Subnets, IP Addresses, Subnetting

# Subnets and Binary Basics

Let’s say we have an IP address that looks like this: 192.168.3.55 / 24. This is a conventional way to write the IP address and the subnet mask associated with it. But what does it mean?

The 24 is a shorthand way to write the subnet mask. In this case, the subnet mask would be 255.255.255.0 (this is the most common way to write a subnet mask). Let’s break this down into binary code.

The IP address and the subnet mask are made up of what are called octets. Basically, an octet is the binary form of a number in decimal notation. Let’s illustrate this for the subnet mask of 255.255.255.0:

* Access the calculator on your computer
* Click View > Programmer
* Make sure the radio button next to Dec is Highlighted
* Type 255. This is the Dec format of the number 255
* Now click the radio button next to Bin. The number changes to eight ones (an octet). In other words, 11111111 is the same number as 255, only in binary form. The eight ones are called an octet.

So, now we know that for the subnet mask 255.255.255.0, 255 is an octet, then next 255 is an octet, etc. If we wrote out this subnet mask in binary, it would look like the subnet mask below. Note that there are 24 ones.

* Subnet mask: 255.255.255.0
* Binary: 11111111.11111111.11111111. 00000000

Let’s dig a bit deeper into binary code: The way binary code works is through the power of 2s (see the Power of Twos Table below).

|  |
| --- |
| **Powers of Two Table** |
| 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 |
| 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |

Each place in an octet corresponds to one of the powers of two. Note that there are 8 numbers in the table above and eight numbers in a single octet. Each of the numbers in the power of twos table corresponds to a place in a binary octet. For example, the number **1** in the binary number **1**0000000, corresponds to the 128s place. If you use your calculator, you will notice that the Dec number 128 is the number 10000000 (a single 1 and 7 zeros). Likewise, the number 64 corresponds to the next place in the octet. So if you use your calculator, you will notice that the binary for the number 64 corresponds to 1000000 (a single 1 and 6 zeros), and so on. Then notation would be 01000000. Here is an example. For the IP address 192.168.3.55, the binary code would look like: 11000000.10101000.00000011.00110111

# Let’s Practice!

|  |
| --- |
| **Powers of Two Table** |
| 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 |
| 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |

Using the table above, let’s get the binary code for this example IP address: 192.168.3.55.

* To get 192, you would need the numbers to add up to 192. You can only use the numbers in the power of twos table to get the numbers to add up to 192. So you would say:
	+ Do I need a 128? Yes (1)
	+ Do I need a 64? Yes (1); *Note:* 128 + 64 = 192 so you wouldn’t need any other numbers
	+ Do I need a 32? No (0)
	+ Di I need a 16? No (0)
	+ Do I need an 8? No (0)
	+ Do I need a 4? No (0)
	+ Do I need a 2? No (0)
	+ Do I need a 1? No (0)
	+ So your first octet, which corresponds to **192**, would be 11000000. Check this on your calculator.
* To get 168, the next number in the example IP address, use the powers of Two Table (remember, you need to use the numbers in the table to add up to 168).
	+ Do I need a 128? Yes (1)
	+ Do I need a 64? No (0)
	+ Do I need a 32? Yes (1) – *note: 128 + 32 = 160 (we still need 8 more numbers)*
	+ Di I need a 16? No (0)
	+ Do I need an 8? Yes (1) - note: 128 + 32 + 8 = 168
	+ Do I need a 4? No (0)
	+ Do I need a 2? No (0)
	+ Do I need a 1? No (0)
	+ So your second octet, which corresponds to 168, would be 10101000. Check this on your calculator.
* To get 3:
	+ Do I need a 128? No (0)
	+ Do I need a 64? No (0)
	+ Do I need a 32? No (0)
	+ Di I need a 16? No (0)
	+ Do I need an 8? No (0)
	+ Do I need a 4? No (0)
	+ Do I need a 2? Yes (1)
	+ Do I need a 1? Yes (1)
	+ So your third octet, which corresponds to 3, would be 00000011. Check this on your calculator
* To get 55:
	+ Do I need a 128? No (0)
	+ Do I need a 64? No (0)
	+ Do I need a 32? Yes (1)
	+ Di I need a 16? Yes (1)
	+ Do I need an 8? No (0)
	+ Do I need a 4? Yes (1)
	+ Do I need a 2? Yes (1)
	+ Do I need a 1? Yes (1)
	+ So your fourth octet, which corresponds to 55, would be 00110111. Check this on your calculator

So overall, the binary code for 192.168.3.55 is: 11000000.10101000.00000011.00110111

Try it out on your own using the powers of two table. What is the binary code for the following IP Addresses?

* 192.168.10.0
* 192.168.20.101
* 192.168.10.222

# Why does this matter??

This information is important because computerscompare the binary code of the IP Address to the binary code of the subnet mask to determine if the frame and data being sent should either stay on the subnet (LAN) or leave the subnet (WAN), and subsequently hop to another router and another subnet. Here’s how this works:

* Before a computer sends out any data, it first compares the destination IP address to its own IP address using the subnet mask.
* If the destination IP address matches the computer’s IP wherever there’s a 1 in the subnet mask, then the sending computer knows the destination is local. In other words, the network IDs match.
* If any 1s in the subnet masks are different, then the sending computer knows it will be sent to another subnet. In other words, the network IDs do not match.

Using the information below (subnet mask, computer A, and computer B), imagine Computer A is sending data to Computer B. Will the data be sent on a LAN (local) or a WAN (another subnet)? How do you know? Once you know, explain the process through which Computer A sends data to Computer B.

Subnet Mask: 11111111.11111111.11111111.00000000 (255.255.255.0)

Computer A (sender): 11000000.10101000.01001101.00101111 (192.168.77.47)

Computer B (destination IP): 11000000.10101000.01001100.00110110 (192.168.76.54)

* Using what you know above, provide a scenario for Computer A, Computer B, and a Subnet Mask when data would leave the subnet and data would stay in a subnet.

# Subnetting: what happens when the subnet mask changes from / 24?

What if we had an IP Address with the notation of: 192.168.3.55 / 26?

Basically, this means we would have a subnet mask with 26 ones, right? Same IP as before, but different number of ones in the subnet mask. In other words, it would look like 255.255.255.192, and the binary code would have 26 ones (instead of the common 24). The binary would look like this: 11111111.11111111.11111111.11000000. Let’s think about this a bit deeper.

## Network Number/Broadcast Number/Magic Number

Let’s discuss a bit about what IP addresses are reserved for a single subnet. Remember, in the classic example, we have 192.168.3.55 / 24. This means we have the following numbers reserved:

* Subnet Mask: 255.255.255.0
* Network number: 192.168.3.0 (the first possible number on the 192.168.3 network)
* Broadcast number: 192.168.3.255 (the last possible number on the 192.168.3 network)

So, in a single subnet, all of the IP Addresses on the 192.168.3 network fall between 192.168.3.0 and 192.168.3.255, with the.0 and .255 numbers reserved for the network and the broadcast number. So overall, in a single subnet, we have 254 total possible IP addresses to reserve for computers, users, phones, etc.

Again, what if we changed the subnet notation? Subnet notation can change from /24 to /25, /26, /27, and /28. What if we change the 192.168.3.55 /24 subnet notation to a /26 subnet notation? The notation would look like this: 192.168.3.55 / 26. In binary, this means mean we have a subnet mask with 26 ones (instead of the common 24): 11111111.11111111.11111111.11000000 (notice there are 26 ones – the two extra are in red).

So why do this? Doing this is a way to create a group of IP addresses on the same subnet that could be used for different purposes. For example, you could block some numbers for more secure IP addresses and computing, or for bandwidth control. Let’s see how to do this using the Powers of Two Table and the concept of the *Magic Number*.

|  |
| --- |
| **Powers of Two Table** |
| 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 |
| 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |

For the notation 192.168.3.55 / 26, let’s create a group of IP Addresses using the power of two table. To do so, first notice which place in the octet the last ones digit is in? In our case, the last 1 (11111111.11111111.11111111.11000000) is in the 64s place of the octet. So, we would say that 64 is our **MAGIC NUMBER**.

This magic number of 64 enables us to see where and how our IP Addresses will be blocked. In the 192.168.3.55 / 26 schema, we determined that the magic number is 64. This means we can create IP blocks within the subnet that have 64 IP addresses assigned to them. The IP blocks for a /26 subnetting scheme will look like the four blocks below (remember they can only go up to 255).

* Block 1: 192.168.3.0 to 192.168.3.63 (64 numbers)
* Block 2: 192.168.3.64 to 192.168.3.127 (64 numbers)
* Block 3: 192.168.3.128 to 192.168.3.191 (64 numbers)
* Block 4: 192.168.3.192 to 192.168.3.255 (64 numbers)

An elementary school, for example, could use these blocks to assign teachers to a certain block, administrators (principals, deans, etc.) to a certain block, IT administrations to a certain block, etc.

## How to Identify the Number of Subnetworks and Hosts

 To identify how many subnetworks and hosts a network will have, you can look at the binary code for the subnet and do the following. Let’s use 192.168.3.55 / 26 as the example IP address.

* The binary notation is: 11111111.11111111.11111111.11000000
* *Number of subnetworks*:
	+ In the binary notation above, there are two ones in the last octet. So we can say that there will be 22, or four total subnetworks (the four IP blocks above).
* *Number of hosts*:
	+ In the binary notation above, there are six zeros, so 26 equals 64. So we have 64 hosts (i.e., 64 possible IP connections within each group).

To illustrate this example, imagine you are a network administrator and you are trying to determine who the IP address 192.168.3.55 / 26 is on your network. Using the above example, you know that the IP Address would be in Block 1, which could correspond to a group of users (e.g., MIS students, marketing students, etc.). Let’s do some examples.

## Apply Your Work -- Questions on Subnetting

Now let’s apply your knowledge by answering some common questions about subnetting that you may find on the Network+ exam.

1. For the IP Address 192.168.1.153 / 27:
	* What is the subnet mask in binary code?
	* What is the subnet mask in decimal notation?
	* What is the magic number? How did you determine this? (hint: use the powers of 2 table).
	* How many hosts are there in the network?
	* How many subnetworks will the 192.168.1.153 / 27 network have?
	* What network block is the IP address (192.168.1.153) in? List the network blocks.
2. Imagine you are a network administrator and need to setup a network for a school. The admin network has 44 users. The teacher network has 123 users. What would be the proper subnet mask notation (e.g., /24, /25, /26, etc.)? Why?
3. Imagine you are a network administrator and need to setup a network.
	* Admin Network: 44 users
	* Faculty Network: 60 users
	* Media network: 22 users
	* Library network: 12 users
	1. What should be the subnet mask based on the information above (select one)?
		1. 255.255.255.128
		2. 255.255.255.192
		3. 255.255.255.224
		4. 255.255.255.240
		5. 255.255.255.248