

CS450 Fall 2009 Midterm 1

Name: _____

1. (5 points) Consider the following HTTP request, issued using telnet:

```
telnet www.somedomain.com 80
GET / HTTP/1.0
(empty line here)
```

Say the domain name 'www.somedomain.com' points to address 12.13.14.15. Another domain name, 'www.someother.com' also points to 12.13.14.15.

Say the owner of this web server wants to display different pages for the two different domains, even though they lead to the same IP address, and consequently to the same server. This is a very common occurrence today.

Is it possible to do what the owner wants, given a request issued as above? If so, how does the web server distinguish between the two? If it is not possible, why not? How does this work when using a modern web browser? If you don't know, suggest a feasible solution.

It's not possible with this request.

Once the request reaches the server, there's no way to distinguish the two domains. DNS is resolved at the client.

In HTTP 1.1, web browsers specify the desired domain using a host: header.

CS450 Fall 2009 Midterm 1

Name: _____

2. (7 points) Consider the following command, issued on a unix terminal:

```
echo Hello | nc 128.30.76.80 1050
```

Give a complete listing, in order, of all packets exchanged (by TCP and above, no need to worry about lower layer details) as a result of this command. For each packet, list whether it is outgoing or incoming, and its content/purpose very briefly.

Several different variations are possible - pick your favorite, and describe only that one. You may assume that there are no packet losses.

- syn

- synack (establish connection)

- ack

- data (hello)

- ack

- fin

- finack (tear down connection)

- ack

3. (a) (1 point) What is the main purpose of the domain name service (DNS)?

resolve hostname into ip address

- (b) (1 point) There exist many types of DNS resource records, including A, NS, CNAME, MX and PTR. What is the purpose of the MX record type?

identify responsible incoming mail server

- (c) (2 points) There are 12 "virtual" root servers, each of which is actually replicated in several locations. However, even if a powerful network conspiracy was to somehow disable every root server, including all their clones, most Internet users wouldn't notice a thing, at least for a while. Why is that, and what users *would* be immediately affected?

it keeps working due to caching
non-cached values would be immediately affected

- (d) (5 points) You just bought an island in the north atlantic, and decided to start your own country, 'Lilliput'. As a first order of business, you install a name server on your island, and request the country-code top-level domain name 'lp' from ICANN.

The country is small, and only has one company (Kelp Enterprises). In your country's TLD, the **only** resource record is an NS record for 'ke.lp'.

```
ke NS ns.kelp.com
```

Now let someone issue a DNS lookup for `www.ke.lp`. Assume this someone has never looked up any of Kelp Enterprises' hosts before. Describe the sequence of DNS requests that are issued by the DNS resolver as a result of this lookup. For each request, show the recipient and the request content. State any assumptions you make.

ask root server for TLD lp
ask In Tl D for ke In

4. Consider the following pseudocode for the `send()` function of a reliable transport layer.

```

timeout = 1 second
reliable_send (socket, packet) {
  do {
    unreliable_send(socket, packet)
  } while(!receive_ack(socket, timeout));
}

```

- (a) (2 points) Assuming there are no packet losses, very briefly describe a scenario in which this protocol would achieve close to optimal bandwidth utilization.

very short rtt, very small bandwidth-delay product

- (b) (2 points) In your scenario from part (a), let approximately 1% of packets be lost due to transmission errors, resulting in very poor throughput. How can we change `reliable_send()` to avoid the drop in throughput? Give a high-level description, not pseudocode.

adapt the timeout period to the rtt + deviation

- (c) (1 point) Still using the same transport layer, let the sender be in Chicago, and the receiver be your boss in New Zealand, let the maximum capacity of the route between the two be 10 Mbps, with no packet losses, no competing traffic, and a maximum packet size of 1250 bytes. Your boss is unhappy with his download speeds, and tells you to upgrade to a 1 Gbps connection. How much faster will your pointy-haired kiwi boss get his spreadsheets?

no improvement

- (d) (4 points) What common transport protocol technique could you have used instead to improve the Chicago-New Zealand download speeds? Estimate the throughput you would achieve with and without this technique. If you can't calculate the throughput based on the parameters given above, then give a symbolic expression.

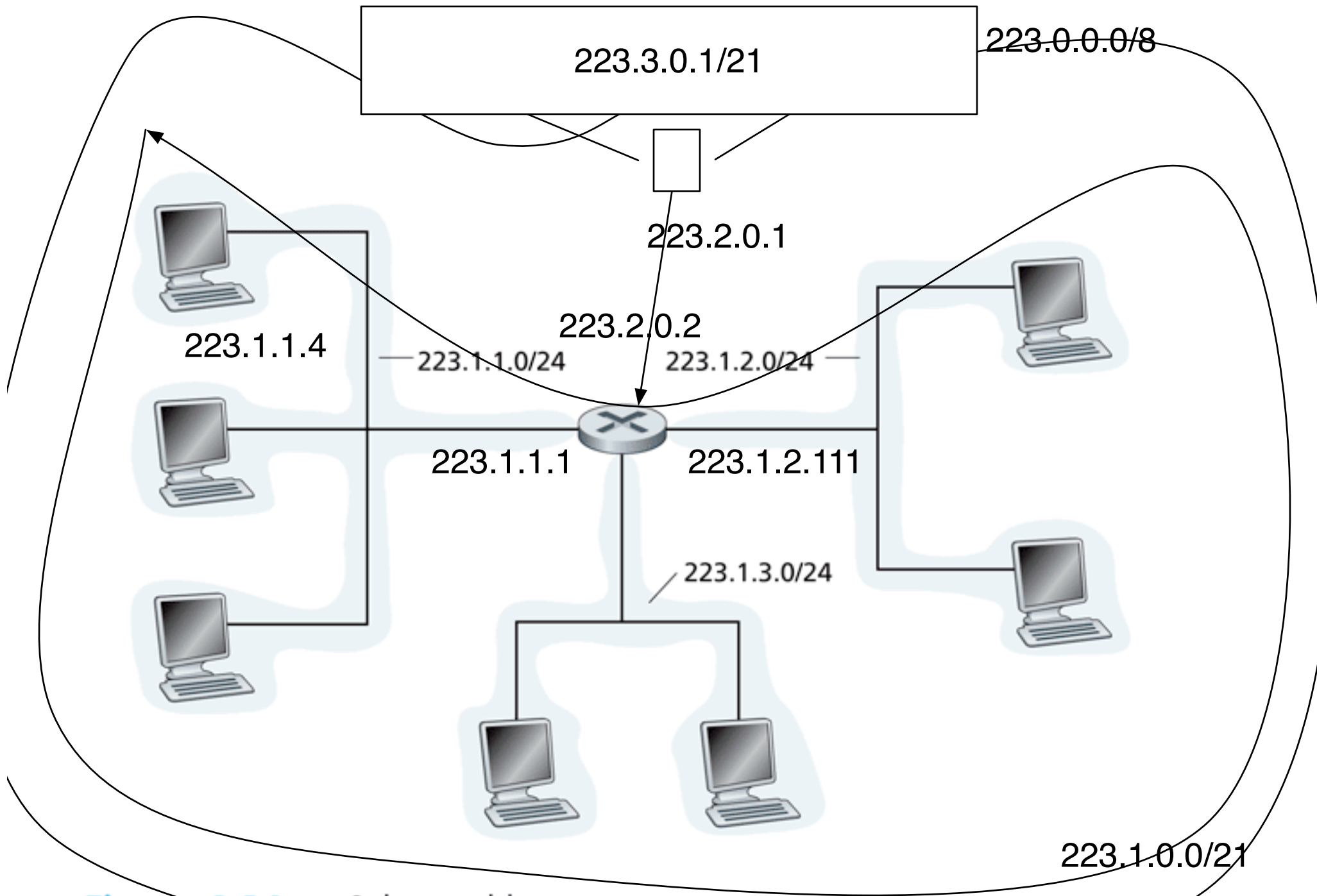


Figure 4.16 ♦ Subnet addresses