CSE 535

Project Pseudocode

Tushar Mohandas (111464344)

Soumyadeep Chakraborty (111425035)

Fall 2017

Design doc

Keys:

1) **publicKeys** is a map indexed by the source (ie client | Olympus | replicas).

2) **clients** is a set.

3) **currentConfig** is an object of properties which capture the list of **replicas**, the **head**, the **tail** and **"t"** which represents the no of faults that can be tolerated by the system.

4) **keysMessage** is an object containing the **public keys** of all replicas, a **particular client** and **Olympus**.

Reconfiguration request message:

It is an object containing a proof of misbehavior which can further take the following forms:

1) Proof of a signature mismatch in the list of order proofs which is a result of the order proof containing an invalid signature. In this case **misbehaviorProof.signatureMismatch** = true.

2) Proof of multiple operations for a single slot in an order proof in which case **misbehaviorProof.conflictingSlot** contains the conflicting slot.

3) Proof of a signature mismatch in the result proof: In this case, **misbehaviorProof.invalidResultProof** contains the result proof which is invalid.

Reconfiguration quorum handling:

1) **replies** is used to track incoming **correct wedge responses**. This is further used by the **quorumSatisifed** condition variable.

2) **slotOperationPairs** : as the name suggests it’s a tuple of **slot** and **operation**.

Replica: Some properties:

1. **state** : immutable (on error detection), active (default)
2. **object** : The object being replicated. For simplicity, we have assumed that there is a single shared object between multiple clients.
3. **lastSignedSlot** : The last slot for which this replica has signed an order statement. This ensures that this replica is not going to skip a slot before signing a new order statement for a new slot. (ie prevents holes)
4. **pre** : The previous replica in the chain.
5. **next** : The next replica in the chain.
6. **history :** A list of all order proofs for all slot and operation pairs maintained at this replica.
7. **privateKey**
8. **publicKeys –** of all players in the system
9. **resultCache –** A map of cached result shuttles which are indexed by the operation.

Update shuttle:

1. **slot**
2. **operation**
3. **orderProof** :
   1. **orderStatements** is a set of signed order statements made by each replica upstream of the current replica.
   2. **replica** – the current replica.
4. **resultProof** – A tuple containing the **operation** and a **hash** of the calculated result at this replica.

Result shuttle: The update shuttle when it is routed backwards from the tail -> head. There is no change in structure.

Wedge response (a **wedged** message): It has the following properties:

1. **replica** – The replica that is sending across the wedged message.
2. **history** – The replica’s history attribute, which contains all the order proofs.

Client request : Has the **client** as an attribute to identify the client for which the retransmission request has been made.

Retransmission reply message : Has the following properties:

1. **result**
2. **operation**
3. **resultProof**

Catch-up message:

1. **deltaHistory –** The set difference of the longest history and the truncated history; to be sent for replicas that require catching up.

Caught-up message:

1. **runningState –** Contains the running state of the replica after it has caught up.
2. **ch –** A hash of the running state.

Client: The client maintains the following state:

1. **currentConfig** – The current config as fetched from Olympus.
2. **isResponseReceived – Condition variable** for the wait on the reply for a particular request.
3. **pendingRequests –** Maintain a **map of requests waiting for a reply**, indexed by **operation**.
4. **privateKey**
5. **publicKeys –** of all players in the system

Client request message: It has the following properties:

1. **client**
2. **operation**
3. **isRetry –** Flag that indicates if the message is a retransmission request.