

Quiz 2 – (10 points)

Your Name:

9/20/2017

1. **(4 points total)** Whenever a local DNS resolver queries its default DNS server for *host1.myserver.com*, it receives a response with a TTL value of 120.

(2 Points) Assuming that the resolver receives on average 10 queries per minute for *host1.myserver.com* from local clients, what is then the approximate average query rate for *host1.myserver.com* (in queries per minute) from the resolver to its DNS server?

A TTL value of 120 means that the resolver can cache the response it received to its original query for 120 secs, or 2 mins. This means that the query rate for host1.myserver.com from the DNS resolver is approximately 1 query every 2 mins, or on average 0.5 query per minute.

(2 point) How would the answer change if the resolver instead received 20 queries per minute for *host1.myserver.com*?

The answer would be essentially unchanged if the query rate was 20 queries per minute for host1.myserver.com. The resolver still only sends one query every two minutes, i.e., when the cached answer expires. The reduction in query rate would, however, now be by a factor 40 rather than 20.

2. **(4 points total)** A host runs the following code snippet

```
ServerSocket sock = new ServerSocket(5576,0,InetAddress.getByName(null));
InetAddress x = sock.getInetAddress();
String hostIP = x.getHostAddress();
System.out.println("x: " + hostIP);
Socket connsock = sock.accept();
```

(2 points) What would you expect to see in the printout statement for *hostIP*?

You would expect to see one of the host's standard local addresses, typically 127.0.0.1.

In general, if the host is null then an InetAddress representing an address of the loopback interface is returned. See RFC 3330 section 2 and RFC 2373 section 2.5.3.

(2 points) What functionality does `sock.accept` performs? Be as comprehensive as possible in characterizing the functionality

Socket.accept is waiting for incoming connections on port 5576 from local clients (same machine)

3. **(2 points total)** Consider now that the host also runs the following code snippet

```
Socket sock = new Socket(InetAddress.getByName(null),5576);
int port = sock.getLocalPort();
System.out.println("port: " + port);
```

What would you expect to see in the printout statement for `port` and what does it correspond to?

The code sets up a local connection to destination port 5576 on the host itself. The `sock.getLocalPort()` statement retrieves the random source port number that the operating system assigned to the socket when it was first created.