Exam - Midterm

Due: 9pm, 17 Mar 2015

Closed Book. Maximum points: 100

NAME:

Tapestry [26 + 8 pts]

a. For an object with id x and a set of Tapestry nodes N with ids $n_1 \ldots n_k$, explain how Tapestry determines the root node for the object. [10 pts]

b. Suppose our Tapestry has the following four nodes: 4046, 4049, 0319, 2319. For each of the following IDs, which node is the root? [6 pts]

4006: 4175: 1999: 2999: 3116: 3999:

c. In Tapestry, several nodes can publish the same object. In a large-scale production Tapestry deployment, the more nodes there are publishing an object, the more likely it is that we can access the object with lower latency. Explain how Tapestry combines caching of locations and RTT measurements to achieve this. [10 pts]

BONUS Differently from Tapestry, in Chord, given a fixed set of nodes in the ring, a node has only one choice in the next hop at a particular position in the finger table: For the table at node i, the j^{th} entry in the table refers to the smallest-numbered node that exceeds i by at least $2^{j-1} \mod 2^m$. How can you change Chord so that you can still have O(log N) routes, but with the possibility of multiple choices of next hops? [BONUS: 8 pts]

Superfish [25 pts] In 2015 it became aparent that Lenovo shipped software with its computers, developed by a company called Superfish, that effectively mounted a man-in-the-middle attack on all SSL/TLS connections made by the users. The basis of the attack is that the computers shipped with an extra trusted root Certificate Authority that is a self-signed certificate. This means that the browser will now trust any certificates that are signed by this root certificate. The Superfish software also has the ability to monitor and intercept all connections made by the user, and has the private key corresponding to the public key of the fake CA (this means that it can sign new certificates on the fly).

a. The original goal of the attack is to insert ads on pages visited by the user. Explain why, under normal circumstances, even if a program can monitor and intercept the communications between the browser and the Internet, it cannot alter the contents of an HTTPS page. [8 pts]

b. With your knowledge of TLS/SSL, explain how the Superfish program described above can now insert ads on a page that is served over HTTPS, while making the Browser still show to the user the padlock that indicates a secure connection. [9 pts]

c. Since the private key corresponding to the Superfish software had to be present in the software itself, it leaked on the Internet, meaning that anyone can now create certificates signed by the fake root certificate. What is the extra implication of this for the affected Lenovo users? [8 pts]

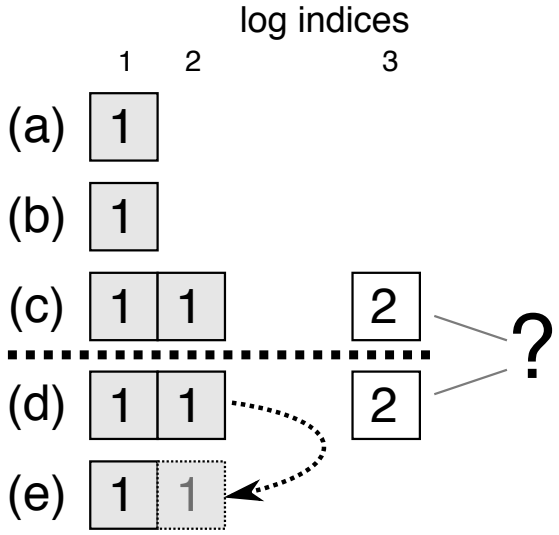
Consistent Snapshots[25 pts] You are building the software for a bank with three branches, one in Providence, one in Denver, and the other in San Francisco. They conciliate the values of the only account they hold by sending messages among themselves. The diagram below shows the execution of an operation that changed the balance of the account. Knowing that machines can crash (assume messages never fail), you decided to add snapshotting capabilities to the system: each node takes periodic snapshots, and if disaster strikes they can restart from the latest snapshot.

a. If you have snapshots a and g, which of the snapshots in process DEN will produce a consistent global snapshot? Briefly justfity your answer. [8 pts]

b. Can snapshots b, c, and f be part of a consistent snapshot? Briefly justify your answer. [8 pts]

c. The Chandy-Lamport algorithm for consistent snapshots we saw in class is a big improvement over the haphazard situation above, and works by sending marker messages along the graph of processes that communicate. The algorithm assumes FIFO channels. Assuming that on the example above the only channels are PVD→DEN and DEN→SFO, and that PVD starts a snapshot at point a, give an example of how the algorigthm can go wrong if the FIFO assumption is violated. [9 pts]

Raft [24 pts] The figure below represents the logs of a 5-node Raft cluster. The squares represent entries of the log, and the numbers inside the squares represent the *term* of the entry (not the actual command).



a. Given the rules for leader election in Raft, if S_1 were to fail now, for each one of the other nodes, **say whether they could or could not be elected leaders**. **If so, say who would vote for them, and if not, say why not.** [8 pts]

S_2 Electable? (Yes) (No)

S_3 Electable? (Yes) (No)

S_4 Electable? (Yes) (No)

S_5 Electable? (Yes) (No)

b. Recall that in Raft once a leader decides to commit an entry, then this entry is guaranteed to be present in all future leaders' logs, forever. Given this, **for each entry** in $S_1's$ log, **say whether or not it is safe to be committed and why/why not**. Refer to the entries by their log index. [7 pts]

Entry 1 Safe? (Yes) (No)

Entry 2 Safe? (Yes) (No)

Entry 3 Safe? (Yes) (No)

Entry 4 Safe? (Yes) (No)

Entry 5 Safe? (Yes) (No)

Entry 6 Safe? (Yes) (No)

Entry 7 Safe? (Yes) (No)

c. Raft has an interesting use of randomized timers. **What would be the effect in the algorithm if you removed the randomization?** [9 pts]