Reliability of Distributed Systems

Ex2, due on December 2

version 2

1 Mobile Adversaries

Synchronous model: message arrive after at most Δ time. There are n servers and at least two clients. Clients may have omission failures.

Mobile fault model: the environment holds a variable *budget* that is initially set to f. The adversary can send $<$ corrupt i $>$ request and $<$ uncorrupt i $>$ request to the environment.

When a $<$ corrupt i $>$ request arrives to the environment and *budget* > 0, then the environment reduces *budget* by one and after 2Δ time allows the adversary to control party i.

When a $<$ uncorrupt i $>$ request arrives to the environment then if the adversary was controlling i then the environment stops allowing the adversary to control party i, the environment sends a $<$ restart $>$ signal to party i, and after 10Δ time it increases *budget* by one.

A protocol in this model is allowed to implement an event handler for the $<$ restart $>$ signal for each party.

During any duration in which the adversary is controlling party i it can cause it to fail:

1. In the **Mobile crash model**, it can cause the party to crash at any point of execution. Note that a party can crash and restart and crash again, etc
2. In the **Mobile omission model**, it can cause the party any type of omission for any message sent or receives by the party. Again a party can have periods on omission faulty, then restart, then again a period of omission, etc

1.1 Question 1: Mobile Crash

Show a protocol solving State Machine Replication for any $f < n$ Mobile crash failures that uses a stable leader. Prove safety and liveness.

1. For safety, prove that your protocol behaves the same as an ideal State Machine that never fails. Hint: the safety proof should probably contain a proof by induction showing that once a certain event is reached, all later leaders will propose the same value. Second Hint: your proof should carefully address the cases where a party restarts (for example, what if it missed many commands). You can assume that there is no bound on message size and execution takes zero time.
2. For Liveness, you need to show that commands sent from non-faulty clients will eventually be executed. State and prove the following bounds:
   (a) State and prove a worst case bound for the time it may take a non-faulty client to receive a response for its request. This should be a function of Δ and f. Consider two cases separately: $f < 6$ and $f > 15$.
   (b) During a duration in which the leader is non-faulty, state and prove a worst case bound for the time it may take a non-faulty client to receive a response for its request is just a function of Δ.

(c) Bonus: For the case of $f > 15$ state a prove a bounded on the expected time it may take a non-faulty client to receive a response for its request. This should be a function of Δ.

1.2 Question 2: Mobile Omission

Show a protocol solving State Machine Replication for any $f < n/2$ Mobile omission failures that uses a stable leader. Prove safety and liveness.

1. For safety, prove that your protocol behaves the same as an ideal State Machine that never fails. Hint: the safety proof should probably contain a proof by induction showing that once a certain event is reached, all later leaders will propose the same value. Second Hint: your proof should carefully address the cases where a party restarts (for example, what if it missed many commands). You can assume that there is no bound on message size and execution takes zero time.

2. For Liveness, you need to show that commands sent from non-faulty clients will eventually be executed. State and prove two bounds:

   (a) State and prove a worst case bound for the time it may take a non-faulty client to receive a response for its request. This should be a function of Δ and f. Consider two cases separately: $f < 6$ and $f > 15$.

   (b) During a duration in which the leader is non-faulty, state and prove a worst case bound for the time it may take a non-faulty client to receive a response for its request is just a function of Δ.

   (c) Bonus: For the case of $f > 15$ state a prove a bounded on the expected time it may take a non-faulty client to receive a response for its request. This should be a function of Δ.