OBJECTIVES

While IPv4 packets are most usually addressed to exactly one recipient, there is sometimes the need to send a packet to all nodes on a network. When a computer sends an IP packet to one recipient, this is termed a *unicast packet*, and the recipient's IP address is termed a *unicast address*. When a computer sends an IP packet to all nodes on the network, this is termed a *broadcast packet*, and it is addressed to the network's *broadcast address*. One objective of this lab is to learn to calculate and work with IPv4 subnet directed broadcast addresses, perhaps the most significant type of broadcast address.

Organizations oftentimes subdivide their networks into smaller networks, and these smaller networks are termed *subnets*. Another objective of this lab is to learn how to perform important calculations specific to subnets.

PREREQUISITES

Before attempting this lab, it is best to read the textbook and lecture material for the week. While this lab shows you how to complete specific steps, the lab does not explain in full the framework and theory, with the assumption that you will first read the lecture and textbook material.

LAB SUBMISSION

Use the submission template provided in the assignment inbox to perform the steps requested by this lab. Return to the assignment inbox to submit your lab.

OVERVIEW

To understand what a subnet directed broadcast is, we first need to understand what a subnet is. Simply put, a subnet is an IPv4 network whose network identifier consists of a *network prefix* and a *subnet identifier*. The network prefix is used to indicate that multiple subnets are on the same overall network, and the subnet identifier is used to distinguish one subnet from another subnet on that same overall network. The conceptual distinction between a subnet and a network is based upon perspective. Network professionals, academics, and texts often associate the term “network” with a single overarching entity that belongs to an organization and represents the organization’s entire network, and they often associate the term “subnet” to refer to one specific sub-network within the organization’s overall network. However, this distinction is somewhat arbitrary. There is nothing to stop an organization from subdividing its overall network into multiple networks, and then assigning subnets to each of these networks. And there is also nothing to stop an Internet Service Provider (ISP) from doing the same across multiple organizations with any public IP addresses it has in its possession.

It is very important to understand that *network adapters, computers, and routers have no knowledge of a network prefix and a subnet identifier*, *and are only aware of a network identifier and host identifier*. To state this another way, network adapters, computers, and routers have nothing in their configuration to further subdivide the network identifier, and they use the network identifier in full as an atomic number; they do not have a finer-grained distinction beyond that of a network. So now you understand that any network whose identifier can be separated into a network prefix and a subnet identifier is a subnet, and conceptually we view a subnet as a sub-network that exists within the context of an overarching network.

Now that we understand what a subnet is, the phrase *subnet directed broadcast* makes a little more sense. It is a broadcast intended for all network adapters on a specific subnet. Almost all IPv4 communications are intended for a single recipient, but some protocols require the use of these broadcasts. For example, the Dynamic Host Configuration Protocol (DHCP) is used to dynamically assign an IPv4 address and a subnet mask to a network adapter. Before the network adapter has been assigned its IPv4 address, it is not possible for the DHCP server to send a unicast packet to the network adapter, so it must broadcast the packet to all nodes on the subnet, relying on the fact that the network adapter in need of the IPv4 address will receive the message.

STEPS

1. A subnet directed broadcast address is always the highest address in the subnet. To compute the highest address in a subnet, we extract the network identifier from the IPv4 address, then append 1 bits to it until we fill out the full 32-bits. In other words, a broadcast address has a network identifier which uniquely identifies the appropriate subnet, and a host identifier consisting of all 1 bits to indicate that the address is not intended for a single recipient, but for all recipients on that subnet.

   Let's take an example, 96.45.82.5/20. First, we need to calculate the network address. We know that /20 full spans the first two octets and does not span the last octet at all, so the network identifier is 96.45.X.0. We then convert the value in the third octet, 82 in decimal, to binary, which comes out to be 01010010. The /20 means that 4 bits are spanned in this third octet, so extracting the first 4 bits and zeroing out the remainder leaves us with 01010000, which is 80 in decimal. So we know that the network address is 96.45.80.0.

   To calculate the broadcast address, we now need to turn all host bits to 1. Doing so is straightforward. If the network identifier spans 20 bits, how many bits does the host identifier span? The answer is 12, calculated as, 32 - 20 = 12. The difference is that we start counting the bits for a network identifier starting from the left, and we start counting bits for the host identifier from the right.

   We can use shortcuts when calculating the broadcast address similar to the shortcuts we use when calculating the network address. Namely, if the host identifier fully spans an octet, it will consist of all 1s, and the octet is represented with 255 in binary. If the host identifier does not span an octet at all, then it has no effect on that octet. The only octet that needs a calculation is one that is partially spanned by the network identifier and by the host identifier.

   Thus far we have calculated the network address for 96.45.82.5/20, which is 96.45.80.0. If we want to now calculate that network's broadcast address, we see that the rightmost octet is fully spanned by the host identifier, and the third octet is partially spanned, so we end up with 96.45.Y.255. Now we just need to calculate Y. Recall that 80 is 01010000 in binary, and that 0101 is the portion spanned by the network identifier. Thus, we simply turn the last 4 bits in the octet to 1s, with a result of 01011111, which is 95 in decimal. Thus, the broadcast address is 96.45.95.255, and this is the highest possible address in the 96.45.80.0 network.

   Did you catch all of this? In summary, we calculate the network address, then set the host identifier bits to all 1s. In doing so, we use the shortcut that any octet fully spanned by the host identifier is 255 in decimal. It is really that simple!

2. Now you give it a try by calculating the broadcast address for the following CIDR entries using the methodology listed in step 1. Represent your answers in dotted decimal notation, and make sure to show your work, using a tool if needed to convert from decimal to binary and vice versa.

   184.168.221.16/18
   64.152.0.21/25

3. Another important aspect of a network address is the number of hosts that can exist on that network. At first glance, you might think that all we need to do is raise 2 to the power of the number of bits available in the host identifier, and you would be close. This is indeed the starting point, but we must also take into account that no host may be assigned the network address itself, and no host may be assigned the network broadcast address. So to determine the number of allowable hosts in the network, we raise 2 to the power of the number of bits and subtract 2 from that number.

   If we use the CIDR entry from step 1, 96.45.82.5/20, we know that the network identifier spans 20 bits. To calculate the number of possible hosts for that network (96.45.80.0), we raise 2 to the power of 12 (32-20), which comes to 4,096, then subtract 2 from it to account for the network address and the broadcast address. So we conclude that the network represented by 96.45.82.5/20 has 4,094 possible hosts. An alternative way to state this is that the network represented by 96.45.82.5/20 has 4,094 assignable IPv4 addresses.

   Not only do we want to know the number of possible IPv4 addresses, we also want to know the range of assignable IPv4 addresses for the network. Once we know that broadcast and network addresses, determining this range is easy. We just add 1 to the network address to obtain the first assignable IPv4 address, and subtract 1 from the broadcast address to obtain the last assignable IPv4 address.

   So still using the example entry 96.45.82.5/20, we know that the network address is 96.45.80.0, so by adding 1 to this, we know the first assignable address is 96.45.80.1. We know that the broadcast address is 96.45.95.255, and so subtracting 1 from that gives us the last assignable address, 96.45.95.254. So the range of assignable addresses for 96.45.82.5/20 is 96.45.80.1 through 96.45.95.254.

4. Now, you give this a try with the CIDR entries below. For each one, give the number of assignable IPv4 addresses, as well as the range of assignable addresses, using the methodology listed in step 3. Make sure to show your work, using a tool if needed to convert from decimal to binary and vice versa.

   66.147.242.171/15
   207.69.200.144/24

5. Recall that any network whose identifier can be separated into a network prefix and a subnet identifier is a subnet, and conceptually we view a subnet as a sub-network that exists within the context of an overarching network. What this means is that, although the computers and devices on any particular network use the network identifier as an atomic number, we can logically group subnets as belonging to the same network when those subnets have the same network prefix.

   For example, if an organization has been assigned the block of addresses represented by the network identifier 193.39.64.0/18, that is, all IP addresses beginning with 11000001 00100111 01, this does not mean that the organization must use this network prefix atomically to define only one single network. If the organization were to do so, it would have one single network consisting of 2^14 -2, or 16,382, hosts. However, it is uncommon to have a single network consisting of such a large number of hosts due to the inefficiencies and management complexity that would create. Instead, the organization would subdivide its networks into smaller networks, each consisting of a smaller number of hosts. These smaller networks are subnets. We now begin to discern the meaningful distinction between a network and a subnet.

   Further, we have now come upon a new concept, which is that of *the number of subnets per network.* Continuing with the example of an organization that has been given the block of addresses beginning with 193.39.64.0/18, suppose the organization decides that it wants to divide its overall network into 8 smaller networks (subnets). How many bits would it need to do this? 3, because 2^3 is 8! Thus, the organization would use 11000001 00100111 01 as its network prefix, then the next three bits to uniquely identify each subnet. The subnet identifiers would be 000, 001, 010, 011, 100, 101, 110, and 111. And the overall network identifiers for these subnets would be:

```
First subnet:  11000001 00100111 01000
Second subnet: 11000001 00100111 01001
Third subnet:  11000001 00100111 01010
Fourth subnet: 11000001 00100111 01011
Fifth subnet:  11000001 00100111 01100
Sixth subnet:  11000001 00100111 01101
Seventh subnet:11000001 00100111 01110
Eighth subnet: 11000001 00100111 01111
```

Did you catch all this? The organization is assigned a network prefix. Rather than using this prefix to create a single, large network, the organization subdivides that network into 8 subnets, and those subnets are identified by an additional 3 bits appended to the end of the network prefix. Thus, *from the perspective of the organization, the network identifier consists of 18 bits, but from the perspective of each subnet, the network identifier consists of 21 bits*. Think about the last statement carefully until you are sure you understand it. Understanding this perspective is crucial understanding the content in the remainder of this lab.

Now we can begin to answer the questions about subnets and combine that with what we have learned previously. For example, if we are given the information that an organization is assigned the block of addresses corresponding to the network prefix 161.225.130.163/20, and decides to subdivide this into 4 subnets, we could answer the following questions.

a. How many bits are needed for the subnet identifier, to distinguish one subnet from another on this network?
Because there are 4 subnets, we would use 2 bits since 2^2 = 4.

b. How many hosts are there per subnet?
We know that 20 bits are taken for the network prefix, and an additional 2 bits are taken for the subnet, so the network identifier for each subnet is 22 bits. That leaves 10 bits for the hosts. 2^10 = 1,024, and subtracting 2 from that gives us 1,022 possible hosts.

c. What are the network identifiers for each subnet, in binary?
First, we need to represent the network identifier for the organization, in binary, which is 10100001 11100001 1000. We obtained this by converting the first three octets to binary, and only selecting the first 4 bits from the third octet. Next, we append the four combinations of 2 bits together to calculate the network identifier for each subnet.

```
First subnet:  10100001 11100001 100000
Second subnet: 10100001 11100001 100001
Third subnet:  10100001 11100001 100010
Fourth subnet: 10100001 11100001 100011
```

d. What are the network addresses for each subnet, in CIDR notation?
We have already represented the network prefixes for each subnet in binary, so we simply convert them to decimal and fill out the remaining bits with 0 bits. We know that 20 bits are taken for the network prefix, and an additional 2 bits are taken for the subnet identifier, so the network identifier for each subnet spans 22 bits. By appending ten 0 bits and then converting each octet from binary to decimal notation, we obtain:

First subnet: 161.225.128.0/22
Corresponds to 10100001 11100001 10000000 000000

Second subnet: 161.225.132.0/22
Corresponds to 10100001 11100001 10000100 000000

Third subnet: 161.225.136.0/22
Corresponds to 10100001 11100001 10001000 000000

Fourth subnet: 161.225.140.0/22
Corresponds to 10100001 11100001 10001100 000000

6. Now, you give this a try with the scenarios below. For each scenario:
   a. give the number of bits needed for the subnet identifier.
   b. give the number of possible hosts for each subnet.
   c. give the network identifiers for each subnet, in binary.
   d. give the network addresses for each subnet, in dotted decimal notation.

   Make sure to use the methodology listed in step 5 and to show your work, using a tool if needed to convert from decimal to binary and vice versa.

   Scenario 1: The organization is given the block of addresses corresponding to 211.174.52.0/22, and decides to create 8 subnets.

   Scenario 2: The organization is given the block of addresses corresponding to 74.86.197.161/15, and decides to create 2 subnets.

7. Wow! Up to now you have learned a lot about working with IPv4 address. You have learned how to represent them in dotted decimal notation and in binary, how to specify the length of the network identifier in CIDR notation, how to work with subnet masks, how to calculate their network and broadcast addresses, and how to determine the number and range of assignable addresses on a network. You have also learned how to subdivide a network into subnets and calculate the corresponding subnet identifiers. In doing so, you have learned many of the addressing concepts integral to the operation of local area and backbone networks throughout the world. Congratulations!

   We can now practice with much of what we learned up to this point all together.

   If an organization is given the range of addresses corresponding to the CIDR entry 209.15.0.0/17, and the organization decides to use 2 subnets, we can answer the following series of questions.

a. How many bits are needed for the subnet identifier, to distinguish one subnet from another on this network?
Because there are 2 subnets, we would use 1 bit.

b. What is the subnet mask for the 2 subnets?
Since the network identifier for the organization is 17 bits, and the subnet identifier is 1 bit, the length of the network identifiers for both of the subnets is 18 bits. So we add 18 1 bits to the mask, followed by 14 0 bits, like so, to arrive at our subnet mask:

11111111 11111111 11000000 00000000

c. How many hosts are there per subnet?
Because the network identifiers are 18 bits, that leaves 14 bits for the hosts. 2^14 = 16,384, and subtracting 2 from that gives us 16,382 possible hosts.

d. What are the network identifiers for each subnet, in binary?
First, we need to represent the network identifier for the organization, in binary, which is 11010001 00001111 0. We obtained this by converting the first three octets to binary, and only selecting the first bit from the third octet. Next, we append the two combinations of bits together to calculate the network identifier for each subnet.

```
First subnet:  11010001 00001111 00
Second subnet: 11010001 00001111 01
```

e. What are the network addresses for each subnet in dotted decimal notation?
We have already represented the network prefixes for each subnet in binary, so we simply convert them to decimal and fill out the remaining bits with 0 bits.

First subnet: 209.15.0.0
Second subnet: 209.15.64.0

f. What are the broadcast addresses for each subnet in dotted decimal notation?
For the first subnet, we know that its network address is 209.15.0.0. When we set all of the 14 host bits to 1, we know that we have 209.15.X.255 because the third octet is partially spanned. To determine X, we list out the third octet with the host bits set to 1, which comes to 00111111. When we convert that to decimal, we see that it is 63. Thus the broadcast address for the first subnet is 209.15.63.255. We can sanity check this because the next address is 209.15.64.0, which starts the next subnet.

For the second subnet, we follow the same steps to arrive at 209.15.X.255. When we list out the third octet, with the host bits set to 1, we arrive at 011111111. For practice, we convert this to decimal by hand. You still remember how to do that, right?

01111111 → 127

Power of 2		128	64	32	16	8	4	2	1
Bit		0	1	1	1	1	1	1	1
Cumulative Amount	0	0	64	96	112	120	124	126	127

Now even by hand we could have taken a shortcut by simply subtracting the highest bit value, 128, from 255 since the first bit is a 0 and the remaining bits are all 1s. That is, if we see an octet that has a single 0 bit coupled with seven 1 bits, we simply subtract that bit's value from 255 to convert it to decimal. Sometimes starting from 0, then adding each 1 bit's value is faster, and sometimes starting from 255, and subtracting each 0 bit's value, is faster.

So by substituting 127 for X, we arrive at a broadcast address of 209.15.127.255 for the second subnet.

g. What are the assignable IP addresses for each subnet?
First subnet: The first address that can be assigned is 1 higher than the network address. Since the network address is 209.15.0.0, the first assignable IP address is 209.15.0.1. The last address that can be assigned is 1 lower than the broadcast address. Since the broadcast address is 209.15.63.255, the highest assignable address is 209.15.63.254. Putting these two together, we can say that all addresses between 209.15.0.1 and 209.15.63.254 are assignable.

Second subnet: Since the network address is 209.15.64.0, the first assignable address is 209.15.64.1. Since the broadcast address is 209.15.127.255, the last assignable address is 209.15.127.254. We can thus say that all addresses between 209.15.64.1 and 209.15.127.254 are assignable.

8. What will we do now? You guessed it! Now it's your turn to try it out.

   Imagine that an organization is assigned the range of addresses identified by CIDR entry 66.216.112.0/21, and that the organization has decided to use 8 subnets. Answer the following series of questions about this scenario. Make sure to show your work, using a tool to convert from binary to decimal and vice versa when necessary. The one exception is that *you are requested to manually convert from binary to decimal for part f only*, in order to help maintain your ability to do so.

a. How many bits are needed for the subnet identifier, to distinguish one subnet from another on this network?

b. What is the subnet mask for the 8 subnets?

c. How many hosts are there per subnet?

d. What are the network identifiers for each subnet, in binary?

e. What are the network addresses for each subnet in dotted decimal notation?

f. What are the broadcast addresses for each subnet in dotted decimal notation? Manually convert from binary to decimal when needed for this part.

g. What are the assignable IP addresses for each subnet?

You are now familiar with many of the addressing concepts integral to the operation of local area and backbone networks throughout the world. Congratulations!