Lab Answer Key: Module 5: Implementing IPv4

Lab: Implementing IPv4

Exercise 1: Identifying Appropriate Subnets

Task 1: Calculate the bits required to support the hosts on each subnet

1. How many bits are required to support 100 hosts on the client subnet?
   
   ✔️ **Answer:** Seven bits are required to support 100 hosts on the client subnet 
   
   \(2^7 - 2 = 126, \ 2^6 - 2 = 62\).

2. How many bits are required to support 10 hosts on the server subnet?
   
   ✔️ **Answer:** Four bits are required to support 10 hosts on the server subnet 
   
   \(2^4 - 2 = 14, \ 2^3 - 2 = 6\).

3. How many bits are required to support 40 hosts on the future expansion subnet?
   
   ✔️ **Answer:** Six bits are required to support 40 hosts on the future expansion subnet 
   
   \(2^6 - 2 = 62, \ 2^5 - 2 = 30\).

4. If all subnets are the same size, can they be accommodated?
   
   ✔️ **Answer:** No. If all subnets are the same size, then all subnets must use 7 bits to support 126 hosts. Only a single class C–sized address with 254 hosts has been allocated. Three subnets of 126 hosts would not fit.

5. Which feature allows a single network to be divided into subnets of varying sizes?
   
   ✔️ **Answer:** Variable length subnet masking allows you to define different subnet masks when subnetting. Therefore, variable length subnet masking allows you to have subnets of varying sizes.
6. How many host bits will you use for each subnet? Use the simplest allocation possible, which is one large subnet and two equal-sized smaller subnets.

Answer: The client subnet is 7 host bits. This allocation can accommodate up to 126 hosts and uses half of the allocated address pool.

The server and future expansion subnets are 6-host bits. This can accommodate up to 62 hosts on each subnet and uses the other half of the address pool.

Task 2: Calculate subnet masks and network IDs

1. Given the number of host bits allocated, what is the subnet mask that you will use for the client subnet? Calculate the subnet mask in binary and decimal.

   • The client subnet is using 7 bits for the host ID. Therefore, you can use 25 bits for the subnet mask.

<table>
<thead>
<tr>
<th>Binary</th>
<th>Decimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>11111111.11111111.11111111.10000000</td>
<td>255.255.255.128</td>
</tr>
</tbody>
</table>

2. Given the number of host bits allocated, what is the subnet mask that you can use for the server subnet? Calculate the subnet mask in binary and decimal.

   • The server subnet is using 6 bits for the host ID. Therefore, you can use 26 bits for the subnet mask.

<table>
<thead>
<tr>
<th>Binary</th>
<th>Decimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>11111111.11111111.11111111.11000000</td>
<td>255.255.255.192</td>
</tr>
</tbody>
</table>

3. Given the number of host bits allocated, what is the subnet mask that you can use for the future expansion subnet? Calculate the subnet mask in binary and decimal.

   • The future expansion subnet is using 6 bits for the host ID. Therefore, you can
use 26 bits for the subnet mask.

<table>
<thead>
<tr>
<th>Binary</th>
<th>Decimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>11111111.11111111.11111111.11000000</td>
<td>255.255.255.192</td>
</tr>
</tbody>
</table>

4. For the client subnet, define the network ID, first available host, last available host, and broadcast address. Assume that the client subnet is the first subnet allocated from the available address pool. Calculate the binary and decimal versions of each address.

In the following table, the bits in bold are part of the network ID.

<table>
<thead>
<tr>
<th>Description</th>
<th>Binary</th>
<th>Decimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network ID</td>
<td>11000000.10101000.1100010.0000000</td>
<td>192.168.98.0</td>
</tr>
<tr>
<td>First host</td>
<td>11000000.10101000.1100010.0000001</td>
<td>192.168.98.1</td>
</tr>
<tr>
<td>Last host</td>
<td>11000000.10101000.1100010.0111110</td>
<td>192.168.98.126</td>
</tr>
<tr>
<td>Broadcast</td>
<td>11000000.10101000.1100010.0111111</td>
<td>192.168.98.127</td>
</tr>
</tbody>
</table>

5. For the server subnet, define the network ID, first available host, last available host, and broadcast address. Assume that the server subnet is the second subnet allocated from the available address pool. Calculate the binary and decimal versions of each address.

In the following table, the bits in bold are part of the network ID.

<table>
<thead>
<tr>
<th>Description</th>
<th>Binary</th>
<th>Decimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network ID</td>
<td>11000000.10101000.1100010.1000000</td>
<td>192.168.98.128</td>
</tr>
<tr>
<td>First host</td>
<td>11000000.10101000.1100010.1000001</td>
<td>192.168.98.129</td>
</tr>
<tr>
<td>Last host</td>
<td>11000000.10101000.1100010.1011110</td>
<td>192.168.98.190</td>
</tr>
<tr>
<td>Broadcast</td>
<td>11000000.10101000.1100010.1011111</td>
<td>192.168.98.191</td>
</tr>
</tbody>
</table>

6. For the future allocation subnet, define the network ID, first available host, last available host, and broadcast address. Assume that the future allocation subnet
is the third subnet allocated from the available address pool. Calculate the binary and decimal versions of each address.

In the following table, the bits in bold are part of the network ID.

<table>
<thead>
<tr>
<th>Description</th>
<th>Binary</th>
<th>Decimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network ID</td>
<td>11000000.10101000.1100010.11000000</td>
<td>192.168.98.192</td>
</tr>
<tr>
<td>First host</td>
<td>11000000.10101000.1100010.11000001</td>
<td>192.168.98.193</td>
</tr>
<tr>
<td>Last host</td>
<td>11000000.10101000.1100010.11111110</td>
<td>192.168.98.254</td>
</tr>
<tr>
<td>Broadcast</td>
<td>11000000.10101000.1100010.11111111</td>
<td>192.168.98.255</td>
</tr>
</tbody>
</table>

**Results:** After completing this exercise, you should have identified the subnets required to meet the requirements of the lab scenario.

**Exercise 2: Troubleshooting IPv4**

**Task 1: Prepare for troubleshooting**

1. On LON-SVR2, on the taskbar, click the **Windows PowerShell** icon.
2. At the Windows PowerShell prompt, type the following cmdlet, and then press Enter:
   ```
   Test-NetConnection LON-DC1
   ```
3. Verify that you receive a reply that contains **PingSucceeded:True** from LON-DC1.
4. Open a **File Explorer** window, and browse to ```\LON-DC1\E$\Labfiles\Mod05```.
5. Right-click **Break2.ps1**, and then click **Run with PowerShell**.
Note: This script creates the problem that you will troubleshoot and repair in the next task.


Task 2: Troubleshoot IPv4 connectivity between LON-SVR2 and LON-DC1

1. On LON-SVR2, at the Windows PowerShell prompt, type the following, and then press Enter:

   ```powershell
   Test-NetConnection LON-DC1
   ```

2. Verify that you receive a reply that contains **PingSucceeded:False** from LON-DC1.

3. At the Windows PowerShell Prompt, type the following, and then press Enter:

   ```powershell
   Test-NetConnection -TraceRoute LON-DC1
   ```

   Notice that the host is unable to find the default gateway, and that the following warning message appears: “**Name resolution of lon-dc1 failed – Status: HostNotFound.**”

4. At the Windows PowerShell Prompt, type the following, and then press Enter:

   ```powershell
   Get-NetRoute
   ```

   Notice that the default route and the default gateway information is missing in the routing table.
Note: You should not be able to locate DestinationPrefix 0.0.0.0/0 and NextHop 10.10.0.1.

5. At the Windows PowerShell Prompt, type the following, and then press Enter:

```
Test-NetConnection 10.10.0.1
```

6. Notice that the default gateway is responding by verifying that you receive a reply that contains PingSucceeded:True from 10.10.0.1.

7. At the Windows PowerShell Prompt, type the following, and then press Enter:

```
New-NetRoute –InterfaceAlias “Ethernet” –DestinationPrefix 0.0.0.0/0 –NextHop 10.10.0.1
```

Note: The New-NetRoute cmdlet will create the default route and the default gateway information that was missing.

8. At the Windows PowerShell Prompt, type the following, and then press Enter:

```
Get-NetRoute
```

9. Notice that the default route and the default gateway information is present in the routing table by locating DestinationPrefix 0.0.0.0/0 and NextHop 10.10.0.1.

10. At the Windows PowerShell prompt, type the following, and then press Enter:

```
Test-NetConnection LON-DC1
```
11. Verify that you receive a reply that contains `PingSucceeded:True` from LON-DC1.

**Results:** After completing this lab, you should have resolved an IPv4 connectivity problem.

**Prepare for the next module**

After you finish the lab, revert the virtual machines back to their initial state. To do this, complete the following steps.

1. On the host computer, start **Hyper-V Manager**.
2. In the **Virtual Machines** list, right-click **20410C-LON-DC1**, and then click **Revert**.
3. In the **Revert Virtual Machine** dialog box, click **Revert**.
4. Repeat steps 2 and 3 for **20410C-LON-RTR** and **20410C-LON-SVR2**.