BRYANT CCNA - SECTION 12 - SUBNET SUCCESS

SEC 12.1 WHYS, INS, AND OUTS

Subnetting is the process of breaking a block of IP Addresses into smaller pieces.

For example, we will start with the network 20.0.0.0/8 and that is the only network block we can use for the rest of this walkthrough

Each subnet, or network block, can only be used on one "side" of a router. So, subnetting is required to create logical segments for our Network.

The other half of subnet decisions is how large to make your subnets. There are two related factors:
- How many subnets needed overall including those reserved for future growth.
- How many host addresses are needed by each subnet.

We want to plan as efficiently as possible, while also leaving "wiggle room" for future growth.

After creating our networks we will need to be prepared to answer these questions:

- what subnet is this IP address on?
- what's the broadcast address for this subnet?
- what's the valid address range for hosts assigned to a particular subnet?

You are more likely (and especially right away!) to be coming into a network someone else setup than to be designing your own, so it is important to be ready!

SEC 12.2 Decimal to Binary conversion

	128	64	32	16	8	4	2	1
45	0	0	1	0	1	1	0	1
			13		5	1		0

SEC 12.3 Decimal to Binary: dotted decimal example

200.17.100.3

	128	64	32	16	8	4	2	1	hex
200	1	1	0	0	1	0	0	0	0x61
17	0	0	0	1	0	0	0	1	0x11
100	0	1	1	0	0	1	0	0	0x64
3	0	0	0	0	0	0	1	1	0x03

SEC 12.4 Binary to Decimal Conversion

	128	64	32	16	8	4	2	1	
11001010	1	1	0	0	1	0	1	0	= 202

SEC 12.5 Number of Valid Subnets: Classic Approach

Always double and triple check the number of valid subnets available in your addressing scheme befory you start assigning addresses!

How many valid subnets exist on the 10.0.0.0/12 (or 10.0.0.0 255.240.0.0) network?

We are much more likely to see prefix notation in the real world, but will have to use both in exam.

	Class A	Class B	Class C
1st octet	1 - 126	128 - 191	192 - 223
mask	255.0.0.0	255.255.0.0	255.255.255.0

Subnetting is performed by borrowing host bits

Back to our question: How many valid subnets are there in the 10.0.0.0/12 network?

We will look at two methods, the first is longer but gives a better view of the process, but the second is quicker and can be used later.

First Method:

Network	1111 1111	0000 0000	00000000	0000 0000
Subnet	1111 1111	1111 0000	0000 0000	0000 0000

8 network bits

4 subnet bits

20 host bits

Number of Valid Subnets: 2^n subnet bits

So, there are 16 valid subnets in this example.

SEC 12.6 Number of Valid Subnets - Bulldog approach

How many valid networks are on the 150.10.0.0/21 network?

1) It's a class B network
B) the class B network mask is /16
c) 21 - 16 = 5
D) 2^5 = 32 valid subnets

Next: 200.1.1.0/27

27-24 = 3

2^3 = 8 valid subnets

SEC 12.7 Number of Valid Hosts

Here we have to start by finding the number of host bits in the network:

$$32 - N_{subnet} = N_{host}$$

Valid Hosts = $2^n - 2$

200.10.10.0/26

32 - 26 = 6 host bits

2^6 = 64 - 2 = 62 valid hosts on 200.10.10.0/26

We subtract two because two addresses in every network are reserved and not usable by hosts:

- the first address is the subnet number itself
- the last address is the broadcast address for that subnet.

SEC 12.8 Finding the subnet number of an IP Address

Sometimes, communication issues result when a host is thought to be on one subnet, but is actually on another subnet.

We will look at three different ways to find which subnet an IP Address is on.

Determine the subnet containing 10.17.2.14/18

1. convert the address to a binary string
2. Add up the network and subnet bits ONLY
3. enjoy victory.

10.17.2.14/18

	128	64	32	16	8	4	2	1
10	0	0	0	0	1	0	1	0
17	0	0	0	1	0	0	0	1
2	0	0	0	0	0	0	1	0
14	0	0	0	0	1	1	1	0

```
10.17.214       0000 1010  0001 0001  0000 0010  0000 1110
255.255.192.0   1111 1111  1111 1111  11000000   00000000
```

The first 18 bits in the binary string give us 10.17.0.0, and that's our subnet – as long as you include the subnet mask. Both 10.17.0.0/18 and 10.17.0.0 255.255.192.0 are acceptable representations.

Now, a faster method: only convert the subnet mask bits to binary. For example: 217.17.23.200/27

	128	64	32	16	8	4	2	1
217	1	1	1	0	1	1	1	1
17	0	0	0	1	0	0	0	1
23	0	0	0	1	0	1	1	1
200	1	1	0					

So: 217.17.23.192/27
or 217.17.23.192 255.255.255.224

An even faster method: only convert the octet where the mask ends. for example: 47.54.100.29/14

	128	64	32	16	8	4	2	1	
54	0	0	1	1	0	1	1	0	
	0	0	1	1	0	1	0	0	= 52

SO: 47.52.0.0/14
OR: 47.52.0.0 – 255.252.0.0

SEC 12.9 Determining Broadcast Address / valid Range

210.46.111.0 /25 | 11010010.00101110.01101111.0|0000000

1. the address for all zeros in the host bits is the subnet address, and not a valid host address
2. the address with all ones in the host bits is the broadcast address

Broadcast: 10.46.111.127
valid: 10.46.111.1 – 10.46.111.126

SEC 12.10 Actually Subnetting, finally

Clients requirements:

1. use network 150.50.0.0
2. at least 200 subnets
3. 120-150 hosts per subnet

We know 150.50.0.0 is aclass B network, so it has a /16 mask. That leases 16 host bits, and we will need to borrow some of those for our subnets.

2 4 8 16 32 64 128 256

So, we need 8 subnet bits to meet the requirement of at least 200 subnets. This will give us 256 subnets.

Now we can check the hosts per subnet:
(2^8)-2 = 254 which is too many host, based on or design requirements. This would waste subnets!

So our answer: 9 subnet bits, 7 host bits

Subnets 2^9 = 512 subnets
hosts (2^7)-2 = 126 host per subnet

So the correct mask is 255.255.255.128 (or /25)

Another Example:

Network: 220.10.10.0
at least 25 subnets
no more than 6 hosts per subnet

220.10.10.0/29 255.255.255.