A blue rectangular object with white border

AI-generated content may be incorrect.

This blue box represents every IP address in the IPv4 internet. This is referred to as the /0 network. We know that we can split this /0 network into 2 equal /1 networks:

A blue and brown rectangles

AI-generated content may be incorrect.

Each CIDR increment doubles the number of networks. We started with one /0 network, then we ended with 2 equal /1 networks. We can continue and break up each /1 network into 4 equal /2 networks inside our original /0 network. We can continue to /3 to get 8 equal /3 networks. Or break it up into 16 /4 networks, or break it up into 32 /5 networks

A colorful rectangular shapes

AI-generated content may be incorrect.

The key is that this is true regardless of where you start, we started with /0, but if we started with /10, we would have 2 /11 networks, 4 /12 networks, 8 /13 networks and so on:

A colorful rectangular shapes

AI-generated content may be incorrect.

If we started from /22, we would have 2 /23 networks, we would have 4 /24 networks, we would have 8 /25 networks and so on.

A colorful rectangular shapes with different colors

AI-generated content may be incorrect.

Every time you increment the CIDR, you double the amount of sub-networks you have.

This can all be used to answer 2 types of subnetting questions.

If you start with a /18, what size sub-network would you need to crate 100 sub-networks?

The question provides us with a starting point of /18, so if we go to the next CIDR which is /19, we double the amount of sub-networks we have, we will have 2 /19 sub-networks.

So, if we continue to /20, we would have 4 /20 networks, if we continue to /21, we would have 8 /21 networks, if we continue to /22, we would have 16 /22 networks, if we continue to /23, we would have 32 /23 networks, if we continue to /24, we would have 64 /24 networks, if we continue to /25, we would have 128 /25 networks.

A diagram of a network

AI-generated content may be incorrect.

This gives us our answer, if we wanted 100 sub-networks, and we started from /18, we would need a /25 to account for 100 sub-networks.

The same pattern would work regardless of where we start, if we start with a /10 and we wanted 100 sub-networks:

A diagram of a network

AI-generated content may be incorrect.

If we started with /22 and we wanted 100 sub-networks:

A diagram of a network

AI-generated content may be incorrect.

The second type of question is:

If you start with a /21, how many sub-networks could you create that could contain 50 IP addresses?

The question gives us the starting point of /21 and the condition of 50 IP addresses that we need to convert into a subnetwork size. If we go back to our chart, we will know that a /26 network is the smallest size network that would account for a minimum of 50 IP addresses:

A close up of numbers

AI-generated content may be incorrect.

So /26 is our ending point, they are basically asking us how many /26 sub-networks exist in a /21. So, we simply list out every CIDR network between /21 and /26 and double at each increment:

A graph of a number of objects

AI-generated content may be incorrect.

As we can see, 32 sub-networks could create 50 IP addresses with /21 as the starting point. There would be /32 sub-networks inside a /21. This same strategy can be used for any starting point.

If we started with a /23:

A white background with black text and blue lines

AI-generated content may be incorrect.

Here our answer would be 8 sub-networks.

If we started with a /20:

A diagram of a graph

AI-generated content may be incorrect.

The answer here would be 64 sub-networks.

These 2 questions are made up of 3 parts:

* Starting point
* Ending point
* Condition

To simplify we should look out for:

* A starting network size
* Number of sub-networks
* Size of sub-network

Both questions provided a starting point, question 1 provided the number of sub networks as a condition whilst question 2 asked us to solve for it (the number of sub-networks). Question 2 provided the size of the sub-network, whilst question 1 asked us to solve for it (the size of sub-network).

Generally, we would be given 2 pieces out of the 3 and we would have to solve for the third.

Also, both questions result in sub-networks which are all the same size, this is known as FLSM – fixed length subnet masks.

Later we will learn about VLSM – variable length subnet masks, here sub-networks can be different sizes, this allows for a more efficient use of IP space.

There is an easier way to answer both questions, we use this chart:

A white rectangular object with black numbers

AI-generated content may be incorrect.

Lets start with the second question, if you start with a /20, how many sub-networks could you create that could contain 50 IP addresses?

The starting point is given which is /20, and we confirmed earlier that a /26 network has the group size of 64. So, we now have the end point too.

We simply do /26 - /20, which is 6, look on the chart for 2^6 and we see the answer is 64, so the answer to the question would be 64 sub-networks could be created that could contain 50 IP addresses if we start from /20.

If the question told us the starting point was /21, we do /26 - /21 which is 5 and 2^5 is 32 so the answer would be 32 sub-networks could be created that could contain 50 IP addresses if we start from /21.

Now for the first question, if you start with a /18, what size sub-network would you need to create 100 sub-networks?

We can look at the chart and see the value that is equal or greater than 100 which is 128, and we can see it is 2^7, so it tells us that if we start from /18, we need to continue 7 CIDR’s further, so /18 + 7 is /25. The size sub-network we need is a /25.