



Department of Electrical Engineering and Computer Science

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

6.033 Computer Systems Engineering: Spring 2002

Handout 31 - Quiz 2

All problems on this quiz are multiple-choice questions. In order to receive credit you must fill in the blank(s) or mark the correct answer or answers for each question. You have 50 minutes to answer this quiz.

Write your name on this cover sheet AND at the bottom of each page of this booklet.

Some questions may be much harder than others. Read them all through first and attack them in the order that allows you to make the most progress. If you find a question ambiguous, be sure to write down any assumptions you make. Be neat. If we can't understand your answer, we can't give you credit!

THIS IS AN OPEN BOOK, OPEN NOTES QUIZ.

CIRCLE your recitation section:

- | | | | |
|--------------|--------------------------|-----------------------------|-------------------------------|
| 10:00 | 1. Balakrishnan/Chambers | 13. Morris+Kaashoek/Gnawali | 11. Amarasinghe/Bhattacharyya |
| 11:00 | 2. Balakrishnan/Salz | 6. Morris+Kaashoek/Chambers | 12. Amarasinghe/Gnawali |
| 12:00 | 5. Ernst/Salz | 14. Witchel/Bauer | |
| 1:00 | 8. Ernst/Yip | 3. Leiserson/Freedman | 10. Teller/Bauer |
| | 7. Saltzer/Vandiver | | |
| 2:00 | 9. Saltzer/Freedman | 4. Leiserson/Vandiver | 15. Teller/Bhattacharyya |

Do not write in the boxes below

1-4 (xx/40)	5-12 (xx/60)	Total (xx/100)

Name:

I Reading questions

1. [10 points]: According to the authors of NFS (Reading #12), the NFS protocol is stateless. Which are true statements about a stateless protocol?

(Circle ALL that apply)

- A. The client doesn't store any information about file system operations it has in progress.
- B. If the client crashes and restarts, the server doesn't need to be informed about the crash.
- C. Every network protocol request contains all the information needed to carry out that request, without relying on anything other than the contents of file systems on disk.
- D. If the server fails and restarts between two protocol requests, the server doesn't have to contact any client in order to process the second protocol request.

2. [10 points]: Which are true statements about network address translators (K. Egevang and P. Francis. The IP Network Address Translator (NAT). RFC 1631):

(Circle ALL that apply)

- A. NATs break the layering abstraction of the network model of Chapter 4.
- B. NATs address the problem that the Internet is in practice running out of IP addresses.
- C. NATs constrain new applications because an application that includes the IP address of its host machine in the payload of a message cannot be guaranteed that the recipient is able to contact that IP address.
- D. NATs improve the reliability of TCP connections that go through them by retransmitting lost packets.

3. [10 points]: What design decisions contribute to the precision of Google (reading #13) when answering queries?

(Circle ALL that apply)

- A. Propagating anchor text.
- B. Storing the offset of every occurrence of each word in an HTML document.
- C. Computing a PageRank.
- D. Storing information about the typesetting of words.
- E. Collecting flocks of pigeons.

4. [10 points]: The class notes discuss both capabilities and access control lists as mechanisms for authorization. Which of the following statements are true?

(Circle ALL that apply)

- A.** A capability system associates a list of object references with each principal, indicating which objects the principal is allowed to use.
- B.** An access control list system associates a list of principals with each object, indicating which principals are allowed to use the object.
- C.** Revocation of a particular access permission of a principal is more difficult in an access control list system than in a capability system.
- D.** Protection in the UNIX file system is based on capabilities only.

II Peer-to-peer networking

Ben Bitdiddle is disappointed that the music industry is not publishing his CD, a rap production based on 6.033. Ben is convinced there is a large audience for his material. Having no alternative, he turns his CD into a set of MP3 files, the digital music standard understood by music playing programs, and publishes the songs through Gnutella.

Gnutella is a distributed file sharing application for the Internet. A user of Gnutella starts a Gnutella node, which presents a user interface to query for songs, talks to other nodes, and makes files from its local disk available to other remote users.

The Gnutella nodes form an *overlay* network on top of the existing Internet. The nodes in the overlay network are Gnutella nodes and the links between them are TCP connections. When a node starts it makes a TCP connection to various other nodes, which are connected through TCP connections to other nodes. When a node sends a message to another node, the message travels over the connections established between the nodes. Thus, a message from one node to another node travels through a number of intermediate Gnutella nodes.

To find a file, the user interface on the node sends a query (e.g., “6.033 rap”) through the overlay network to other nodes. While the search propagates through the Gnutella network, nodes that have the desired file send a reply, and the user sees a list filling with file names that match the query. Both the queries and their replies travel through the overlay network.

The user then selects one of the files to download and play. The user’s node downloads the file *directly* from the node that has the file, instead of through the Gnutella network.

The format of the header of a Gnutella message is:

Message ID	Type	TTL	Hops	Length
16 bytes	1 byte	1 byte	1 byte	4 bytes

The header is followed by the payload, which is `Length` bytes long.

The main message types in the Gnutella protocol are:

- **Ping:** A node finds additional Gnutella nodes in the network using Ping messages. A node wants to be connected to more than one other Gnutella node to provide a high degree of connectivity in the case of node failures. Gnutella nodes are not very reliable, because a user might turn off his machine running a Gnutella node at any time. Ping messages have no payload.
- **Pong:** A node responds by sending a Pong message via the Gnutella network whenever it receives a Ping message. The Pong message has the same `MessageID` as the corresponding `Ping` message. The payload of the Pong message is the IP address of the node that is responding to the Ping Message.
- **Query:** Used to search the Gnutella network for files; its payload contains the query string that the user typed.

Name:

- **QueryHit:** A node responds by sending a QueryHit message via the Gnutella network if it has a file that matches the query in a Query message it receives. The payload contains the IP address of the node that has the file, so that the user's node can connect directly to the node that has the song and download it. The QueryHit message has the same `MessageID` as the corresponding Query message.

(The Gnutella protocol also has a Push message to deal with firewalls and network address translators, but we will ignore it.)

In order to join the Gnutella network, the user must specify an IP address of one or more existing nodes. The node connects to those nodes using TCP. Once connected, the node uses Ping messages to find more nodes (more detail below), and then directly connects to some subset of the nodes that the Ping message found.

The Gnutella protocol is a kind of broadcast protocol. A node receiving a Ping or a Query message forwards that message to all the nodes it is connected to, except the one from which it received the message. A node decrements the `TTL` field and increments the `Hops` field before forwarding the message. If after decrementing the `TTL` field, the `TTL` field is zero, the node does not forward the message at all. The `Hops` field is set to zero by the originating user's node.

To limit flooding and to route Pong and QueryHit messages, a node maintains a message table, indexed by `MessageID` and `type`, with an entry for each message seen recently. The entry also contains the IP address of the Gnutella node that forwarded the message to it. The message table is used as follows:

- If a Ping or Query message arrives and there is an entry in the message table with the same `message ID` and `type`, then the node discards that message.
- For a QueryHit or Pong message for which there is a corresponding Query or Ping entry with the same message ID in the message table, then the node forwards the QueryHit or Pong to the node from which the Query or Ping was received.
- If the corresponding Query or Ping message doesn't appear in the table, then the node discards the QueryHit or Pong message.
- Otherwise, the node makes a new entry in the table, and forwards the message to all the nodes it is connected to, except the one from which it received the message.

5. [5 points]: Assume one doesn't know the topology of the Gnutella network or the propagation delays of messages. According to the protocol, a node should forward all QueryHit messages for which it saw the corresponding Query message back to the node from which it received the Query message. If a node wants to guarantee that rule, when can the node remove the Query entry from the message table?

(Circle BEST answer)

- A. Never, in principle, because a node doesn't know if another QueryHit for the same Query will arrive.
- B. Whenever it feels like, since the table is not necessary for correctness. It is only a performance optimization.
- C. As soon as it has forwarded the corresponding QueryHit message.
- D. As soon as the entry becomes the least recently used entry.

Both the Internet and the Gnutella network form graphs. For the Internet, the nodes are routers and the edges are links between the routers. For the Gnutella network, the nodes are Gnutella nodes and the edges are TCP connections between the nodes. The shortest path in a graph between two nodes A and B is the path that connects A with B through the fewest number of nodes.

6. [5 points]: Assuming a stable Internet and Gnutella network, is the shortest path between two nodes in the Gnutella overlay network always the shortest path between those two nodes in the Internet?

(Circle BEST answer)

- A. Yes, because the Gnutella network uses the Internet to set up TCP connections between its nodes.
- B. No, because TCP is slower than IP.
- C. Yes, because the topology of the Gnutella network is identical to the topology of the Internet.
- D. No, because for node A to reach node B in the Gnutella network, it might have to go through node C, even though there is a direct, Internet link between A and B.

7. [10 points]: Which of the following relationships always hold? ($TTL(i)$ and $Hop(i)$ are the values of TTL and HOP fields respectively after the message has traversed i hops)?

(Circle ALL that apply)

- A. $TTL(0) = Hops(i) - TTL(i)$
- B. $TTL(i) = TTL(i - 1) - 1$, for $i > 0$
- C. $TTL(0) = TTL(i) + Hops(i)$
- D. $TTL(0) = TTL(i) \times Hops(i)$

8. [5 points]: Ben observes that both Ping and Query messages have the same forwarding rules, so he proposes to delete Ping and Pong messages from the protocol and to use a Query message with a null query (which requires a node to respond with a QueryHit message) to replace Ping messages. Is Ben's modified protocol a good replacement for the Gnutella protocol?

(Circle BEST answer)

- A. Yes, good question. Beats me why the Gnutella designers included both Ping and Query messages.
- B. No, a Ping message will typically have a lower value in the TTL field than Query message when it enters the network
- C. No, because Pong and QueryHit messages have different forwarding rules.
- D. No, because there is no way to find nodes using Query messages.

9. [10 points]: Assume that only one node S stores the song "6.033 rap," and that the query enters the network at a node C . Further assume TTL is set to a value large enough to explore the whole network. Gnutella can still find the song "6.033 rap" despite the failures of some sets of nodes (either Gnutella nodes or Internet routers). On the other hand, there are sets of nodes whose failure would prevent Gnutella from finding the song. Which of the following are among the latter sets?

(Circle ALL that apply)

- A. any set containing S
- B. any set containing a single node on the shortest path from C to S
- C. any set of nodes that collectively disconnects C from S in the Gnutella network
- D. any set of nodes that collectively disconnects C from S in the Internet

10. [10 points]: To which of the following attacks is Gnutella vulnerable (i.e., an attacker can implement the described attack)?

(Circle ALL that apply)

- A. A single malicious node can always prevent a client from finding a file by dropping QueryHits.
- B. A malicious node can respond with a file that doesn't match the query.
- C. A malicious node can always change the contact information in a QueryHit message that goes through the node, for example, misleading the client to connect to it.
- D. A single malicious node can always split the network into two disconnected networks by never forwarding Ping and Query messages.
- E. A single malicious node can always cause a Query message to circle forever in the network by incrementing the TTL field (instead of decrementing it).

11. [5 points]: Ben wants to protect the content of a song against eavesdroppers during downloads. Ben thinks a node should send $RC4(k, song)$ as the download, but Alyssa thinks the node should send $SHA-1(song)$. Who is right?

(Circle ALL that apply)

- A. Ben is right, because no one can compute $song$ from the output of $SHA-1(song)$, unless they already have $song$.
- B. Alyssa is right, because even if one doesn't know the shared-secret key k anyone can compute the inverse of the output of $RC4(k, song)$.
- C. Alyssa is right, because $SHA-1$ doesn't require a key and therefore Ben doesn't have to design a protocol for key distribution.
- D. Both are wrong, because RSA is obviously the right choice, since sealing with RSA is computationally more expensive than $SHA-1$ and $RC4$.

Ben is worried that an attacker might modify the "6.033 rap" song. He proposes that every node that originates a message signs the payload of a message with its private key. To discover the public keys of nodes, he modifies the Pong message to contain the public key of the responding node along with its IP address. When a node is asked to serve a file it signs the response (including the file) with its private key.

12. [10 points]: Which attacks does this scheme prevent?

(Circle ALL that apply)

- A. It prevents malicious nodes from claiming they have a copy of the "6.033 rap" song and then serving music written by Bach.
- B. It prevents malicious nodes from modifying Query messages that they forward.
- C. It prevents malicious nodes from discarding Query messages.
- D. It prevents nodes from impersonating other nodes and thus prevents them from forging songs.
- E. None. It doesn't help.

End of Quiz 2