IP Addressing and Subnetting Exercises

Module: Communications Protocols and Distributed Systems

Dr. Mohamed Amine Ferrag
ferrag.mohamedamine@univ-guelma.dz

Guelma University
Year 2024/2025
Master 1 SIQ

Exercise 1: Conversion to CIDR Notation

Question: Convert the subnet mask `255.255.255.0` to CIDR notation.
Hint: Count the number of 1 bits in the binary representation of the mask.

Solution: CIDR (Classless Inter-Domain Routing) notation is a compact representation of an IP address and its associated network mask. It is written as the IP address, followed by a slash (`/`), and then the number of consecutive 1 bits in the subnet mask.

For the subnet mask `255.255.255.0`:
In binary, it is:
`11111111.11111111.11111111.00000000`
There are 24 bits set to 1.
Therefore, the CIDR notation is /**24**.

Exercise 2: Calculating the Number of Usable Hosts

Question: For a subnet defined by the `/26` notation:

1. How many bits are reserved for hosts?
2. What is the total number of IP addresses in this subnet?
3. How many IP addresses are usable (excluding the network and broadcast addresses)?

Hint: Use the formula $2^{(\text{number of host bits})}$ and remember to subtract 2 for the network and broadcast addresses.

Solution: In a /26 subnet, 26 bits are used for the network portion, leaving $32-26 = 6$ bits for hosts.

1. The number of bits reserved for hosts is **6**.
2. The total number of IP addresses is $2^6 = \mathbf{64}$.

3. The number of usable IP addresses is $64 - 2 = \mathbf{62}$ (subtracting the network and broadcast addresses).

Exercise 3: Determining the Address Range of a Subnet

Question: Consider the network `192.168.1.128/26`.

1. Determine the broadcast address for the subnet.
2. Specify the first and the last usable IP addresses in this subnet.

Hint: The subnet contains 64 addresses; the network address and the broadcast address are not usable.

Solution:

A /26 subnet means that 26 bits are used for the network portion and the remaining $32 - 26 = 6$ bits are used for the host portion. This gives a total of $2^6 = 64$ addresses per subnet.

Step 1: Determine the Broadcast Address

The given network address is `192.168.1.128`. In a block of 64 addresses, the addresses range from 0 to 63 relative to the network address. Since the network address itself is counted as the first address, the broadcast address is found by adding $64 - 1 = 63$ to the network address.

$$\texttt{192.168.1.128} + 63 = \texttt{192.168.1.191}$$

Thus, the broadcast address is **192.168.1.191**.

Step 2: Determine the Usable Address Range

The usable IP addresses are those that can be assigned to hosts. They exclude the network address (first address) and the broadcast address (last address).
- The first usable IP address is one more than the network address: `192.168.1.128` + 1 = **192.168.1.129**.
- The last usable IP address is one less than the broadcast address: `192.168.1.191` − 1 = **192.168.1.190**.

Summary:

- Broadcast address: **192.168.1.191**
- First usable IP address: **192.168.1.129**
- Last usable IP address: **192.168.1.190**

Exercise 4: Subdividing a Network into Subnets

Question: You have the network `192.168.1.0/24`.

1. Divide it into 4 equal subnets.

2. For each subnet, provide the CIDR notation, the network address, and the broadcast address.

Hint: Dividing a /24 network into 4 subnets means borrowing 2 bits for subnetting. Each subnet will contain 64 addresses.

Solution:

A /24 network has 256 addresses (from 0 to 255). Dividing it into 4 equal subnets requires borrowing 2 bits from the host portion, resulting in a new subnet mask of /26 (since $24 + 2 = 26$). Each /26 subnet contains $2^6 = 64$ addresses.

Step 1: Calculate the Subnet Size
Since each subnet has 64 addresses, the subnets will be:

Subnet 1: 192.168.1.0 to 192.168.1.63Subnet 2: 192.168.1.64 to 192.168.1.127Subnet 3: 192.168.1.128 t

Step 2: Identify the Network and Broadcast Addresses for Each Subnet
- **Subnet 1:**
- **Network Address:** `192.168.1.0`
- **Broadcast Address:** `192.168.1.63`
- **CIDR Notation:** `/26`

- **Subnet 2:**
- **Network Address:** `192.168.1.64`
- **Broadcast Address:** `192.168.1.127`
- **CIDR Notation:** `/26`

- **Subnet 3:**
- **Network Address:** `192.168.1.128`
- **Broadcast Address:** `192.168.1.191`
- **CIDR Notation:** `/26`

- **Subnet 4:**
- **Network Address:** `192.168.1.192`
- **Broadcast Address:** `192.168.1.255`
- **CIDR Notation:** `/26`

Summary:
Dividing the network `192.168.1.0/24` into 4 equal subnets yields four /26 subnets with the following details:

- **Subnet 1:** Network Address: `192.168.1.0`, Broadcast Address: `192.168.1.63`, CIDR: `/26`.

- **Subnet 2:** Network Address: `192.168.1.64`, Broadcast Address: `192.168.1.127`, CIDR: `/26`.

- **Subnet 3:** Network Address: `192.168.1.128`, Broadcast Address: `192.168.1.191`, CIDR: `/26`.
- **Subnet 4:** Network Address: `192.168.1.192`, Broadcast Address: `192.168.1.255`, CIDR: `/26`.

Exercise 5: Conversion Between Decimal Mask and Binary

Question: Convert the subnet mask `255.255.254.0` into its binary representation. Then, determine the corresponding CIDR notation by counting the number of 1 bits.
Hint: Convert each octet to binary and sum up the bits set to 1 to get the prefix length.

Solution:
First, convert each octet of the subnet mask to its binary form:

- `255` in binary is `11111111`.
- `255` in binary is `11111111`.
- `254` in binary is `11111110`.
- `0` in binary is `00000000`.

Thus, the full binary representation of `255.255.254.0` is: 11111111.11111111.11111110.00000000
Next, count the number of 1-bits:

- First octet: 8 bits.
- Second octet: 8 bits.
- Third octet: 7 bits (since `11111110` has one 0).
- Fourth octet: 0 bits.

Total number of 1 bits $= 8 + 8 + 7 + 0 = 23$.
Therefore, the corresponding CIDR notation is /**23**.

Exercise 6: Determining if Two IP Addresses Belong to the Same Subnet

Question: Given the subnet mask `255.255.255.0` and the two IP addresses `192.168.1.45` and `192.168.1.200`, determine if they belong to the same subnet.
Hint: Compare the network portions of both IP addresses using the subnet mask.

Solution:
A subnet mask of `255.255.255.0` corresponds to a binary mask of `11111111.11111111.11111111.00000000`, meaning the first 24 bits are used to identify the network portion.

Step 1: Extract the Network Portion
For both IP addresses, apply the subnet mask to obtain their network address:

- For `192.168.1.45`: The network portion is `192.168.1` (first three octets).
- For `192.168.1.200`: The network portion is also `192.168.1`.

Step 2: Compare the Network Portions
Since both IP addresses have the identical network portion (`192.168.1`), they belong to the same subnet.

Conclusion: The IP addresses `192.168.1.45` and `192.168.1.200` are in the same subnet.

Exercise 7: Variable Length Subnet Masking (VLSM)

Question: You have the network `192.168.10.0/24` and need to create three subnets with the following requirements:

1. The first subnet must support at least 100 hosts.
2. The second subnet must support at least 50 hosts.
3. The third subnet must support at least 25 hosts.

Design the subnets by choosing appropriate subnet masks and determining the network addresses for each.
Hint: Calculate the number of host bits required for each subnet, then allocate subnets accordingly while keeping the networks contiguous if possible.

Solution:
We begin with the network `192.168.10.0/24`, which provides a total of 256 IP addresses (ranging from 0 to 255). We will design subnets in order of descending host requirements to optimize address allocation.

Step 1: Determine the Required Subnet Sizes

- **First Subnet (at least 100 hosts):**
  To have at least 100 usable addresses, we need $2^n - 2 \geq 100$. With $n = 7$, $2^7 - 2 = 128 - 2 = 126$ usable addresses, which satisfies the requirement.
  This means we use 7 bits for hosts, giving a subnet mask of $32 - 7 = 25$, i.e., `/25`.

- **Second Subnet (at least 50 hosts):**
  For at least 50 usable addresses, $2^n - 2 \geq 50$ is needed. With $n = 6$, $2^6 - 2 = 64 - 2 = 62$ usable addresses, which is sufficient.
  This requires a subnet mask of $32 - 6 = 26$, i.e., `/26`.

- **Third Subnet (at least 25 hosts):**
  For at least 25 usable addresses, $2^n - 2 \geq 25$ is needed. With $n = 5$, $2^5 - 2 = 32 - 2 = 30$ usable addresses, meeting the requirement.
  This gives a subnet mask of $32 - 5 = 27$, i.e., `/27`.

Step 2: Allocate the Subnets Contiguously

1. **First Subnet:**
   - **Network Address:** `192.168.10.0/25`
   - **Address Range:** This subnet covers 128 addresses, from `192.168.10.0` to `192.168.10.127` (with 126 usable addresses).
2. **Second Subnet:**
   - **Network Address:** The next available address after the first subnet is `192.168.10.128`. With a `/26` mask, this subnet covers 64 addresses, from `192.168.10.128` to `192.168.10.191` (with 62 usable addresses).
3. **Third Subnet:**
   - **Network Address:** The next available address after the second subnet is `192.168.10.192`. With a `/27` mask, this subnet covers 32 addresses, from `192.168.10.192` to `192.168.10.223` (with 30 usable addresses).

Summary:

- **First Subnet:** `192.168.10.0/25`
  - Address Range: `192.168.10.0` to `192.168.10.127` (126 usable addresses)
- **Second Subnet:** `192.168.10.128/26`
  - Address Range: `192.168.10.128` to `192.168.10.191` (62 usable addresses)
- **Third Subnet:** `192.168.10.192/27`
  - Address Range: `192.168.10.192` to `192.168.10.223` (30 usable addresses)

Note: The remaining addresses from `192.168.10.224` to `192.168.10.255` are unused in this allocation.

Exercise 8: Converting a Binary Subnet Mask to Decimal and Identifying the Network Class

Question: Given the binary subnet mask `11111111.11111111.11110000.00000000`, convert it to decimal format. Then, determine the default network class associated with the IP address `172.16.0.0` using the standard classful addressing scheme.
Hint: Count the 1 bits for CIDR, convert each octet to its decimal equivalent, and recall that `172.16.0.0` belongs to Class B.

Solution:

Step 1: Convert the Binary Subnet Mask to Decimal

The binary subnet mask is:

$$11111111.11111111.11110000.00000000$$

Convert each octet:

- `11111111` converts to 255.
- `11111111` converts to 255.
- `11110000` converts to 240 (i.e., $128 + 64 + 32 + 16 = 240$).
- `00000000` converts to 0.

Thus, the decimal representation is **255.255.240.0**.

Step 2: Determine the CIDR Notation

Count the number of 1 bits in the binary mask:

- First octet: 8 ones.
- Second octet: 8 ones.
- Third octet: 4 ones.
- Fourth octet: 0 ones.

Total $= 8 + 8 + 4 = 20$. Therefore, the CIDR notation is /**20**.

Step 3: Identify the Network Class for `172.16.0.0`

The IP address `172.16.0.0` falls within the range of 128.0.0.0 to 191.255.255.255, which is designated as Class B in the standard classful addressing scheme.

Conclusion:

The binary subnet mask `11111111.11111111.11110000.00000000` converts to **255.255.240.0** (CIDR: /**20**), and the IP address `172.16.0.0` belongs to **Class B**.

Exercise 9: Calculating the Wildcard Mask

Question: For the subnet mask `255.255.255.192`, calculate the corresponding wildcard mask.
Hint: The wildcard mask is the inverse of the subnet mask (i.e., subtract each octet from 255).

Solution:
A **wildcard mask** is used primarily in routing protocols and access control lists (ACLs) to specify a range of IP addresses. It is the inverse of the subnet mask, meaning that you subtract each octet of the subnet mask from 255. The wildcard mask indicates which bits in the IP address should be ignored (wildcarded) during matching.

Step 1: Calculate the Wildcard Mask
Given the subnet mask `255.255.255.192`:

- First octet: $255 - 255 = 0$
- Second octet: $255 - 255 = 0$
- Third octet: $255 - 255 = 0$
- Fourth octet: $255 - 192 = 63$

Thus, the corresponding wildcard mask is **0.0.0.63**.

Explanation:
The wildcard mask **0.0.0.63** means that in the fourth octet, the last 6 bits are wildcarded (ignored) when matching IP addresses, while the first 2 bits must match exactly. This is because 192 in binary is `11000000` and its inverse, 63, is `00111111`.

Exercise 10: IP Address Summarization (Supernetting)

Question: You have been assigned the following networks:

- `10.10.1.0/24`
- `10.10.2.0/24`
- `10.10.3.0/24`

Find the smallest supernet that can contain all three networks.
Hint: Identify the common bits in the network addresses and determine the appropriate summarized prefix.

Solution:
To summarize the networks, we need to find the common prefix bits among the three network addresses.

Step 1: Write the Relevant Octets in Binary
Since the first two octets are the same for all three networks (10.10), we focus on the third octet.

- `10.10.1.0` → Third octet: 1 = `00000001`

- `10.10.2.0` → Third octet: 2 = `00000010`

- `10.10.3.0` → Third octet: 3 = `00000011`

Step 2: Determine the Common Bits in the Third Octet
Comparing the binary representations of the third octet:

- For 1: `00000001`

- For 2: `00000010`

- For 3: `00000011`

The first six bits are identical (`000000`) for all three values. The 7th bit differs (0 in 1 vs. 1 in 2 and 3).

Step 3: Calculate the Total Number of Common Bits
The first two octets contribute 16 bits. The third octet contributes 6 common bits. Thus, the total common bits $= 16 + 6 = 22$.
This gives us a summarized prefix of /**22**.

Step 4: Determine the Summarized Network Address
With a /22 mask, the summarized network will have the following characteristics:

- It covers 22 bits for the network and 10 bits for hosts.
- The summarized network address will have the common 22 bits and zeros in the remaining bits.

Since the first two octets are 10.10 and the first 6 bits of the third octet are `000000`, the summarized network address is **10.10.0.0/22**.

Verification:
A /22 network provides $2^{10} = 1024$ addresses, covering the range from:

$$10.10.0.0 \text{ to } 10.10.3.255$$

This range indeed includes:

- `10.10.1.0/24`
- `10.10.2.0/24`
- `10.10.3.0/24`

Thus, the smallest supernet that can contain all three networks is **10.10.0.0/22**.

Exercise 11: Summarizing Four /24 Networks

Question: Summarize the following networks into a single supernet:

- `192.168.4.0/24`
- `192.168.5.0/24`
- `192.168.6.0/24`
- `192.168.7.0/24`

Hint: Convert the third octet of each network address to binary, identify the common bits, and determine the summarized prefix.

Solution:

Each network is a /24, meaning the first 24 bits are fixed. The first two octets are the same (`192.168`). Focus on the third octet:

- `192.168.4.0` $\rightarrow$ Third octet: $4 =$ `00000100`
- `192.168.5.0` $\rightarrow$ Third octet: $5 =$ `00000101`
- `192.168.6.0` $\rightarrow$ Third octet: $6 =$ `00000110`
- `192.168.7.0` $\rightarrow$ Third octet: $7 =$ `00000111`

The common bits in the third octet are the first 6 bits (`000001`).
Total common bits $= 16$ (first two octets) $+ 6 = 22$.
Thus, the summarized prefix is /**22**.
The summarized (supernet) network is **192.168.4.0/22**, which covers the range from `192.168.4.0` to `192.168.7.255`.

Exercise 12: Summarizing Two /24 Networks

Question: Summarize the following two networks into one supernet:

- `10.0.0.0/24`
- `10.0.1.0/24`

Hint: Examine the difference in the third octet in binary and determine how many bits are common.

Solution:

Both networks are /24. The first two octets (`10.0`) are identical. Look at the third octet:

- For `10.0.0.0`: Third octet = 0 = `00000000`
- For `10.0.1.0`: Third octet = 1 = `00000001`

In the third octet, the first 7 bits are identical (i.e., `0000000`).
Total common bits = 16 (from the first two octets) + 7 = 23, so the summarized prefix is /**23**.
The summarized network is **10.0.0.0/23**, which spans from `10.0.0.0` to `10.0.1.255`.

Exercise 13: Summarizing Eight /24 Networks

Question: Summarize the following contiguous networks into a single supernet:

- `172.16.0.0/24`
- `172.16.1.0/24`
- `172.16.2.0/24`
- `172.16.3.0/24`
- `172.16.4.0/24`
- `172.16.5.0/24`
- `172.16.6.0/24`
- `172.16.7.0/24`

Hint: Eight contiguous /24 networks can be summarized by determining the number of bits needed to cover them.

Solution:

Each network is a /24, meaning each covers 256 addresses. Since there are 8 contiguous networks, the total number of addresses is $8 \times 256 = 2048$.

To determine the summarized prefix, note that a /21 network provides $2^{32-21} = 2^{11} = 2048$ addresses.

Thus, the summarized supernet is **172.16.0.0/21**, which covers the IP range from `172.16.0.0` to `172.16.7.255`.

Exercise 14: Summarizing Four /25 Networks

Question: Summarize the following networks into a single supernet:

- `192.168.20.0/25`
- `192.168.20.128/25`
- `192.168.21.0/25`
- `192.168.21.128/25`

Hint: These four /25 networks form two contiguous /24 networks. Summarize the two /24 networks into one supernet.

Solution:

The first two networks, `192.168.20.0/25` and `192.168.20.128/25`, combine to form `192.168.20.0/24`. Similarly, `192.168.21.0/25` and `192.168.21.128/25` form `192.168.21.0/24`.

These two /24 networks are contiguous, covering the address range from `192.168.20.0` to `192.168.21.255`.

A contiguous block of two /24 networks is summarized by a /23 network.

Thus, the smallest supernet that covers all four /25 networks is **192.168.20.0/23**.

Exercise 15: Summarizing Two /23 Networks

Question: Summarize the following networks into one supernet:

- `10.10.0.0/23`
- `10.10.2.0/23`

Hint: Verify that the networks are contiguous and then determine the appropriate summarized prefix.

Solution:

A /23 network contains 512 addresses.

- `10.10.0.0/23` covers addresses from `10.10.0.0` to `10.10.1.255`.
- `10.10.2.0/23` covers addresses from `10.10.2.0` to `10.10.3.255`.

These two blocks are contiguous since the first block ends at `10.10.1.255` and the next begins at `10.10.2.0`.
Combined, they span 1024 addresses, which corresponds to a /22 network (since $2^{32-22} = 2^{10} = 1024$).
The summarized supernet is **10.10.0.0/22**, covering the range from `10.10.0.0` to `10.10.3.255`.

Exercise 16: Summarizing Five Contiguous /24 Networks

Question: Summarize the following networks into a single supernet:

- `192.168.100.0/24`
- `192.168.101.0/24`
- `192.168.102.0/24`
- `192.168.103.0/24`
- `192.168.104.0/24`

Hint: Determine the overall range of addresses and then find the smallest supernet that can include them all.

Solution:

These five /24 networks span from `192.168.100.0` to `192.168.104.255`.
The total number of addresses needed is $5 \times 256 = 1280$. The next power of two that can cover 1280 addresses is 2048, corresponding to a block size of a /21 network ($2^{11} = 2048$).
However, a /21 network must be aligned on a boundary where the third octet is a multiple of 8. For example, a /21 block starting at `192.168.96.0` spans from `192.168.96.0` to `192.168.103.255`, which does not fully cover the range since `192.168.104.0/24` is outside this block.
The next /21 block would start at `192.168.104.0` and cover `192.168.104.0` to `192.168.111.255`, but this does not include `192.168.100.0/24`.
Since a single /21 block cannot be properly aligned to cover the range from `192.168.100.0` to `192.168.104.255`, we must choose a larger block. A /20 network provides $2^{12} = 4096$ addresses and is aligned on boundaries where the third octet is a multiple of 16.
The /20 block starting at `192.168.96.0` spans from `192.168.96.0` to `192.168.111.255` and fully covers the range of our five networks.
Thus, the smallest supernet that can cover all five networks is **192.168.96.0/20**.

Exercise 17: TCP Throughput Calculation

Question: A TCP connection has a round-trip time (RTT) of 100 ms and a TCP window size of 64 KB. Calculate the maximum achievable throughput in Mbps (assuming no packet loss).

Hint: Use the formula:

$$\text{Throughput} = \frac{\text{Window Size (in bits)}}{\text{RTT}}$$

Solution:

Step 1: Convert the TCP Window Size to Bits.
The window size is given as 64 KB. Since 1 KB = 1024 bytes, we have:

$$64 \times 1024 = 65\,536 \text{ bytes}$$

Each byte consists of 8 bits, so the total number of bits is:

$$65\,536 \times 8 = 524\,288 \text{ bits}$$

Step 2: Convert the RTT to Seconds.
The RTT is provided as 100 ms. Recall that 1 ms = 0.001 seconds, hence:

$$100 \text{ ms} = 0.1 \text{ s}$$

Step 3: Calculate the Throughput.
Substitute the values into the throughput formula:

$$\text{Throughput} = \frac{524\,288 \text{ bits}}{0.1 \text{ s}} = 5\,242\,880 \text{ bits/s}$$

Step 4: Convert the Throughput to Mbps.
1 Mbps = 10^6 bits/s, so:

$$\frac{5\,242\,880 \text{ bits/s}}{10^6} \approx 5.24 \text{ Mbps}$$

Explanation:
This calculation shows the maximum data rate achievable given the window size and RTT. Essentially, the window size limits the amount of data that can be in transit at any given time, and the RTT determines how quickly acknowledgments are received. The throughput is the window size divided by the RTT, giving an idea of the data transfer rate under ideal conditions.

Exercise 18: Protocol Overhead Calculation

Question: A network packet consists of a 40-byte header and a 1000-byte payload. Calculate the overhead percentage of the header in the packet.

Hint: Overhead percentage is calculated as:

$$\frac{\text{Header Size}}{\text{Header Size} + \text{Payload Size}} \times 100\%$$

Solution:
Step 1: Determine the Total Packet Size.
The header is 40 bytes and the payload is 1000 bytes, so:

$$\text{Total Packet Size} = 40 + 1000 = 1040 \text{ bytes}$$

Step 2: Calculate the Overhead Percentage.
Using the formula:

$$\text{Overhead Percentage} = \frac{40}{1040} \times 100\%$$

Perform the calculation:

$$\frac{40}{1040} \approx 0.03846 \quad \text{and} \quad 0.03846 \times 100\% \approx 3.85\%$$

Explanation:
This value represents the fraction of the packet that is not used for actual data but is necessary for routing, error checking, and other control purposes. A lower overhead percentage means a higher efficiency in data transfer.

Exercise 19: End-to-End Delay Calculation

Question: A packet of 1500 bytes is transmitted over a link with a bandwidth of 100 Mbps and a propagation delay of 20 ms. Calculate the total end-to-end delay for this packet (i.e., the sum of the transmission delay and the propagation delay).

Hint: The transmission delay is given by:

$$\text{Transmission Delay} = \frac{\text{Packet Size (in bits)}}{\text{Bandwidth}}$$

Solution:
Step 1: Convert the Packet Size to Bits.
The packet is 1500 bytes, so in bits:

$$1500 \times 8 = 12\,000 \text{ bits}$$

Step 2: Calculate the Transmission Delay.
Given the bandwidth is 100 Mbps:

$$\text{Transmission Delay} = \frac{12\,000 \text{ bits}}{100 \times 10^6 \text{ bits/s}} = 0.00012 \text{ s} = 0.12 \text{ ms}$$

Step 3: Add the Propagation Delay.
The propagation delay is provided as 20 ms. Hence, the total delay is:

$$0.12 \text{ ms} + 20 \text{ ms} = \mathbf{20.12 \text{ ms}}$$

Explanation:
Transmission delay is the time required to push all the packet's bits onto the link, while propagation delay is the time taken for the signal to travel from the sender to the receiver. The total end-to-end delay is the sum of these two delays, giving an overall measure of the latency in the network.

Exercise 20: TCP Connection Setup and File Transfer Delay

Question: A client needs to download a 1 MB file from a server. Assume the following:

- The TCP connection requires 1 RTT for the three-way handshake.
- The RTT is 50 ms.
- The link bandwidth is 10 Mbps.

Calculate the total time required (in ms) to establish the TCP connection and download the file (ignore congestion and processing delays).

Hint: Total delay = Handshake delay + File transmission delay.
Solution:
Step 1: Calculate the TCP Handshake Delay.
The handshake requires one RTT:

$$\text{Handshake Delay} = 50 \text{ ms}$$

Step 2: Calculate the File Transmission Delay.
The file size is 1 MB. Convert 1 MB to bytes:

$$1\ \text{MB} = 1 \times 1024 \times 1024 = 1\,048\,576\ \text{bytes}$$

Now convert to bits:

$$1\,048\,576 \times 8 = 8\,388\,608\ \text{bits}$$

Given the bandwidth is 10 Mbps (or 10×10^6 bits/s), the transmission delay is:

$$\text{Transmission Delay} = \frac{8\,388\,608\ \text{bits}}{10 \times 10^6\ \text{bits/s}} \approx 0.83886\ \text{s} = 838.86\ \text{ms}$$

Step 3: Calculate the Total Delay.
Total delay = Handshake delay + Transmission delay:

$$50\ \text{ms} + 838.86\ \text{ms} \approx \mathbf{888.86\ ms}$$

Explanation:
This exercise demonstrates that the initial handshake delay can be significant for small files, but for larger files, the transmission delay dominates. The overall delay reflects both the time taken to establish the connection and to transmit the data.

Exercise 21: HTTP Transaction Delay with DNS Lookup

Question: A client initiates an HTTP request to fetch a web page. The following delays are observed:

- DNS lookup delay: 40 ms
- TCP handshake delay: 50 ms
- HTTP request/response delay: 150 ms
- Page content download delay: 300 ms

Calculate the total time (in ms) from the initial request until the page is fully loaded.

Hint: Sum all the individual delays.

Solution:

Step 1: List All Delays.

- DNS lookup delay: 40 ms
- TCP handshake delay: 50 ms
- HTTP request/response delay: 150 ms
- Page content download delay: 300 ms

Step 2: Sum the Delays.

$$40 \text{ ms} + 50 \text{ ms} + 150 \text{ ms} + 300 \text{ ms} = 540 \text{ ms}$$

Explanation:
Each component of the delay contributes to the overall time needed to load a web page. The DNS lookup resolves the domain name to an IP address, the TCP handshake establishes the connection, the HTTP request/response carries the initial exchange, and finally, the page content is downloaded. The sum of these delays gives a complete picture of the time a user experiences when accessing a web page.