Ryan J. Skelly RJS1070@wildcats.unh.edu CS/IT725 9/14/22

1.) Q1
   a.) 10000100.10110001.00001100.11000000/28
      i.) Prefix range 128.0.0.0 <-> 143.255.255.255
   b.) 00001010.00010100.00011110.00000000 ->
   00001010.00010100.00100101.11111111
      i.) Prefix with Subnet 14, addr: 10.20.30.0
   c.) 00000000.00000011.11111111.11111111
      i.) 255.252.0.0/21
   d.) 11111111.11111111.11111110.00000000
      i.) Prefix length /9
2.) Q2
   a.) Ifconfig
      i.) eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      ii.) inet 172.27.153.46 netmask 255.255.240.0 broadcast
      172.27.159.255
      iii.) inet6 fe80::215:5dff:fe0a:a494 prefixlen 64 scopeid 0x20<link>
      iv.) ether 00:15:5d:0a:a4:94 txqueuelen 1000 (Ethernet)
      v.) RX packets 2 bytes 434 (434.0 B)
      vi.) RX errors 0 dropped 0 overruns 0 frame 0
      vii.) TX packets 6 bytes 516 (516.0 B)
      viii.) TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
      ix.)
      x.) lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
      xi.) inet 127.0.0.1 netmask 255.0.0.0
      xii.) inet6 ::1 prefixlen 128 scopeid 0x10<host>
      xiii.) loop txqueuelen 1000 (Local Loopback)
   b.) Ifconfig
      i.) eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      ii.) inet 172.27.153.46 netmask 255.255.240.0 broadcast
      172.27.159.255
      iii.) inet6 fe80::215:5dff:fe0a:a494 prefixlen 64 scopeid 0x20<link>
      iv.) ether 00:15:5d:0a:a4:94 txqueuelen 1000 (Ethernet)
      v.) RX packets 2 bytes 434 (434.0 B)
      vi.) RX errors 0 dropped 0 overruns 0 frame 0
      vii.) TX packets 6 bytes 516 (516.0 B)
      viii.) TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
      ix.)
      x.) lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
      xi.) inet 127.0.0.1 netmask 255.0.0.0
      xii.) inet6 ::1 prefixlen 128 scopeid 0x10<host>
   c.) Ip route show/ip -6 route show
      i.) default via 172.27.144.1 dev eth0

ii.) 172.27.144.0/20 dev eth0 proto kernel scope link src 172.27.153.46
iii.)
iv.) fe80::/64 dev eth0 proto kernel metric 256 pref medium
v.) multicast ff00::/8 dev eth0 proto kernel metric 256 pref medium

d.) Arp
i.) No output although Im on WSL, and I havent attached the interface to a bridge to have it do ARP

3.) Then the two switches would communicate after time ‘q’ has past
4.) Q4
a.) First two is a DNS query request/response for the namespace www.apple.com, the next two is the main host using its newly discovered address of the secondary host to send subsequent echo requests(Which are all replied to)
b.) Im guessing mac.apple-dns.net(Which should be www.apple.com), although from what I know of ICMP, I dont believe hostnames are sent out for DNS queries as only v4 addresses are used
c.) A0:ce:c8:cd:b5:60
d.) There is a DNS server onlink that serves DNS responses to queries
e.) 34:e5:ec:b7:d0:49, 132.177.4.30
f.) About a second for each req/rep exchange
5.) Q5
a.) 9 hops
b.) I’d guess in California, because its in a completely different subnet
c.) Im currently on xfinity wifi, so definitely through there although Im not sure where else
d.) Yes, because there are always other packets being sent and received, which all have indeterminate lengths, leading to X amount of time passing comparatively, where X could be both positive and negative.
e.) They use different routers to distinguish where the packets should go, leading to different times and addresses showing up
f.) NOTE: I used powershells Tracert