TCP/IP Cheatsheet v2.1

A Free Study Guide by Boson Software, Inc.
Warning: No warranty expressed or implied.

Check out one of our other great Study Guides, "IP Summary Addressing Cheatsheet" ("IP-SUMMARY.PDF") for detail information on VLSM and CIDR.

Section 1:

Address Class Summary

Internet		Number of Networks		Number of Host Per Network		Range of Network ID's (First Octet)	
	Class A	126		16,777,214		1 -- 126	
	Class B	16,384		65,534		128 -- 191	
	Class C	2,097,152		254		192 -- 223	

The First Octet Rule: High order bits determine the Class of Address

The Most Significant (highest) bits for each Class are frozen, leaving the remainder for the Hosts portion

An Octet is a binary number of 8 bits. The smallest number is '00000000' and the largest is '11111111'

A shortcut is: 2^8 This 2 to the power of 8 is the largest octet (all 8 bits are 1's). It's 255 in decimal.

	Binary Start		Binary End			First Octet	Most Sig Bits	Special Note
Class A =	**00000000**	**-**	**01111111**	**=**	Class A =	**1-126**	**0**	Assignable
Class B =	**10000000**	**-**	**10111111**	**=**	Class B =	**128-191**	**10**	Assignable
Class C =	**11000000**	**-**	**11011111**	**=**	Class C =	**192-223**	**110**	Assignable
Class D =	**11100000**	**-**	**11101111**	**=**	Class D =	**224-239**	**1110**	Multicast
Class E =	**11110000**	**-**	**11110111**	**=**	Class E =	**240-247**	**11110**	InterNIC

Total # Networks per Class		**Network**
Class A	**01111111.**00000000.00000000.00000000 = 2 to the 7th power -2 = 126	Net.H.H.H
Class B	**10111111.11111111.**00000000.00000000 = 2 to the 14th power = 16384	Net.Net.H.H
Class C	**11011111.11111111.11111111.**00000000 = 2 to the 21st power = 2097152	Net.Net.Net.H

Total # Hosts per Network		**Host**
Class A	00000000.**11111111.11111111.11111111** = 2 to the 24th power - 2 = 16777214	x.255.255.255
Class B	00000000.00000000.**11111111.11111111** = 2 to the 16th power -2 = 65534	x.y.255.255
Class C	00000000.00000000.00000000.**11111111** = 2 to the 8th power - 2 = 254	x.y.z.255

Reserved Address Space

*** RFC 1166 & 1918 = Private (internal use only) address space**

Netblock	Special Use	Reference
10.X.X.X	Private	RFC 1918
127.X.X.X	Loopback	Diagnostics
172.(16-31).X.X	Private	RFC 1918
192.0.0.X	Reserved	JBP
192.0.1.X	Backbone-Test-C	RH6
192.0.2.X	Internet-Test-C	JBP
192.0.(3-255).X	Unassigned	NIC
192.1.(0-1).X	Backbone Local Nets	SGC
192.1.2.X	Backbone Fiber Nets	SGC
192.1.3.X	Backbone Apollo Nets	SGC
192.168.X.X	Private	RFC 1918

Section 2:

Binary Breakdown - Decimal to Binary

Step1 is always drawing out this chart, from the right to the left, each time doubling the value starting at 1.

Most Significant Bit | Weighted Values | Least Significant Bit

128	64	32	16	8	4	2	1
X	X	X	X	X	X	X	X

Step 2 is subtracting each Weighted Value from your Decmial value, until you reach 0.

Let's say we want to conver the decimal value 29 to binary. It goes like this:
Can 128 go into 29? No, so it's a 0. Can 64 go into 29? No, so it's a 0. You get the idea…

Most Significant Bit | Weighted Values | Least Significant Bit

128	64	32	16	8	4	2	1
0	0	0	1	1	1	0	1

After going all the way down the Weighted Values chart, we now have our answer: **00011101**

Binary Breakdown - Binary to Decimal

Step1 is always drawing out this chart, from the right to the left, each time doubling the value starting at 1.

Most Significant Bit | Weighted Values | Least Significant Bit

128	64	32	16	8	4	2	1
X	X	X	X	X	X	X	X

Step 2 is adding up the binary values under the Weighted Values, until you reach your decimal value.

Let's say we want to conver the binary value 11000011 to decimal. It goes like this:
Draw a line under 128 and add 128, draw a line under 64 and add 64, draw a line under 2 and add 2, etc.

Most Significant Bit | Weighted Values | Least Significant Bit

128	64	32	16	8	4	2	1
1	1	0	0	0	0	1	1

After going all the way down the Weighted Values chart, we now have our answer: 128+64+2+1 = **195**

Section 3:

Subnet Masking

This is the idea of taking the larger network block and chopping it into smaller pieces of equal size.
Routers require this logical segmentation to be able to address different logical subnets.
Clients require a default gateway IP address (the routers interface) to get off their local subnet.
The subnet mask is more important to a router than any other TCP/IP value.
The number of bits that are used in the subnet mask determine how many logical subnets you get.

There are a few rules involved in subnet masking.

1. The subnets cannot be all 0 or all 1 in the network or host portion of the address.
2. The all 0 subnet is called Subnet Zero, and sometimes can be used (not recommended).
3. The all 1 subnet is called the All 1's subnet, and can never be used (directed broadcast).
4. The incremental value is the IP host address starting point for the subnet.
5. Each incremental value represents another logical subnet.
6. Routers only care about the Net ID, and the subnet Broadcast.
7. Clients only care about local IP address on their subnet, and their subnet broadcast.
8. Each bit represents a power of 2. The easiest way to determine subnets is to use powers of 2.

Binary Mask	Decimal Mask		Binary Hosts	Bits Used	Number of Subnets	Valid Host Increments
00000000	0	<--->	00000000	0	Net ID	Not a Sub
10000000	128*	<--->	00000001	1	1	128
11000000	192	<--->	00000011	2	2	64
11100000	224	<--->	00000111	3	6	32
11110000	240	<--->	00001111	4	14	16
11111000	248	<--->	00011111	5	30	8
11111100	252	<--->	00111111	6	62	4
11111110	254*	<--->	01111111	7	126	2 *
11111111	255*	<--->	11111111	8	254	1 *

* = 7 subs valid for Class A or B in 1st octet. Class C has only 5 valid - the last 2 are binary all 1.

Section 4:

Logical Addressing

*** All SubNets can be broken down to: 7 ranges, 3 classes each (A, B, C)**

Max. No. Subnets	Math For Host ID	Max. Host per Subnet	Subnet Mask	Sub Bits Required	Host Bits Available	
Class A						
0	2^{24-2}	16,777,214	255.0.0.0	0	24	**<u>Class A</u>**
1	2^{23-2}	Invalid	128 = Subnet 0	1	23	Not recommended
2	2^{22-2}	4,194,302	255.192.0.0	2	22	Range 1/7
6	2^{21-2}	2,097,150	255.224.0.0	3	21	Range 2/7
14	2^{20-2}	1,048,574	255.240.0.0	4	20	Range 3/7
30	2^{19-2}	524,286	255.248.0.0	5	19	Range 4/7
62	2^{18-2}	262,142	255.252.0.0	6	18	Range 5/7
126	2^{17-2}	131,070	255.254.0.0	7	17	Range 6/7
Class B						
0	2^{16-2}	65,534	255.255.0.0	8	16	**A' Range 7/7 & Start B**
1	2^{15-2}	Invalid	128 = Subnet 0	9	15	Not recommended
2	2^{14-2}	16,382	255.255.192.0	10	14	Range 1/7
6	2^{13-2}	8,190	255.255.224.0	11	13	Range 2/7
14	2^{12-2}	4,094	255.255.240.0	12	12	Range 3/7
30	2^{11-2}	2,046	255.255.248.0	13	11	Range 4/7
62	2^{10-2}	1,022	255.255.252.0	14	10	Range 5/7
126	2^{9-2}	510	255.255.254.0	15	9	Range 6/7
Class C						
0	2^{8-2}	254	255.255.255.0	16	8	**B' Range 7/7 & Start C**
1	2^{7-2}	Invalid	128 = Subnet 0	17	7	Not recommended
2	2^{6-2}	62	255.255.255.192	18	6	Range 1/5
6	2^{5-2}	30	255.255.255.224	19	5	Range 2/5
14	2^{4-2}	14	255.255.255.240	20	4	Range 3/5
30	2^{3-2}	6	255.255.255.248	21	3	Range 4/5
62	2^{2-2}	2	255.255.255.252	22	2	Range 5/5

Section 5:

Class C example breakdown

It is only necessary to focus in on the octet that has been broken at the bit boundary.
Disregard all other octets, they will either be 255 (at the front) or 0 (at the end).
Remember, a decimal value of 255 means 8 bits of binary masking.

Using 2 bit subnet mask: Class C: 255.255.255.192 & Class B: 255.255.192.0 & Class A: 255.192.0.0

SUBNET	OCTET	RANGE	Incremental Value	DESCRIPTION
*Network (Subnet 0).	00/000000*	0*	64	Net ID (goes in routing table)
(do not use)	00/000001*	1*		First Host ID in Subnet 0
	00/111110*	62*		Last Host ID in Subnet 0
	00/111111*	63*		Broadcast for only Subnet 0
Subnetwork 1 of 4	01/000000	64	64	Net ID (goes in routing table)
	01/000001	65		First Host ID for Sub 1
	01/111110	126		Last Host ID for Sub 1
	01/111111	127		Broadcast for only Subnet 1
Subnetwork 2 of 4	10/000000	128	64	Net ID (goes in routing table)
	10/000001	129		First Host ID for Sub 2
	10/111110	190		Last Host ID for Sub 2
	10/111111	191		Broadcast for only Subnet 2
*Broadcast subnet	11/000000*	192*	64	Net ID (goes in routing table)
(do not use)	11/000001*	193*		First Host ID in Sub All 1's
	11/111110*	254*		Last Host ID in Sub All 1's
	11/111111*	255*		Local Wire (All Subnets) Broadcast

Section 5: CONTINUED

Using 3 bit subnet mask: Class C: 255.255.255.224 & Class B: 255.255.224.0 & Class A: 255.224.0.0

SUBNET	**OCTET**	**RANGE**	**Incremental Value**	**DESCRIPTION**
*Network (Subnet 0).	000/00000*	0*	32	Net ID (goes in routing table)
(do not use)	000/00001*	1*		First Host ID in Subnet 0
	000/11110*	30*		Last Host ID in Subnet 0
	000/11111*	31*		Broadcast for only Subnet 0
Subnetwork 1 of 8	001/00000	32	32	Net ID (goes in routing table)
	001/00001	33		First Host ID for Sub 1
	001/11110	62		Last Host ID for Sub 1
	001/11111	63		Broadcast for only Subnet 1
Subnetwork 2 of 8	010/00000	64	32	Net ID (goes in routing table)
	010/00001	65		First Host ID for Sub 2
	010/11110	94		Last Host ID for Sub 2
	010/11111	95		Broadcast for only Subnet 2
Subnetwork 3 of 8	011/00000	96	32	Net ID (goes in routing table)
	011/00001	97		First Host ID for Sub 3
	011/11110	126		Last Host ID for Sub 3
	011/11111	127		Broadcast for only Subnet 3
Subnetwork 4 of 8	100/00000	128	32	Net ID (goes in routing table)
	100/00001	129		First Host ID for Sub 4
	100/11110	158		Last Host ID for Sub 4
	100/11111	159		Broadcast for only Subnet 4
Subnetwork 5 of 8	101/00000	160	32	Net ID (goes in routing table)
	101/00001	161		First Host ID for Sub 5
	101/11110	190		Last Host ID for Sub 5
	101/11111	191		Broadcast for only Subnet 5
Subnetwork 6 of 8	110/00000	192	32	Net ID (goes in routing table)
	110/00001	193		First Host ID for Sub 6
	110/11110	222		Last Host ID for Sub 6
	110/11111	223		Broadcast for only Subnet 6
*Broadcast subnet	111/00000*	224*	32	Net ID (goes in routing table)
(do not use)	111/00001*	225*		First Host ID in Sub All 1's
	111/11110*	254*		Last Host ID in Sub All 1's
	111/11111*	255*		Local Wire (All Subnets) Broadcast

Section 6:

Powers of 2 Shortcut

Think of subnetting as stealing from Peter to give to Paul.

You have a maximum number of bits (determined by class) to play with, and if you overlap, it won't work.

Steal host bits moving from Left to Right to acquire more Subnets.

Steal subnet bits moving from Right to Left to acquire more Hosts per Sub.

Left-to-Right: Use Network *n* bits (2^*n*)-2=x to get *x* total number of SubNets per stolen Host bit.

Example: Steal 3 Host bits out of total 8 for Subnets on a Class C (see below)
Answer: This means you can have (2^3)-2 = (8-2) = 6 Subnets from this 3-bit Mask.
Note: The leading 3 bits (11100000) = 128+64+32 = 224 Decimal SubNet Mask.

Right-to-Left: Use remaining Host *h* bits (2^*h*)-2=x to get total *x* number of Hosts per SubNet.

Example: There are 5 Host bits remaining out of 8 after using 3 for Subnets on Class C (see above)
Answer: This means you can have (2^5)-2 = (32-2) = 30 Hosts per Subnet.
Note: The Subnet Mask answer would be either slash notation /27 or decimal 255.255.255.224
You are using a Class C address, with 3 bits of subnetting, and 5 host bits remaining.

Class A	Bits Req (n)	Subnets (2^n-2)	Subnets (Decimal)	Hosts Per (2^(24-n)-2)	Hosts Per (Decimal)	Slash (Notation)	Masks (Decimal)	Sub (Slice)
Total 24 bits	1	(2^1)-2	2-2=0	(2^23)-2	8,388,606	/9	255.128.0.0	Subnet 0
to use for	2	(2^2)-2	4-2=2	(2^22)-2	4,194,302	/10	255.192.0.0	Sub 1/7
subnetting	3	(2^3)-2	8-2=6	(2^21)-2	2,097,150	/11	255.224.0.0	Sub 2/7
	4	(2^4)-2	16-2=14	(2^20)-2	1,048,574	/12	255.240.0.0	Sub 3/7
	5	(2^5)-2	32-2=30	(2^19)-2	524,286	/13	255.248.0.0	Sub 4/7
	6	(2^6)-2	64-2=62	(2^18)-2	262,142	/14	255.252.0.0	Sub 5/7
	7 *	(2^7)-2	128-2=126	(2^17)-2	131,070	/15	255.254.0.0	Sub 6/7
	8 *	(2^8)-2	256-2=254	(2^16)-2	65,534	/16	255.255.0.0	Sub 7/7

* = 7 subs valid for Class A or B in 1st octet. Class C has only 5 valid - the last 2 are binary all 1.

Class B	Bits Req (n)	Subnets (2^n-2)	Subnets (Decimal)	Hosts Per (2^(16-n)-2)	Hosts Per (Decimal)	Slash (Notation)	Masks (Decimal)	Sub (Slice)
Total 16 bits	1	(2^1)-2	2-2=0	(2^15)-2	32,766	/17	255.255.128.0	Subnet 0
to use for	2	(2^2)-2	4-2=2	(2^14)-2	16,382	/18	255.255.192.0	Sub 1/7
subnetting	3	(2^3)-2	8-2=6	(2^13)-2	8,190	/19	255.255.224.0	Sub 2/7
	4	(2^4)-2	16-2=14	(2^12)-2	4,094	/20	255.255.240.0	Sub 3/7
	5	(2^5)-2	32-2=30	(2^11)-2	2,046	/21	255.255.248.0	Sub 4/7
	6	(2^6)-2	64-2=62	(2^10)-2	1,022	/22	255.255.252.0	Sub 5/7
	7 *	(2^7)-2	128-2=126	(2^9)-2	510	/23	255.255.254.0	Sub 6/7
	8 *	(2^8)-2	256-2=254	(2^8)-2	254	/24	255.255.255.0	Sub 7/7

* = 7 subs valid for Class A or B in 1st octet. Class C has only 5 valid - the last 2 are binary all 1.

Class C	Bits Req (n)	Subnets (2^n-2)	Max Subs (Decimal)	Hosts Per (2^(8-n)-2)	Hosts Per (Decimal)	Slash (Notation)	Masks (Decimal)	Sub (Slice)
Total 8 bits	1	(2^1)-2	2-2=0	(2^7)-2	0	/25	5.255.255.128	Subnet 0
to use for	2	(2^2)-2	4-2=2	(2^6)-2	62	/26	5.255.255.192	Sub 1/5
subnetting	3	(2^3)-2	8-2=6	(2^5)-2	30	/27	5.255.255.224	Sub 2/5
	4	(2^4)-2	16-2=14	(2^4)-2	14	/28	5.255.255.240	Sub 3/5
	5	(2^5)-2	32-2=30	(2^3)-2	6	/29	5.255.255.248	Sub 4/5
	6	(2^6)-2	64-2=62	(2^2)-2	2	/30	5.255.255.252	Sub 5/5

Section 7:

Real World Walkthrough

Given: You have address 132.7.0.0. You need 5 equal-size SubNets with 1,500 Hosts Per Sub.
Objectives: Compute the following information, in the following order:

1. Find the number of Host bits to steal to get the required number of Subs.
2. Find the number of Hosts per Subnet you will get.
3. Find the decimal value of the new subnet mask.
4. Find the Incremental Value of the Subnets.
5. Find the First Host, Broadcast, and Last Host of each Subnet.

1.) Using powers of 2, first see how many bits you need to steal from Hosts to acquire 5 subnets.

- a. First octet is 132, which is in the range of 128-191, so this is a Class B. Each octet has 8 bits.
- b. Class B address have the first 2 octets (16 bits) locked in, so you can't touch those. There are 2 octets (16 bits) remaining.
- c. Going from Left to Right, start gobbling up Host bits. If you hit 16, you are out of bounds, and it won't work.
- d. Steal 1 bit? (2^1)-2=0, no that never works
- e. Steal 2 bits? (2^2)-2=2, not enough subs yet, keep going
- f. Steal 3 bits? (2^3)-2= 6, hey we found it, only need 3 bits for 6 subs (leaves only 1 sub for future expansion!!)
- g. Write those first 3 bits into your Weighted Values chart.

	Most Significant Bit			Weighted Values			Least Significant Bit		
Subs	128	64	32	16	8	4	2	1	Hosts
	1	1	1	0	0	0	0	0	

2.) Using powers of 2, next see how many Host bits you have remaining.

- a. Going from Right to Left, count the remaining bits. This will be how many Hosts per Sub you get.
- b. Since we know we had 16 host bits total for our Class B, and we stole 3 bits, that leave us with 13 bits for Hosts.
- c. Compute (2^13)-2= 8,190 Hosts Per Sub. Wow!!
- d. Since this is so many more Hosts than we need, consider stealing extra Host bits to create extra Subs for future expansion!!

3.) Using the Weighted Values chart, find out what the decimal Subnet Mask will be.

- a. Simply add 128 + 64 + 32 = 224 from the chart above.
- b. Since this was a Class B and the first 2 octets are reserved, the default mask is 255.255.0.0
- c. We stole the first 3 bits out of octet #3, so that is the only octet we really ever cared about.
- d. The new decimal mask is 255.255.224.0, or shorthand notation /19 (8+8+3=19)

4.) Using the Decimal Mask, OR the last Network Bit's Weighted Value, find the Incremental Value.

- a. Option 1: Take the new Subnet mask (the octet found in Step 3a above) and subtract from 256.
- a. Option 2: Look at the Weighted Values chart and find the last bit flipped to 1 going Left to Right.
- b. Either way, we now have the Incremental Value of 32.
- c. This means Valid Subnetwork #1 is going to be 32, and each valid subnet will increase by 32, until the Subnet Mask is hit.

5.) Using the Incremental Value, count up and find the First Host IP, Broadcast for that Sub, and Last Host IP.

- a. Start at the Incremental Value, which is 32.
- b. Add up the next Incremental Value, which is 32 + 32 = 64.
- c. Take one LESS than the NEXT Incremental value (64-1=63), and that is the PREVIOUS subnets Broadcast (for sub 32).
- d. Add one to the current Incremental Value (32+1=33), that is your First Host (for sub 32).
- e. Subtract one from the current subnets Broadcast (63-1=62), that is your Last Host (for sub 32).
- f. See the chart below for details….

Notes: There are (2^3)-2=6 subnets created. The -2 is important, because you cannot use all 0 or all 1 subnets.
The special all 0 address is the network ID for that subnet, and will be used by a router in its routing table.
The special all 1 address is the network broadcast for all subnets on this wire.

Range 0 x 32	0	Subnet # 0	Subnet Zero should be considered invalid on any vendor's certification exam.
Range 1 x 32	**32**	Subnet #1	Increment 32 = Subnet ID used by Routing Table
			First Host = 33
			Last Host = 62
			Subnet Broadcast = 63
Range 2 x 32	**64**	Subnet #2	Increment 64 = Subnet ID used by Routing Table
			First Host = 65
			Last Host = 94
			Subnet Broadcast = 95
Range 3 x 32	**96**	Subnet #3	One less than 128 is 127, so that is the broadcast for the 96 subnet.
Range 4 x 32	**128**	Subnet #4	One less than 160 is 159, so that is the broadcast for the 128 subnet.
Range 5 x 32	**160**	Subnet #5	One less than 192 is 191, so that is the broadcast for the 160 subnet.
Range 6 x 32	**192**	Subnet #6	One less than 224 is 223, so that is the broadcast for the 192 subnet.
Range 7 x 32	224	Subnet # 7	Broadcast Reserved

Section 8:

Flash Cards to Practice
(Print this page and cut out these boxes)

"Bit combinations per Mask"

Decimal	Hosts per	Class Valid
192 Mask	(64-2)=62	Any Class 1/7
224 Mask	(32-2)=30	Any Class 2/7
240 Mask	(16-2)=14	Any Class 3/7
248 Mask	(8-2)= 6	Any Class 4/7
252 Mask	(4-2)= 2	Any Class 5/7
254 Mask *	(2-1)=1	Class A or B only
255 Mask *	(1-1)=0	Class A or B only

Quiz yourself 1/2: "What is the Mask for xx Hosts in Class C?"
Quiz yourself 2/2: "How many Hosts for xx Mask in Class C?"

"Valid SubNet Masks"

11000000	2 bits	192 mask	Any Class
11100000	3 bits	224 mask	Any Class
11110000	4 bits	240 mask	Any Class
11111000	5 bits	248 mask	Any Class
11111100	6 bits	252 mask	Any Class
11111110 *	7 bits	254 mask	Class A or B only
11111111 *	8 bits	255 mask	Class A or B only

Quiz yourself: "What are the 7 valid subnet mask bits?"

"Valid Subnets per Bit"

Mask bits	Increment	Subnets per
2 bits	64	2 subnets
3 bits	32	6 subnets
4 bits	16	14 subnets
5 bits	8	30 subnets
6 bits	4	62 subnets
7 bits	2	126 subnets
8 bits	1	254 subnets

Quiz yourself: "How many SubNets are in those 7 Masks?"

"Incremental Value of each Mask"

192 mask	Inc. = 64	Any Class
224 mask	Inc. = 32	Any Class
240 mask	Inc. = 16	Any Class
248 mask	Inc. = 8	Any Class
252 mask	Inc. = 4	Any Class
254 mask	Inc. = 2	Class A or B only
255 mask	Inc. = 1	Class A or B only

Quiz yourself: "What is the Incremental Value of each of the 7 Masks?"

"Valid SubNets per Mask"

192 mask	2 subnets	Any Class
224 mask	6 subnets	Any Class
240 mask	14 subnets	Any Class
248 mask	30 subnets	Any Class
252 mask	62 subnets	Any Class
254 mask *	126 subnet:	Class A or B only
255 mask *	254 subnet:	Class A or B only

Quiz yourself: "How many Subnets do you get on which masks?"

"Address Class Ranges"

Class A	1-126	Network ID 0
Class B	128-191	Network ID 10
Class C	192-223	Network ID 110

Quiz yourself 1/3: "What is the range for Class A?"
Quiz yourself 2/3: "What is the range for Class B?"
Quiz yourself 3/3: "What is the range for Class C?"

"Octet Bit Breakdown"

Binary 128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 = Decimal 255

Quiz yourself 1/2: "What is 255 in binary?"
Quiz yourself 2/2: "What are all 8 binary bits in Decimal?"

	Most Significant Bit			Weighted Values			Least Significant Bit		
Subs	128	64	32	16	8	4	2	1	Hosts
	X	X	X	X	X	X	X	X	