

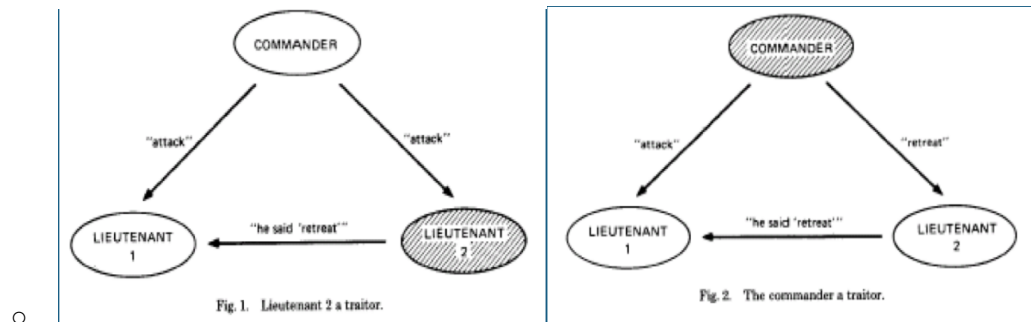
## 273 Research Paper

### Research paper Presentation

#### 1. Shivani Parate

- Byzantine Generals Problem
- Byzantine Fault tolerance
- Introduction to PBFT

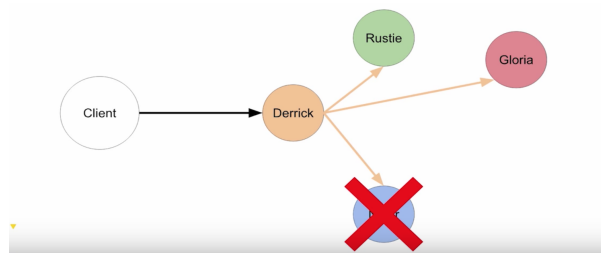
#### 2. Aditya Bhole



- Explanation of proof :  $3f + 1$  minimum number of replicas to be present in the system
- Safety and Liveness properties

#### 3. Rikitha Manjunath

- Algorithm description covering state machine replication and views
- Working of algorithm
- Details abt Client side



#### 4. Rajat Dineshchandra Chaurasia

- Normal case operation
- 3 phase protocol : PrePrepare , Prepare, Commit

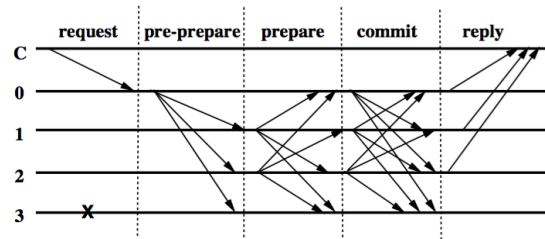


Figure 1: Normal Case Operation

## 5. Shraddha Kabade

- Applications of PBFT
- Demo of the working code
- Github Link : <https://github.com/bigpicturelabs/consensusPBFT>

## Executing the code:

- Start 4 servers (Apple, IBM, MS, Google)
- Command to start the server: ***go run main.go 'Apple'***
- Make post call using : ***curl -H "Content-Type: application/json" -X POST -d '{"clientID": "ssk5", "operation": "GetMyName", "timestamp": 1540694624}' http://localhost:1111/req***

## Working screenshot:

```
Terminal Shell Edit View Window Help
src — main • go run main.go Apple — 71x28
Last login: Sun Nov 4 17:13:01 on ttys005
[sk@skabade$ cd Documents/gowp/src/
[sk@skabade$ go run main.go 'Apple'
Server will be started at localhost:1111...
[REQUEST] ClientID: ssk5, Timestamp: 1540694624, Operation: GetMyName
[STAGE-DONE] Create the replica status
[STAGE-BEGIN] Consensus Process (ViewID:1000000000)
[STAGE-DONE] Pre-prepare
[PREPARE] NodeID: MS
[Prepare-Vote]: 1
[PREPARE] NodeID: Google
[Prepare-Vote]: 2
[STAGE-DONE] Prepare
[STAGE-BEGIN] Commit
[COMMIT] NodeID: MS
[Commit-Vote]: 1
[COMMIT] NodeID: Google
[Commit-Vote]: 2
[STAGE-DONE] Commit
Committed value: ssk5, 1540694624, GetMyName, 1541380494601997000
Result: Executed by MS
Result: Executed by Google
Result: Executed by IBM
Result: Executed by Google
Result: Executed by IBM
Result: Executed by Apple
[STAGE-DONE] Reply

src — main • go run main.go MS — 68x28
[sk@skabade$ cd Documents/gowp/src/
[sk@skabade$ go run main.go 'MS'
Server will be started at localhost:1112...
[PREPARE] ClientID: ssk5, Operation: GetMyName, SequenceID: 1541380494601997000
[STAGE-DONE] Create the replica status
[STAGE-BEGIN] Pre-prepare
[PREPARE] NodeID: IBM
[Prepare-Vote]: 1
[PREPARE] NodeID: Google
[Prepare-Vote]: 2
[STAGE-DONE] Prepare
[STAGE-BEGIN] Commit
[COMMIT] NodeID: Apple
[Commit-Vote]: 1
[COMMIT] NodeID: Google
[Commit-Vote]: 2
[STAGE-DONE] Commit
Committed value: ssk5, 1540694624, GetMyName, 1541380494601997000
[STAGE-DONE] Reply

src — main • go run main.go Google — 73x26
Server will be started at localhost:1113...
[PREPARE] ClientID: ssk5, Operation: GetMyName, SequenceID: 1541380494601997000
[STAGE-DONE] Create the replica status
[STAGE-BEGIN] Pre-prepare
[PREPARE] NodeID: MS
[Prepare-Vote]: 1
[PREPARE] NodeID: IBM
[Prepare-Vote]: 2
[STAGE-DONE] Prepare
[STAGE-BEGIN] Commit
[COMMIT] NodeID: Apple
[Commit-Vote]: 1
[COMMIT] NodeID: MS
[Commit-Vote]: 2
[STAGE-DONE] Commit
Committed value: ssk5, 1540694624, GetMyName, 1541380494601997000
[STAGE-DONE] Reply
[COMMIT] NodeID: IBM
[Commit-Vote]: 3
[STAGE-DONE] Commit
Committed value: ssk5, 1540694624, GetMyName, 1541380494601997000
[STAGE-DONE] Reply

src — main • go run main.go IBM — 69x26
Server will be started at localhost:1114...
[PREPARE] ClientID: ssk5, Operation: GetMyName, SequenceID: 1541380494601997000
[STAGE-DONE] Create the replica status
[STAGE-BEGIN] Pre-prepare
[PREPARE] NodeID: Google
[Prepare-Vote]: 1
[PREPARE] NodeID: MS
[Prepare-Vote]: 2
[STAGE-DONE] Prepare
[STAGE-BEGIN] Commit
[COMMIT] NodeID: Apple
[Commit-Vote]: 1
[COMMIT] NodeID: MS
[Commit-Vote]: 2
[STAGE-DONE] Commit
Committed value: ssk5, 1540694624, GetMyName, 1541380494601997000
[STAGE-DONE] Reply
[COMMIT] NodeID: Google
[Commit-Vote]: 3
[STAGE-DONE] Commit
Committed value: ssk5, 1540694624, GetMyName, 1541380494601997000
[STAGE-DONE] Reply

skabade — -bash — 54x24
Last login: Sun Nov 4 17:14:02 on ttys001
[sk@skabade$ curl -H "Content-Type: application/json" -X POST -d '{"clientID": "ssk5", "operation": "GetMyName", "timestamp": 1540694624}' http://localhost:1111/req
[sk@skabade$ ]
```