# Internet Infrastructure and Protocols (COMP5311)

Solution to Assignment One

Each question carries 8 marks, unless stated otherwise.

Rocky K. C. Chang

1) A certain class B network `160.10.0.0/16` is subnetted according to the following structure. Note that variable length subnet mask is used and the subnetting is done with a certain pattern.

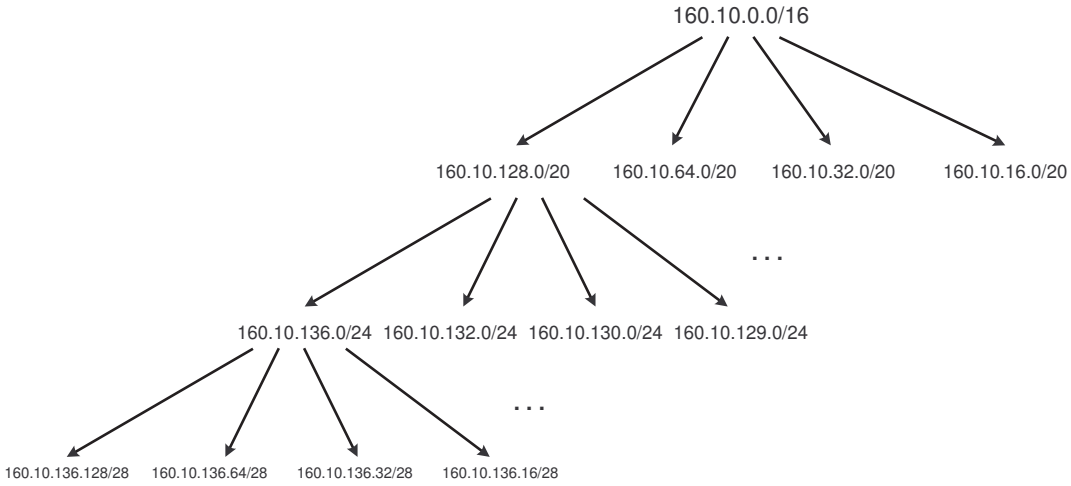


Fig. 1. The subnet structure of `160.10.0.0/16`.

a) (3 marks) Which subnet corresponds to the subnet-directed broadcast address `160.10.47.255`?
b) (3 marks) Which subnet corresponds to the subnet-directed broadcast address `160.10.130.31`?
c) (2 marks) Is `160.10.127.255` a subnet-directed broadcast address in this class B network?

**Solutions:**

a) (3 marks) `160.10.32.0/20`. This broadcast address does not possibly belong to the `/24` level, because none of them has a third byte of 47. Therefore, it also does not belong to the `/28` level. On the `/20` level, `160.10.32.0/20` has a broadcast address of `160.10.47.255`.
b) (3 marks) `160.10.130.16/28`. Since the last byte in the broadcast address is 31, it does not belong to the `/20` and `/24` levels. On the `/28` level, `160.10.130.16/28` has a broadcast address of `160.10.47.255`.
c) (2 marks) No, it is not. Since the last byte in the broadcast address is 255, it possibly belongs to the `/20` and `/24` levels. But it does not belong to any of the four subnets on the `/20` level. It also

does not belong to the /24 level, because none of the subnet numbers are 127.

2) In a ping of death attack, an attacker sends out an IP datagram that exceeds the maximum size of an IP datagram (65,535 bytes). Since this attack packet will be fragmented, the victim would not know the actual size of the original packet until it attempts to reassemble the fragments. However, some systems do not know how to handle an IP datagram of more than the maximum size; as a result, the attack causes them to hang and to reboot.

   The following is a tcpdump trace of such an attack. The fragmentation is performed according to an MTU of 1500 bytes, and the IP headers do not contain options.

```
17:26:11.013622 cslwin95 > arkroyal: icmp: echo request (frag 6144:1480@0+)
17:26:11.015079 cslwin95 > arkroyal: (frag 6144:1480@1480+)
17:26:11.016637 cslwin95 > arkroyal: (frag 6144:1480@2960+)
17:26:11.017577 cslwin95 > arkroyal: (frag 6144:1480@4440+)
17:26:11.018833 cslwin95 > arkroyal: (frag 6144:1480@5920+)
17:26:11.020112 cslwin95 > arkroyal: (frag 6144:1480@7400+)
17:26:11.021346 cslwin95 > arkroyal: (frag 6144:1480@8880+)
17:26:11.022641 cslwin95 > arkroyal: (frag 6144:1480@10360+)
17:26:11.023869 cslwin95 > arkroyal: (frag 6144:1480@11840+)
17:26:11.025140 cslwin95 > arkroyal: (frag 6144:1480@13320+)
17:26:11.026604 cslwin95 > arkroyal: (frag 6144:1480@14800+)
17:26:11.027628 cslwin95 > arkroyal: (frag 6144:1480@16280+)
17:26:11.028871 cslwin95 > arkroyal: (frag 6144:1480@17760+)
17:26:11.030100 cslwin95 > arkroyal: (frag 6144:1480@19240+)
17:26:11.031307 cslwin95 > arkroyal: (frag 6144:1480@20720+)
17:26:11.032542 cslwin95 > arkroyal: (frag 6144:1480@22200+)
17:26:11.033774 cslwin95 > arkroyal: (frag 6144:1480@23680+)
17:26:11.035018 cslwin95 > arkroyal: (frag 6144:1480@25160+)
17:26:11.036576 cslwin95 > arkroyal: (frag 6144:1480@26640+)
17:26:11.037464 cslwin95 > arkroyal: (frag 6144:1480@28120+)
17:26:11.038696 cslwin95 > arkroyal: (frag 6144:1480@29600+)
17:26:11.039966 cslwin95 > arkroyal: (frag 6144:1480@31080+)
17:26:11.041218 cslwin95 > arkroyal: (frag 6144:1480@32560+)
17:26:11.042579 cslwin95 > arkroyal: (frag 6144:1480@34040+)
17:26:11.043807 cslwin95 > arkroyal: (frag 6144:1480@35520+)
17:26:11.046276 cslwin95 > arkroyal: (frag 6144:1480@37000+)
17:26:11.047236 cslwin95 > arkroyal: (frag 6144:1480@38480+)
17:26:11.048478 cslwin95 > arkroyal: (frag 6144:1480@39960+)
17:26:11.049698 cslwin95 > arkroyal: (frag 6144:1480@41440+)
17:26:11.050929 cslwin95 > arkroyal: (frag 6144:1480@42920+)
17:26:11.052164 cslwin95 > arkroyal: (frag 6144:1480@44400+)
17:26:11.053398 cslwin95 > arkroyal: (frag 6144:1480@45880+)
17:26:11.054685 cslwin95 > arkroyal: (frag 6144:1480@47360+)
17:26:11.056347 cslwin95 > arkroyal: (frag 6144:1480@48840+)
17:26:11.057313 cslwin95 > arkroyal: (frag 6144:1480@50320+)
17:26:11.058357 cslwin95 > arkroyal: (frag 6144:1480@51800+)
17:26:11.059588 cslwin95 > arkroyal: (frag 6144:1480@53280+)
17:26:11.060787 cslwin95 > arkroyal: (frag 6144:1480@54760+)
17:26:11.062023 cslwin95 > arkroyal: (frag 6144:1480@56240+)
17:26:11.063247 cslwin95 > arkroyal: (frag 6144:1480@57720+)
17:26:11.064479 cslwin95 > arkroyal: (frag 6144:1480@59200+)
17:26:11.066252 cslwin95 > arkroyal: (frag 6144:1480@60680+)
17:26:11.066957 cslwin95 > arkroyal: (frag 6144:1480@62160+)
17:26:11.068220 cslwin95 > arkroyal: (frag 6144:1480@63640+)
17:26:11.069107 cslwin95 > arkroyal: (frag 6144:398@65120)
```

   Answer the following questions concerning the trace.

   a) (1 mark) What does the number immediately before "@" represent?

b) (1 mark) What does the number immediately after "@" represent?
c) (1 mark) What does the "+" symbol represent?
d) (1 mark) What is the number 6144 referred to?
e) (4 marks) What is the size of the original IP packet before fragmentation?

**Solutions:**

a) (1 mark) The size of the IP packet payload
b) (1 mark) The fragment offset
c) (1 mark) The more fragment flag
d) (1 mark) The IP packet's ID
e) (4 marks) The payload of the original IP packet = $44 \times 1480 + 398$ = 65,518 bytes. Therefore, the original IP packet size = 65,518+20 = 65,538 bytes $>$ 65,535 bytes.

3) Consider a class B network `158.132.0.0` which is subnetted with a subnet mask of `255.255.255.0`. Moreover, a host with IP address `158.132.1.1` is compromised in that an attack program was installed in that machine. Discuss the effect of the attack if the attack program in that machine sends out an ICMP echo request message (ping) with the following source and destination addresses. Note that the source addresses in ping messages are spoofed (i.e., not equal to `158.132.1.1`). Assume the followings:

- All routers inside the network turn off the support for subnet-directed and all-subnet-directed IP broadcasts, i.e., drop those packets.
- All nodes (hosts and routers) must reply with an ICMP echo reply message when receiving an ICMP echo request message.
- When forwarding a packet, a router only examines the destination IP address, but not the source IP address.
- The destination addresses of the ping messages belong to hosts, but not to routers.

a) (3 marks) Source address = `158.132.1.100` and destination address = `255.255.255.255`.
b) (3 marks) Source address = `158.132.2.100` and destination address = `255.255.255.255`.
c) (2 marks) Source address = `158.132.2.100` and destination address = `158.132.2.255`.

**Solutions:**

a) (3 marks) All nodes on subnet 1 will receive this ICMP request message and send reply messages to `158.132.1.100`. If the number is large enough, the victim host will be overwhelmed by these packets.
b) (3 marks) All nodes on subnet 1 will receive this ICMP request message and send reply messages to `158.132.2.100`. Therefore, all the reply messages will be forwarded to subnet 2. If the number is large enough, the router that is responsible for forwarding these packets or the victim host will be overwhelmed by these packets.
c) (2 marks) Since the ICMP request message is a subnet-directed packet, the router will drop it. As a result, this attack does not have any impact on the network.

4) An IP packet with a total length of 3000 bytes and a header length of 20 bytes is to be fragmented into 1500-byte IP packets.

a) (3 marks) Write down total length, the offset values, and the M-bit value in each fragment.

b) (2 marks) If the total length is changed to 3100 bytes, how would your answers be different from (a)?

c) (3 marks) If the header length becomes 40 bytes (and the IP packet's total length remains 3000 bytes), how would your answers be different from (a)?

**Solutions:**

a) (3 marks) The first fragment: total length = 1500, offset = 0, and the `M` bit = 1. The second fragment: total length = 1500, offset = 1480/8 = 185, and the `M` bit = 1. The third fragment: total length = 40, offset = (2x1480)/8 = 370, and the `M` bit = 0.

b) (2 marks) The only change is the total length in the last fragment which becomes 140.

c) (3 marks) The first fragment: total length = 1496, offset = 0, and the `M` bit = 1. The second fragment: total length = 1496, offset = 1456/8 = 182, and the `M` bit = 1. The third fragment: total length = 88 (40+(3000-40-1456-1456)), offset = 364, and the `M` bit = 0.