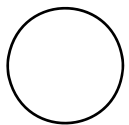




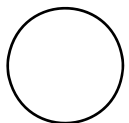
# BUSINESS

- Next week is a lab week to work on linux. I will not be here so use the time to work on cislabs and get your labs up to date.



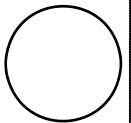
# NETWORK AND HOST PORTIONS

- Layer 3 devices on the network only care about what network is involved.
- Class C network
- BA.CC.DD.XX
- In the layer three example, the XX portion is the host since 24 bits are reserved for the network.



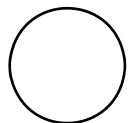
# THE BINARY AND

- $1 \& 1 == 1$
- $1 \& 0 == 0$
- $0 \& 1 == 0$
- $0 \& 0 == 0$
- Remember 0 is false or off



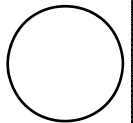
# THUS, HOW IS THE NETWORK RESOLVED

- Class C (or /24 CIDR)
  - 255.255.255.0 (ff.ff.ff.0)
  - 11111111.11111111.11111111.00000000 (as a mask)
  - Consider an address like: 138.96.200.150/24
  - 10001010.01100000.11000100.10010110
  - 11111111.11111111.11111111.00000000
  - &
  - 10001010.01100000.11000100.00000000
  - So the network is 138.96.200.0 which is pretty obvious if you think about /24 masks



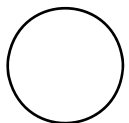
## CLASS B

- Try this:
- 144.25.12.19/16
- 10010000.00011001.00001100.00010011
- 11111111.11111111.00000000.00000000
- 10010000.00011001.00000000.00000000
- Fixed length subnets are when the break is between octets



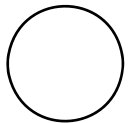
## THE BAD NEWS

- Variable length subnets are ways to break further the network and host groups when the fixed length is not possible (class C networks).
- The fixed length portions are predefined (by rules) in the sense that a Class C is /24 bits that you cannot manipulate. Thus, you are left manipulating the remaining 8 bits for host and network.



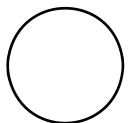
## BASIC RULE

- If a bit is on in the subnet mask (1), that bit is dedicated to the network portion of the address.
- If a bit is off in the subnet mask (0), that bit is dedicated to the host portion of the address.
- The network administrator may choose to divide an octet into smaller segments for security or management purposes



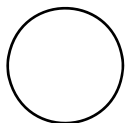
## SIMPLE EXAMPLE

- Say you have class C 200.200.15.0 and you fix the length CIDR /24
- 11000100.11000100.00001111.00000000
- 11111111.11111111.11111111.00000000
- Now any time a bit in an address changes in any of the places indicated by 1 in the mask, the network changes!



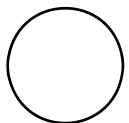
## SIMPLE EXAMPLE CONT.

- 11000100.11000100.00001111.00000000
- 11111111.11111111.11111111.00000000
- 11000100.11000100.00001111.00000000
- $V_s$
- 11000100.11000100.00010000.00000000
- 11111111.11111111.11111111.00000000
- 11000100.11000100.00010000.00000000
- Now you are on a different network 16 vs 15
- In fact in this particular scenario all hosts (255) in the 4<sup>th</sup> octet are in the same network. Only when a bit changes in the 3<sup>rd</sup> octet (or elsewhere) does it change



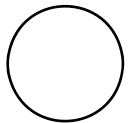
So:

- Say you have a class C network address
  - 200.200.50.0
  - If you committed a single extra bit to the network your mask would be:
  - 255.255.255.128
  - 11111111.11111111.11111111.10000000
  - This means you sacrificed something, so what addresses are on which segments in this model



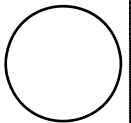
## ANSWER

- 11000100.11000100.00110010.00000000
- 11111111.11111111.11111111.10000000
- So now any network that changes the first bit of the 4<sup>th</sup> Octet, is a change in the network segment
- So the two networks (and their hosts) are
- 00000000 – 01111111 (0 – 127)
- 10000000 – 11111111 (128 – 255)



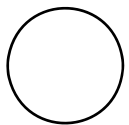
So

- If you saw the address:
- 200.200.50.102/25
- Is it on the 0 net or the 128 net?
- 0 network.



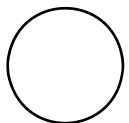
## A PROBLEM IN SUBNETTING

- Suppose you have a class c network
- 175.28.13.0 and you want to establish 5 networks. How many hosts will be available per network?



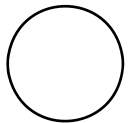
# FIRST FIGURE OUT HOW MANY BITS ARE NEEDED FOR 5 NETWORKS

- The 0 network problem:
  - Can a network have a 0 address?
  - It actually can with most modern equipment if the routers are running beyond RIP1 (OSPF, EGRP, et. Al)
  - So, if you include 0 as a network:
    - 1000 0000 (2 networks)
    - 1100 0000 (00, 01, 10, 11 – 4 networks)
    - 1110 0000 (000, 001, 010, 011, 100, 101, 110, 111 – 8 networks)



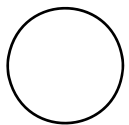
So:

- The only way to do it is to mask as
- 1110 0000 or 224
- Then you can figure up how many hosts are left:
  - 000 00000 – 000 11111 == 0 - 31
  - 001 00000 – 001 11111 == 32 - 63
  - 010 00000 – 010 11111 == 64 - 95
  - 011 00000 – 011 11111 == 96 .....
  - 100 00000 – 100 11111
  - 101 00000 – 101 11111
  - 110 00000 – 110 11111
  - 111 00000 – 111 11111



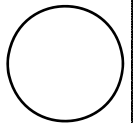
# ANSWER

- 8 networks
- 32 hosts (if you count the 0s) per network.
- Normally, they subtract 2 (0 and broadcast) for 30 total hosts



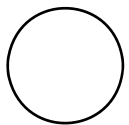
## ANOTHER EXAMPLE

- Suppose you have a class C network and need 17 networks, how many hosts are possible?



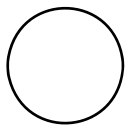
# ANSWER

- 11111000
- 31 networks possible
- This leaves only three bits for host addresses
- Or 000, 001, 010, 011, 100, 101, 110, 111 or 8 hosts including the 0 and broadcast.



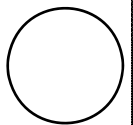
## OK, ONE MORE PROBLEM

- High order bits are the standard for networking masks and low order bits are the standard for host portions
- Suppose you saw this mask
- 255.255.255.146
- Would it work?
- How many networks and hosts are available?



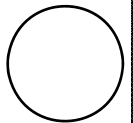
## SOLUTION

- 146 is 10010010
- Yes it would work
- Three bits in the network means 8 possible
- 5 bits in the host means 31 hosts per network
- What are the addresses? Good luck



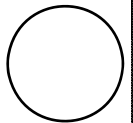
# THE COMPUTER DOESN'T CARE

- Bottom line, if a bit is on, it is network, if it is off, it is host. Other than that, it is just a binary and for the whole operation.



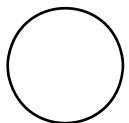
# SUPERNETS

- And if you can play with the bits in one direction, you can play with them in the other.
- What if instead of seperating networks into parts, you wanted to collapse networks together into single segments?



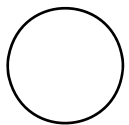
## A SIMPLE EXAMPLE

- Suppose you have the subnet:
- /25 (255.255.255.128)
- That means that you have two networks in your class c address. If you suddenly switch to /24 (255.255.255.0) you have collapsed the 0 and 128 networks into a single segment. This is an example of supernetting.



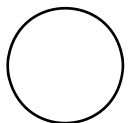
## WHY?

- Suppose you need 300 hosts on a network but when you call your local network ip broker, they are only willing to sell class C networks 200.200.50.0 and 200.200.51.0.
- So, if you needed all 300 on one network segment you wouldn't have that with /24.
- (you can test this with the and operation easily.)



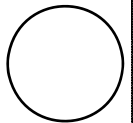
# SUPERNETTING

- You have to figure out what bits would allow 50 and 51 to be on the same segment:
- Convert to binary
- 00110010 and 00110011
- The only difference in these two is in the last bit. So, you can trick the system a bit (NPI)



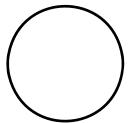
# CHANGES

- /23
- 255.255.254.0 is now a supernet
- X.x.11111110.0
- X.X.00110010.0
- X.x.00110010.0
- X.x.11111110.0
- X.x.00110011.0
- X.x.00110010.0 (same network 50)

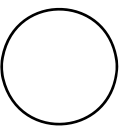


# So

- This means that all the layer 3 devices will treat 50 and 51 as the same network.
- Say you want to supernet 50 – 59 together.
- Convert
- 00110010
- 00110011
- 00110100
- 00110101
- 00110110
- 00110111
- 00111000
- 00111001
- 00111010
- 00111011
- Where do the bits differ?

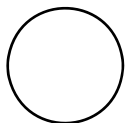


- 00110010
- 00110011
- 00110100
- 00110101
- 00110110
- 00110111
- 00111000
- 00111001
- 00111010
- 00111011



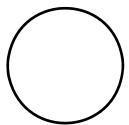
## 4 BITS REQUIRED

- 11110000 or 240 as the 3<sup>rd</sup> octet (/20)
- This means that some other networks are also included:
  - 00111100
  - 00111101
  - 00111110
  - 00111111
- Which you don't own but it shouldn't present a problem if those nets aren't inside your own network.



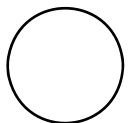
# MODERN SOLUTIONS

- You can simply NAPT as many addresses as you need.
- This is not always possible.
- Note 172.16.32.67/19 (255.255.224.0)
- Which breaks the lab into multiple networks in an orderly fashion



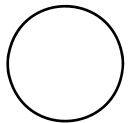
## SO AN EXAMPLE OF THE ENTIRE LAYER 1, 2, 3 PROCESS

- Network has a router at the border
  - 198.7.243.50 (outside)
  - 172.16.32.1 (inside)
  - Mac address is 1 on the inside and outside
    - NOTE: This is just to save typing mac address is 48 bits.



## 1,2,3 EXAMPLE CONT

- Next router outside is 198.7.243.11 with mac 2
- Inside the router is a switch with 4 servers connected (MAC is 3)
  - 172.16.32.2 – mac 4
  - 172.16.32.3 – mac 5
  - 172.16.32.4 – mac 6
  - 172.16.32.5 – mac 7



# 1,2,3 EXAMPLE CONT. OUTBOUND

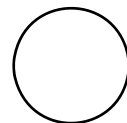
Data Packet
----------------

Packet from 4 going to  
the internet (cnn.com)

Data Packet	Source 4 Dest cnn
----------------	----------------------

Now the layer 2 info is  
added at the nic.

Data Packet	Source 4 Dest cnn	Source 6 Dest 3
----------------	----------------------	--------------------



## 1,2,3 CONT. OUTBOUND

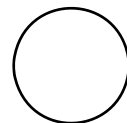
At the switch, the layer 2 is revised and the packet is sent onward

Data Packet	Source 4 Dest cnn	Source 3 Dest 1
-------------	----------------------	--------------------

When the router gets it, changes are made to both layer 2 and 3, assume PAT in use.

Data Packet	Source 198.7.243.50 Dest cnn	Source 1 Dest 2
-------------	------------------------------------	--------------------

The source port is then logged to remember which internal generated the packet



# LINUX BASICS

- NMAP – localhost only
- Nmap is a powerful tool if it works for you.
- You may also nmap ips in the 172.16.32.0 network
  - Nmap -p 80 localhost
- Wireshark
  - Packet sniffing will capture packets on the wire for analysis.

