must be discarded.
- A link-level protocol that wants to deliver frames reliably must recover from these discarded frames.
- This is accomplished using a combination of two fundamental mechanisms
  - Acknowledgements and Timeouts

Reliable Transmission

- An *acknowledgement* (ACK for short) is a small control frame that a protocol sends back to its peer saying that it has received the earlier frame.
  - A control frame is a frame with header only (no data).
- The receipt of an *acknowledgement* indicates to the sender of the original frame that its frame was successfully delivered.
- If the sender does not receive an *acknowledgment* after a reasonable amount of time, then it retransmits the original frame.
- The action of waiting a reasonable amount of time is called a *timeout*.
- The general strategy of using *acknowledgements* and *timeouts* to implement reliable delivery is sometimes called Automatic Repeat reQuest (ARQ).



Stop and Wait Protocol

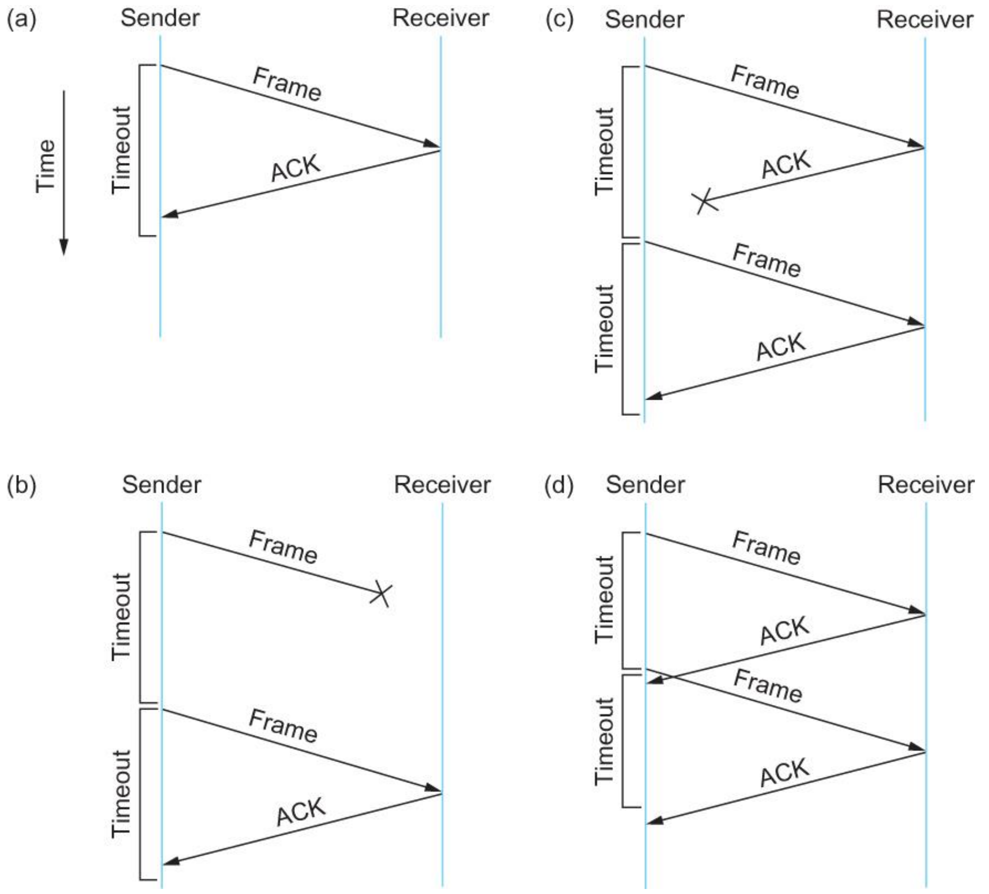
- Is a method to for error and flows control
- Idea of stop-and-wait protocol is straightforward
  - After transmitting one frame, the sender waits for an acknowledgement before transmitting the next frame.
  - If the acknowledgement does not arrive after a certain period of time, the sender times out and retransmits the original frame

Stop and Wait Protocol

QR Code For Q&A

Wait for how long?

1. RoundTripTime (**RTT**) = Amount of time taken by a packet to reach the receiver + Time taken by the Acknowledgement to reach the sender
2. TimeOut (**TO**) = 2* RTT
3. Time To Live (**TTL**) = 2* TimeOut. (Maximum TTL is 255 seconds)



Timeline showing four different scenarios for the stop-and-wait algorithm.
(a) The ACK is received before the timer expires; (b) the original frame is lost; (c) the ACK is lost; (d) the timeout fires too soon

國立中正大學 Cybersecurity Lab

Stop and Wait Protocol

- If the acknowledgment is lost or delayed in arriving
  - The sender times out and retransmits the original frame, but the receiver will think that it is the next frame since it has correctly received and acknowledged the first frame
  - As a result, duplicate copies of frames will be delivered

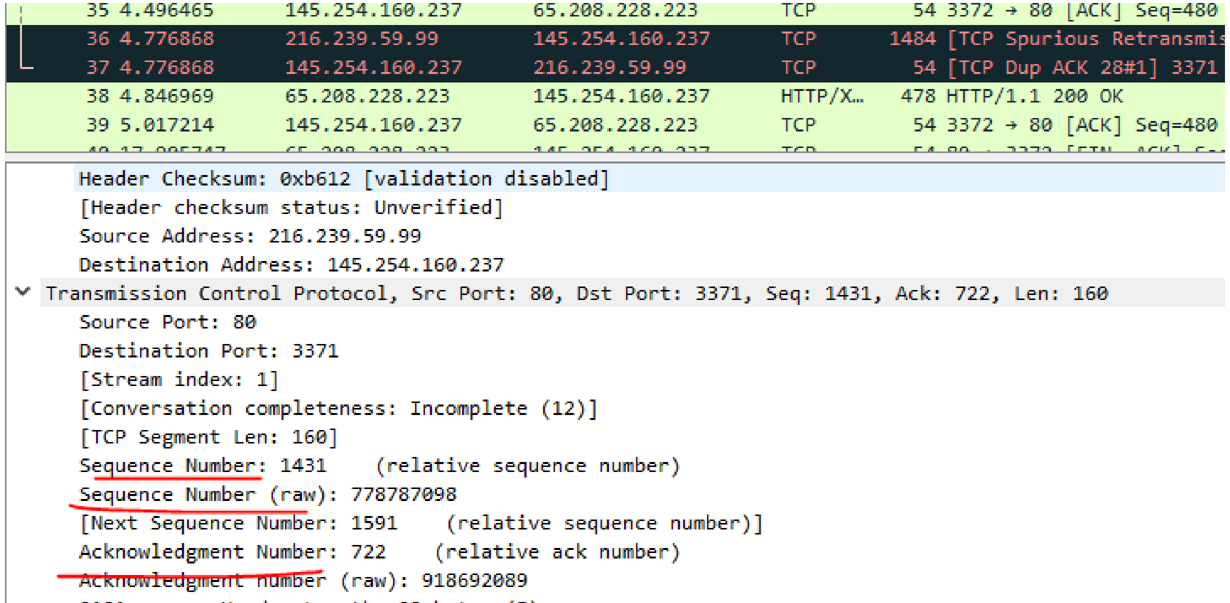
- How to solve this
  - Use 1 bit sequence number (0 or 1)
  - When the sender retransmits frame 0, the receiver can determine that it is seeing a second copy of frame 0 rather than the first copy of frame 1 and therefore can ignore it (the receiver still acknowledges it, in case the first acknowledgement was lost)

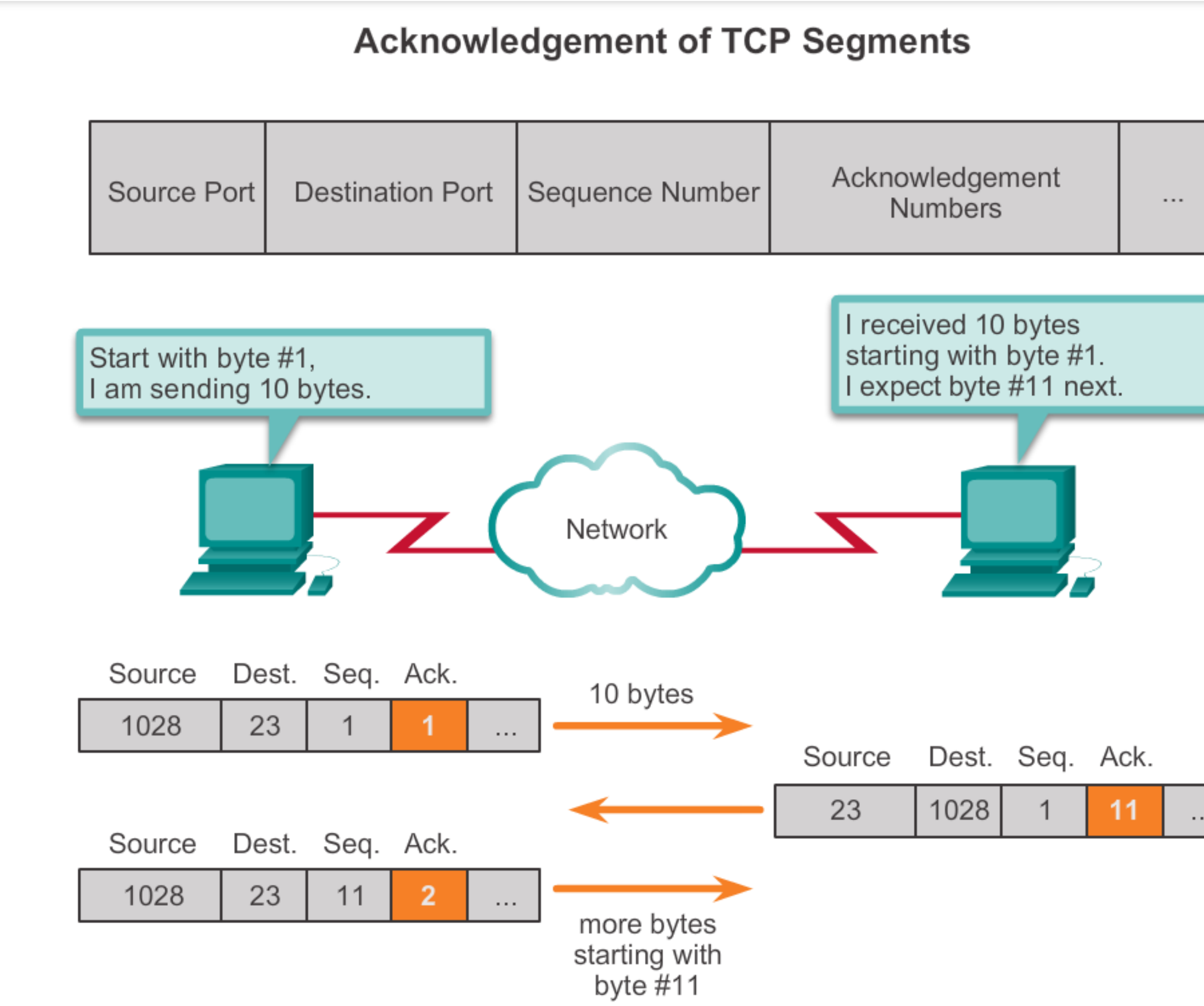
Sequence number

- ACK and Sequence Number

- To determine the data

order and retransmission data





QR Code For Q&A

Stop and Wait Protocol

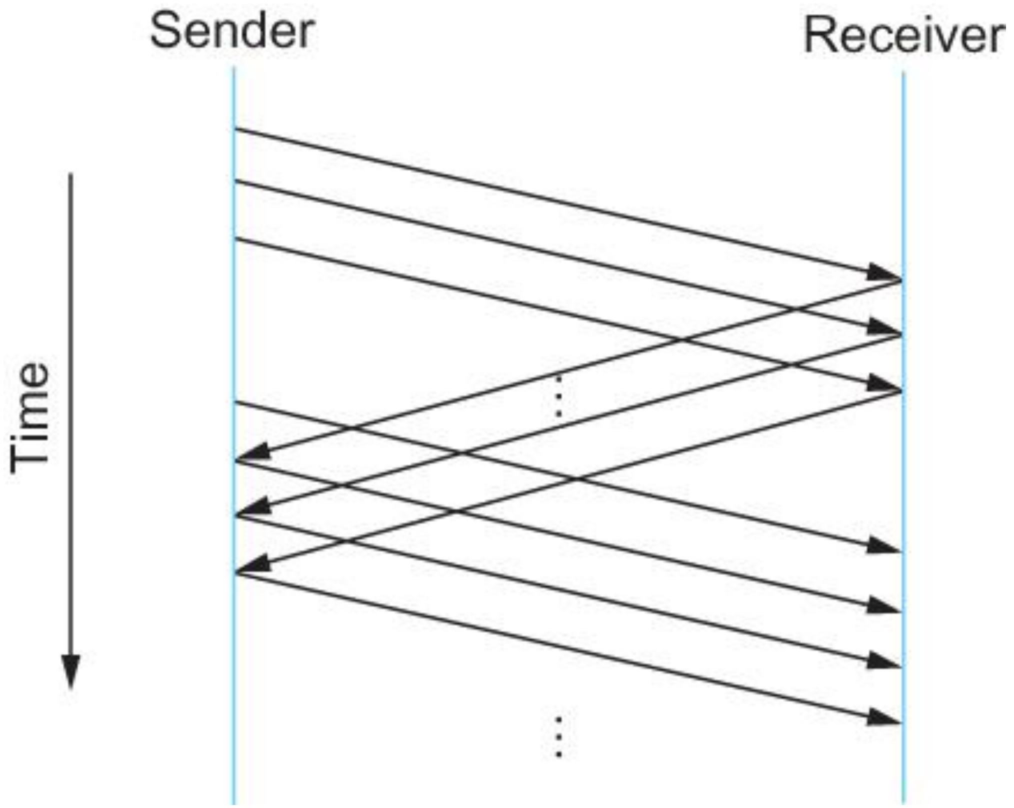
- The sender has only one outstanding frame on the link at a time
  - This may be far below the link's capacity
- Consider a 1.5 Mbps link with a 45 ms RTT
  - The link has a delay × bandwidth product of 67.5 Kb or approximately 8 KB
  - Since the sender can send only one frame per RTT and assuming a frame size of 1 KB
  - Maximum Sending rate
    - Bits per frame ÷ Time per frame = 1024 × 8 ÷ 0.045 = 182 Kbps

      Or about one-eighth of the link's capacity
  - To use the link fully, then sender should transmit up to eight frames before having to wait for an acknowledgement

QR Code For Q&A

Sliding Window Protocol

1. RoundTripTime (**RTT**) = Amount of time taken by a packet to reach the receiver + Time taken by the Acknowledgement to reach the sender
2. TimeOut (**TO**) = 2* RTT
3. Time To Live (**TTL**) = 2* TimeOut. (Maximum TTL is 255 seconds)



Timeline for Sliding Window Protocol

國立中正大學 Cybersecurity Lab

Sliding window size in Wireshark

No.	Time	Source	Destination	Protocol	Length	Info
27	3.955688	216.239.59.99	145.254.160.237	HTTP	214	HTTP/1.1 200 OK (text/html)
28	3.955688	145.254.160.237	216.239.59.99	TCP	54	3371 → 80 [ACK] Seq=722 Ack=1591 Win=8760 Len=0
29	4.105904	65.208.228.223	145.254.160.237	TCP	1434	80 → 3372 [PSH, ACK] Seq=12421 Ack=480 Win=6432 Len=1380 [TCP segment of
30	4.216062	145.254.160.237	65.208.228.223	TCP	54	3372 → 80 [ACK] Seq=480 Ack=13801 Win=9660 Len=0
31	4.226076	65.208.228.223	145.254.160.237	TCP	1434	80 → 3372 [ACK] Seq=13801 Ack=480 Win=6432 Len=1380 [TCP segment of a re
32	4.356264	65.208.228.223	145.254.160.237	TCP	1434	80 → 3372 [ACK] Seq=15181 Ack=480 Win=6432 Len=1380 [TCP segment of a re
33	4.356264	145.254.160.237	65.208.228.223	TCP	54	3372 → 80 [ACK] Seq=480 Ack=16561 Win=9660 Len=0
34	4.496465	65.208.228.223	145.254.160.237	TCP	1434	80 → 3372 [ACK] Seq=16561 Ack=480 Win=6432 Len=1380 [TCP segment of a re
35	4.496465	145.254.160.237	65.208.228.223	TCP	54	3372 → 80 [ACK] Seq=480 Ack=17941 Win=9660 Len=0
36	4.776868	216.239.59.99	145.254.160.237	TCP	1484	[TCP Spurious Retransmission] 80 → 3371 [PSH, ACK] Seq=1 Ack=722 Win=314
37	4.776868	145.254.160.237	216.239.59.99	TCP	54	[TCP Dup ACK 28#1] 3371 → 80 [ACK] Seq=722 Ack=1591 Win=8760 Len=0
38	4.846969	65.208.228.223	145.254.160.237	HTTP/X…	478	HTTP/1.1 200 OK
39	5.017214	145.254.160.237	65.208.228.223	TCP	54	3372 → 80 [ACK] Seq=480 Ack=18365 Win=9236 Len=0

Header Checksum: 0xb612 [validation disabled]
[Header checksum status: Unverified]
Source Address: 216.239.59.99
Destination Address: 145.254.160.237
Transmission Control Protocol, Src Port: 80, Dst Port: 3371, Seq: 1431, Ack: 722, Len: 160
Source Port: 80
Destination Port: 3371
[Stream index: 1]
[Conversation completeness: Incomplete (12)]
[TCP Segment Len: 160]
Sequence Number: 1431 (relative sequence number)
Sequence Number (raw): 778787098
[Next Sequence Number: 1591 (relative sequence number)]
Acknowledgment Number: 722 (relative ack number)
Acknowledgment number (raw): 918692089
0101 .... = Header Length: 20 bytes (5)
Flags: 0x018 (PSH, ACK)
Window: 31460
[Calculated window size: 31460]
[~~Window size scaling factor: -1~~ (unknown)]
Checksum: 0xde29 [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0

ty Lab

Sliding Window Protocol

- Sender assigns a sequence number denoted as SeqNum to each frame.
  - Assume it can grow infinitely large

- Sender maintains three variables
  - Sending Window Size (SWS)
    - Upper bound on the number of outstanding (unacknowledged) frames that the sender can transmit
  - Last Acknowledgement Received (LAR)
    - Sequence number of the last acknowledgement received
  - Last Frame Sent (LFS)
    - Sequence number of the last frame sent

Sliding Window Protocol

- Receiver maintains three variables
  - Receiving Window Size (RWS)
    - Upper bound on the number of out-of-order frames that the receiver is willing to accept
  - Largest Acceptable Frame (LAF)
    - Sequence number of the largest acceptable frame
  - Last Frame Received (LFR)
    - Sequence number of the last frame received

Issues with Sliding Window Protocol

- When timeout occurs, the amount of data in transit decreases
  - Since the sender is unable to advance its window

- When the packet loss occurs, this scheme is no longer keeping the pipe full
  - The longer it takes to notice that a packet loss has occurred, the more severe the problem becomes

- How to improve this
  - Negative Acknowledgement (NAK)
  - Additional Acknowledgement
  - Selective Acknowledgement

Issues with Sliding Window Protocol

- Negative Acknowledgement (NAK)
  - Receiver sends NAK for frame 6 when frame 7 arrive (in the previous example)
    - However this is unnecessary since sender's timeout mechanism will be sufficient to catch the situation

- Additional Acknowledgement
  - Receiver sends additional ACK for frame 5 when frame 7 arrives
    - Sender uses duplicate ACK as a clue for frame loss

- Selective Acknowledgement
  - Receiver will acknowledge exactly those frames it has received, rather than the highest number frames
    - Receiver will acknowledge frames 7 and 8
    - Sender knows frame 6 is lost
    - Sender can keep the pipe full (additional complexity)

Issues with Sliding Window Protocol

How to select the window size

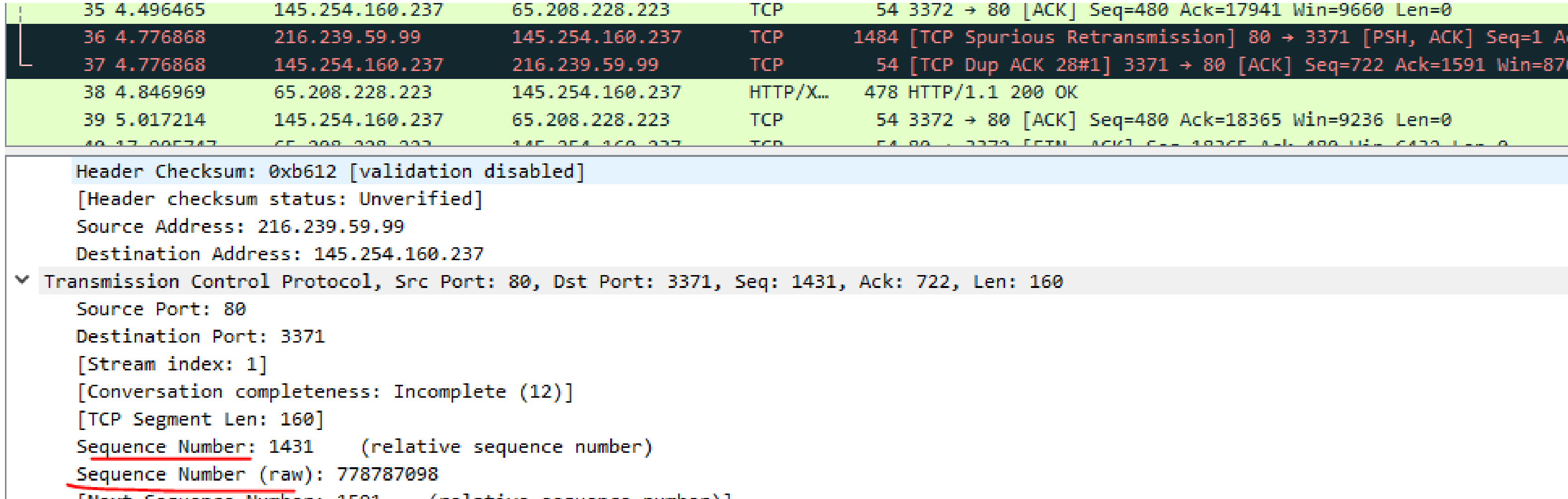
- SWS is easy to compute
  - Delay × Bandwidth
- RWS can be anything
  - Two common setting
    - RWS = 1

      No buffer at the receiver for frames that arrive out of order
    - RWS = SWS

      The receiver can buffer frames that the sender transmits

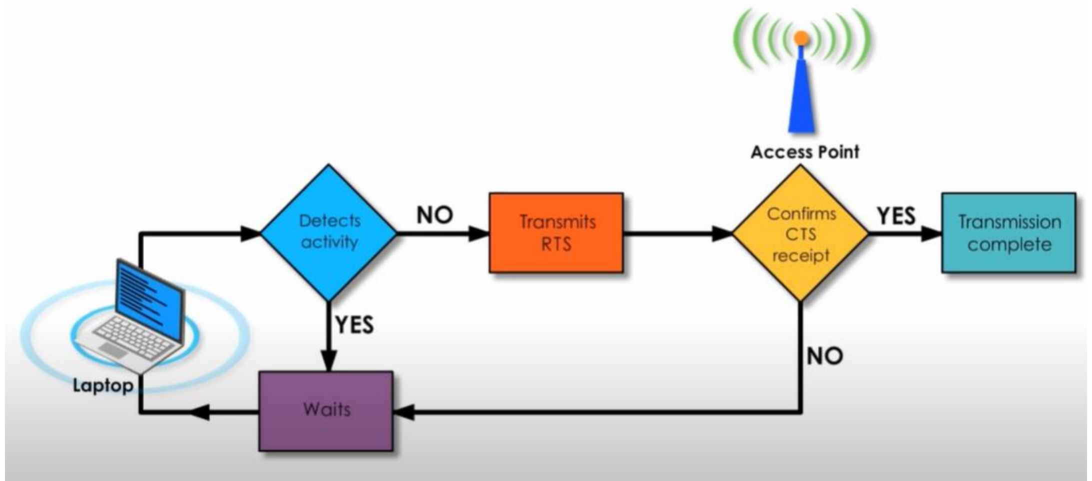
Issues with Sliding Window Protocol

- How to distinguish between different incarnations of the same sequence number?
  - Number of possible sequence number must be larger than the number of outstanding frames allowed
    - Stop and Wait: One outstanding frame
      - 2 distinct sequence number (0 and 1)
    - Let MaxSeqNum be the number of available sequence numbers
    - SWS + 1 ≤ MaxSeqNum
      - Is this sufficient?



Ethernet

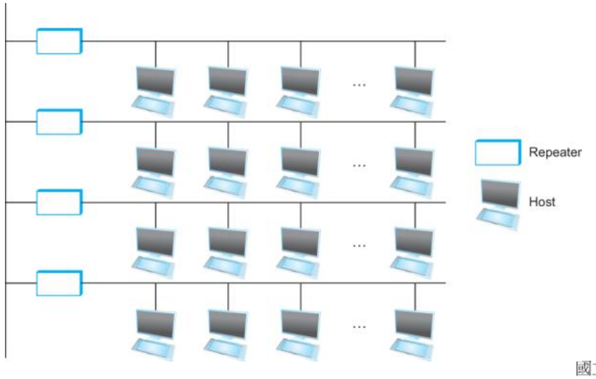
- Most successful local area networking technology of last 20 years.
- Developed in the mid-1970s by researchers at the Xerox Palo Alto Research Centers (PARC).
- Uses CSMA/CD technology
  - Carrier Sense Multiple Access with Collision Detection.
  - A set of nodes send and receive frames over a shared link.
  - Carrier sense means that all nodes can distinguish between an idle and a busy link.
  - Collision detection means that a node listens as it transmits and can therefore detect when a frame it is transmitti has collided with a frame transmitted by another node.



QR Code For Q&A

Ethernet

- This standard formed the basis for IEEE standard 802.3
- More recently 802.3 has been extended to include a 100-Mbps version called Fast Ethernet and a 1000-Mbps version called Gigabit Ethernet.
- Multiple Ethernet segments can be joined together by *repeaters.*
- A *repeater* is a device that forwards digital signals.
- No more than four repeaters may be positioned between any pair of hosts.
    - An Ethernet has a total reach of only 2500 m.



QR Code For Q&A

Ethernet

- Any signal placed on the Ethernet by a host is broadcast over the entire network
  - Signal is propagated in both directions.
  - Repeaters forward the signal on all outgoing segments.
  - Terminators attached to the end of each segment absorb the signal.
- Ethernet uses Manchester/Other encoding schemes.

Ethernet LAN Standard	Symbol Encoding	Symbol Rate (Mbaud)	Data Encoding	Data Bits/Symbol	Pairs/Transmit Channel	Number of Pairs Used	Minimum Cable Category Required
10BaseT	Manchester	10	None	1	1	2	3
100BaseT4	Multi-level, 2T/Hz	25	8B6T	8/6	3	4	3
100BaseTX	MLT-3	125	4B5B	4/5	1	2	5
100BaseT2	PAM5x5 (2D-PAM5)	25	None	2	2	2	3
1000BaseT	4D-PAM5	125	None	2	4	4	5 (5e)
10GBase-T	DSQ128 (2D-PAM16)	800	LDPC(1723,2048), 64B/65B, CRC8	3.125	4	4	5e (6a)

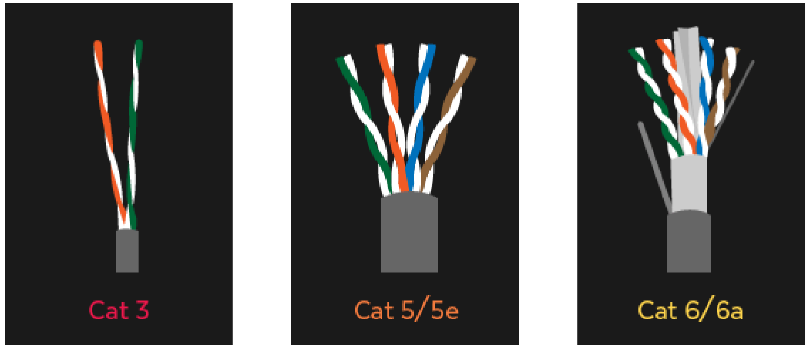
Ethernet

- New Technologies in Ethernet
  - Instead of using coax cable, an Ethernet can be constructed from a thinner cable, e.g., as 10Base2, 100BaseT
    - Base means the cable is used in a baseband system
    - 2 means that a given segment can be no longer than 200 m

Different Ethernet Categories

	Category 3	Category 5	Category 5e	Category 6	Category 6a	Category 7
Cable Type	UTP	UTP	UTP	UTP or STP	STP	S/FTP
Max. Data Transmission Speed	10 Mbps	10/100/1000 Mbps	10/100/1000 Mbps	10/100/1000 Mbps	10,000 Mbps	10,000 Mbps
Max. Bandwidth	16 MHz	100 MHz	100 MHz	250 MHz	500 MHz	600 MHz

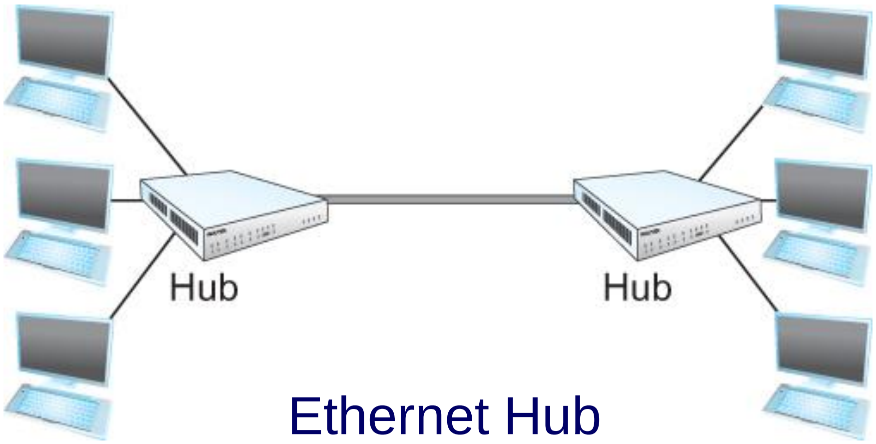
Category Cable Wiring



QR Code For Q&A

Ethernet

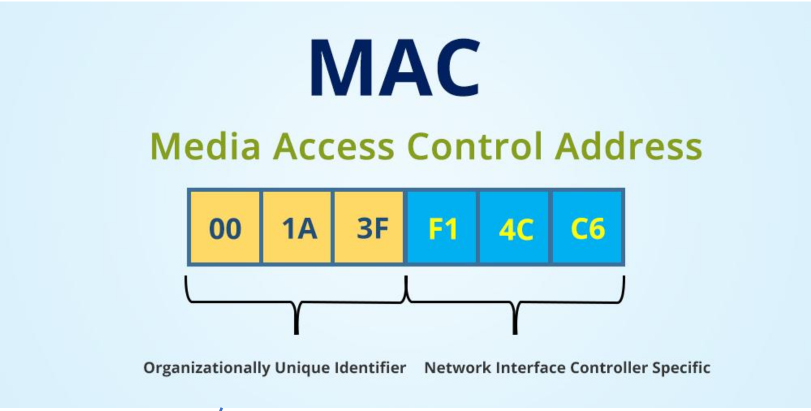
- New Technologies in Ethernet
  - Another cable technology is 10BaseT
    - T stands for twisted pair
    - Limited to 100 m in length
  - With 10BaseT, the common configuration is to have several point to point segments coming out of a multiway repeater, called *Hub*



Ethernet Hub

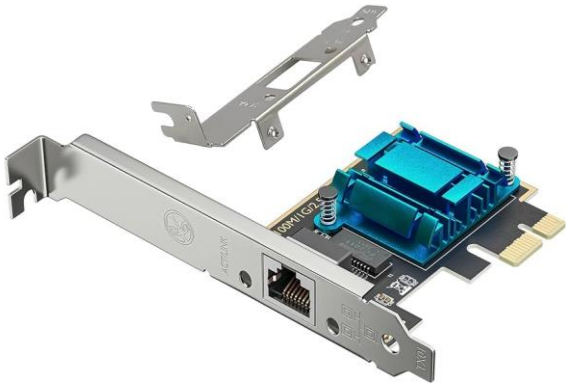
Hub is an old technology

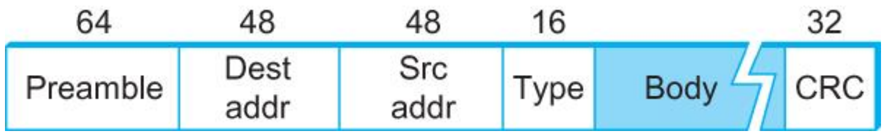
Access Protocol for Ethernet



- The algorithm is commonly called Ethernet's Media Access Control (MAC
  - It is implemented in Hardware on the network adaptor.
- Frame format
  - Preamble (64bit): allows the receiver to synchronize with the signal (sequence of alternating 0s and 1s).
  - Host and Destination Address (48bit each).
  - Packet type (16bit): acts as demux key to identify the higher level protocol.
  - Data (up to 1500 bytes)
    - Minimally a frame must contain at least 46 bytes of data.
    - Frame must be long enough to detect collision.
  - CRC (32bit)

Asus/Tp-Link





Ethernet Frame Format

QR Code For Q&A

Ethernet Addresses

- Each host on an Ethernet (in fact, every Ethernet host in the world) has a unique Ethernet Address.
- The address belongs to the adaptor, not the host.
  - It is usually burnt into ROM.
- Ethernet addresses are typically printed in a human readable format
  - As a sequence of six numbers separated by colons.
  - Each number corresponds to 1 byte of the 6 byte address and is given by a pair of hexadecimal digits, one for each of the 4-bit nibbles in the byte
  - Leading 0s are dropped.
  - For example, 8:0:2b:e4:b1:2 is
    - 00001000 00000000 00101011 11100100 10110001 00000010

QR Code For Q&A

Ethernet Transmitter Algorithm

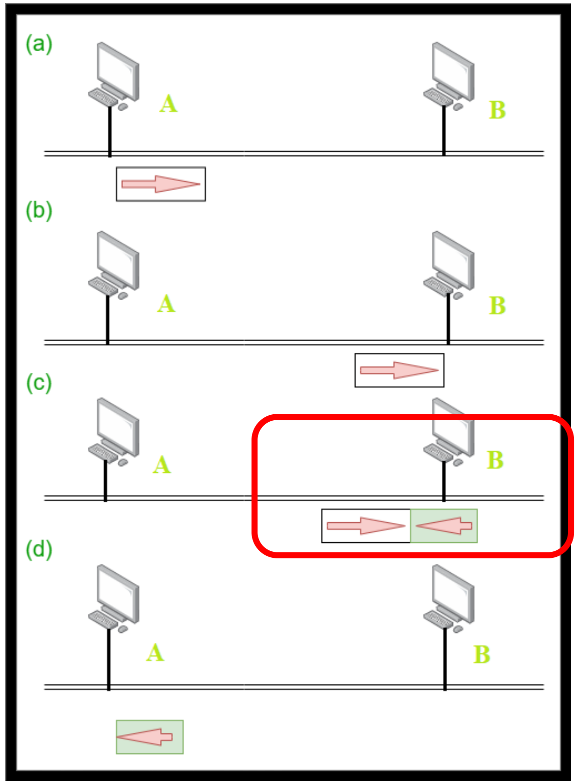
- Wired/Wireless Channel state:
  - ✓ Idle: The adaptor transmits frames.
    - The upper bound of 1500 bytes in the message means that the adaptor can occupy the line for a fixed length of time.
  - ✓ Busy: The adaptor waits for the line to go idle
  - ✓ Collision: Stop transmitting. After a random amount of time, it will sense the channel (if idle) before transmitting
    - It first makes sure to transmit (64-bit preamble + 32-bit jamming sequence = 96 bits) and then stops transmission
    - Each time the adaptor tries to transmit but fails, it doubles the amount of time it waits before trying again.
    - The strategy of doubling the delay interval between each retransmission attempt is known as *Exponential Backoff*

國立中正大學 Cybersecurity Lab

Ethernet Transmitter Algorithm

- The worst case scenario happens when the two hosts are at opposite ends of the Ethernet.
- To know for sure that the frame its just sent did not collide with another frame, the transmitter may need to send as many as 512 bits.
  - Every Ethernet frame must be at least 512 bits (64 bytes) long = 14 bytes of header + 46 bytes of data + 4 bytes of CRC
- Why 512 bits?
  - Why is its length limited to 2500 m?
- The farther apart two nodes are, the longer it takes for a frame sent by one to reach the other, and the network is vulnerable to collision during this time

The worst case



Worst-case scenario: (a) A sends a frame at time *t; (b) A's frame arrives* at B at time *t* + *d; (c) B begins transmitting at time t + d and collides with A's frame;* (d) B's runt (32-bit) frame arrives at A at time *t + 2d.*

Experience with Ethernet

- Ethernets work best under lightly loaded conditions.
  - Under heavy loads, too much of the network's capacity is wasted by collisions.
- Most Ethernets are used in a conservative way.
  - Have fewer than 200 hosts connected to them which is far fewer than the maximum of 1024.
- Most Ethernets are far shorter than 2500m with a round-trip delay of closer to 5 μs than 51.2 μs.
- Ethernets are easy to administer and maintain.
  - There are no switches that can fail and no routing and configuration tables that have to be kept up-to-date.
  - It is easy to add a new host to the network.
  - It is inexpensive.
    - Cable is cheap, and only other cost is the network adaptor on each host.

Summary

QR Code For Q&A

- Links
- Encoding
- CRC
- Slide window
- Ethernet and Ethernet Address