

# IMPLEMENTASI RAFT CONSENSUS

## SISTEM TERDISTRIBUSI KELOMPOK 5

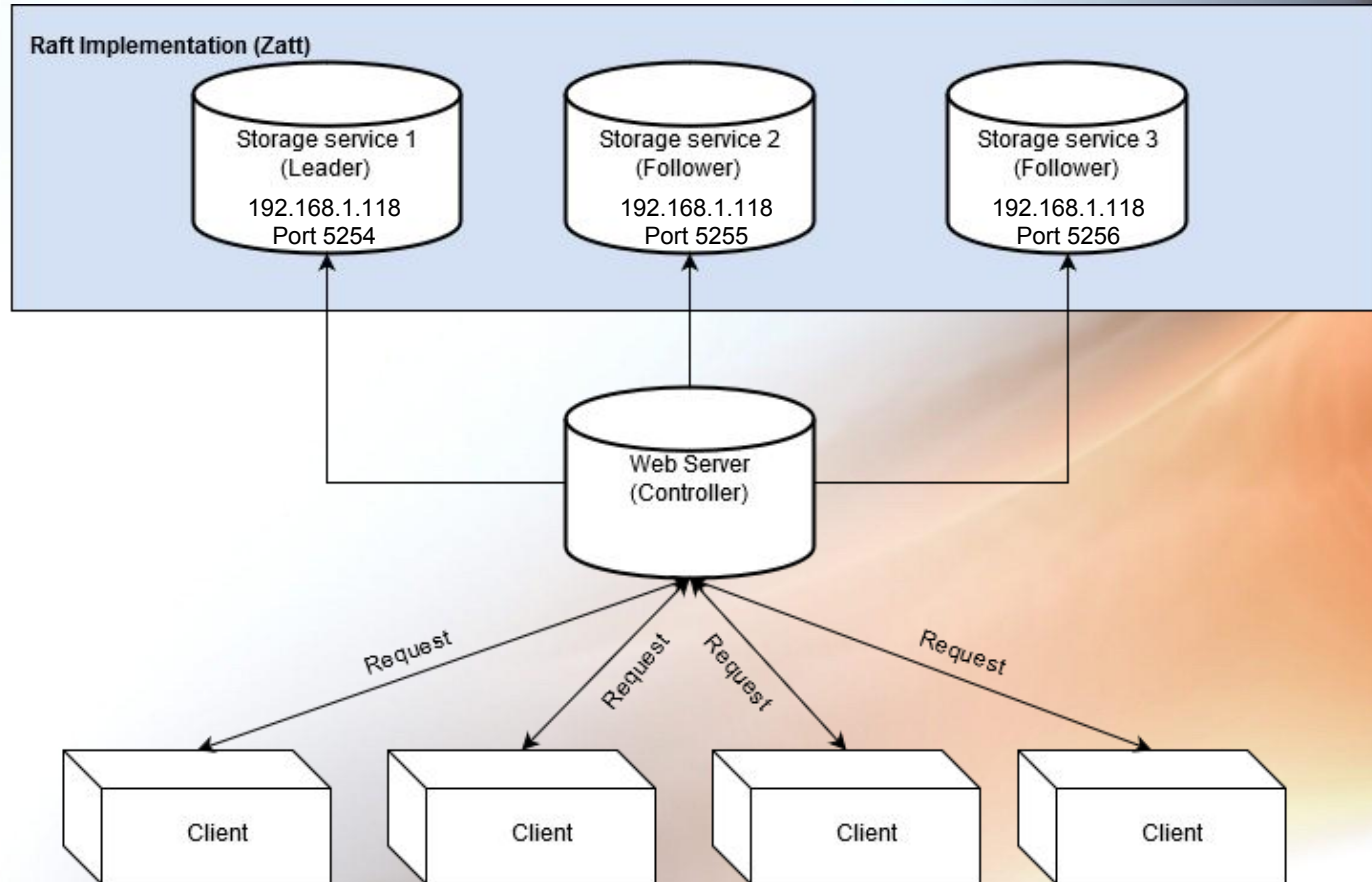
Faiq	05111540000007
Naufal P F	05111540000057
Dicky Kaisar Utomo	05111540000077
Subhan Maulana	05111540000149
Wahyu Pujiono	05111540000151
Rakhma Rufaida Hanum	05111540000161



# Pembagian Kerja

NRP	Nama	Pembagian Kerja
05111540000007	Faiq	Desain Model Flowchart
05111540000057	Naufal P F	Implementasi dan Konfigurasi
05111540000077	Dicky Kaisar Utomo	Implementasi dan Testing
05111540000149	Subhan Maulana	Desain Arsitektur Implemetasi
05111540000151	Wahyu Pujiono	Aplikasi HTTP
05111540000161	Rakhma Rufaida H	Dokumentasi Hasil Uji

# Desain Arsitektur Implementasi



# Desain Model Flowchart Implementasi

## Konfigurasi

1. Host name
2. IP
3. vagrant box
4. Memory

## Instalasi Zatt

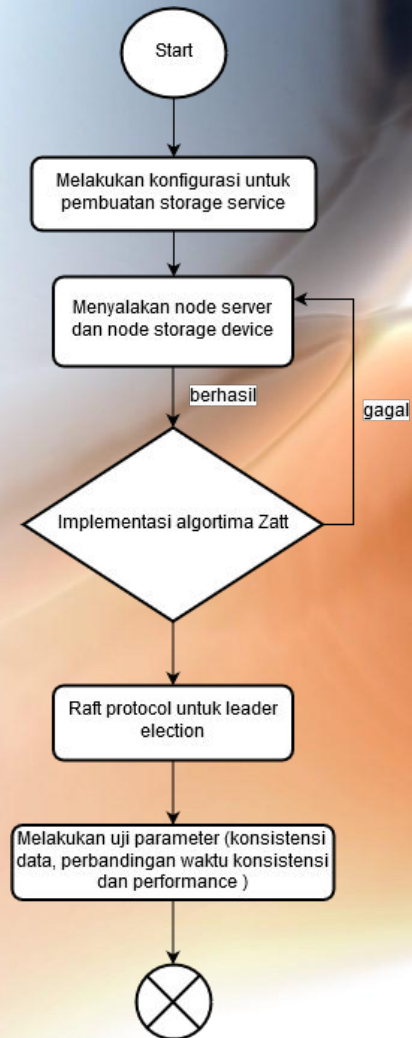
1. cloning github
2. membuat folder yang berisi konfigurasi cluster yang berisi id,ip dan port

## Request oleh Client

1. running zatt pada node yang telah dibuat
2. Menggunakan web server http untuk mengakses

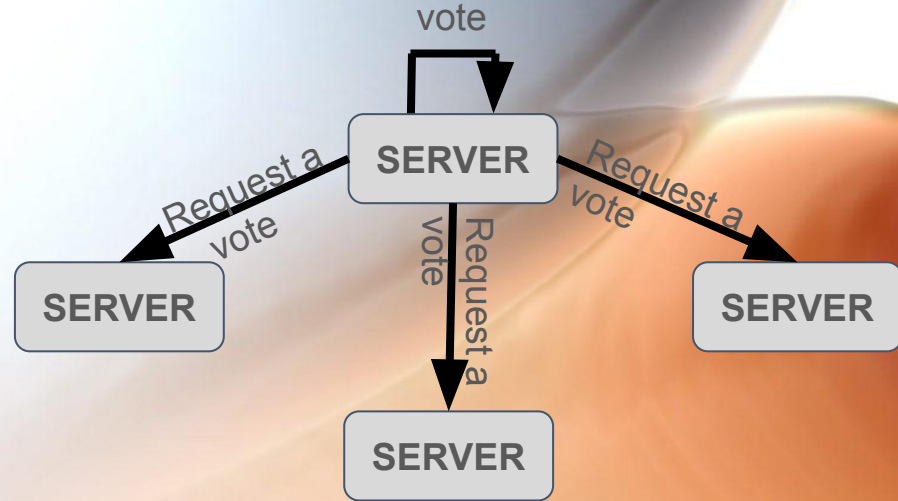
## Uji Parameter

- Konsistensi data
- Performa Raft
- Perbandingan waktu dengan acuan banyaknya data (ms)

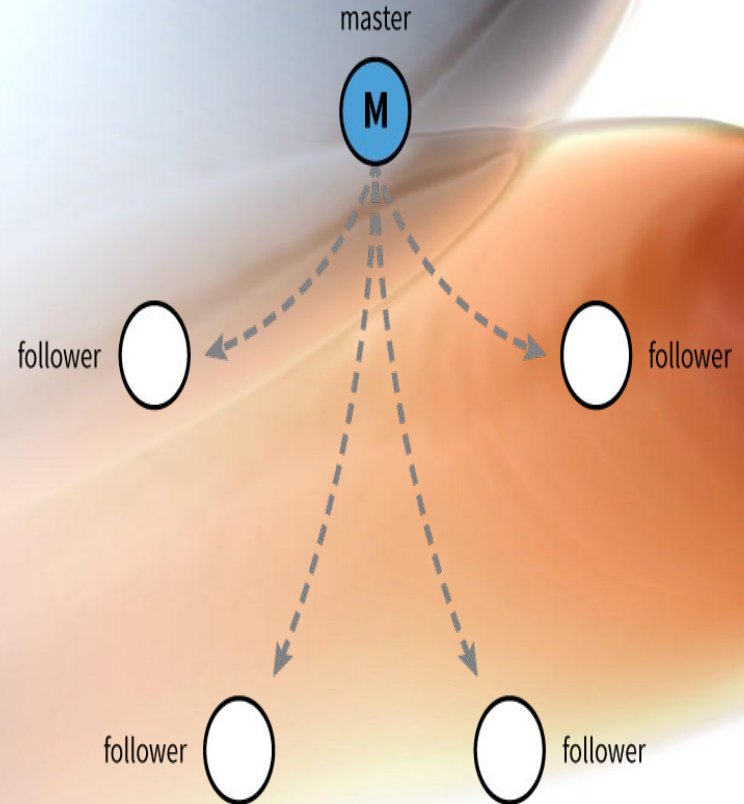
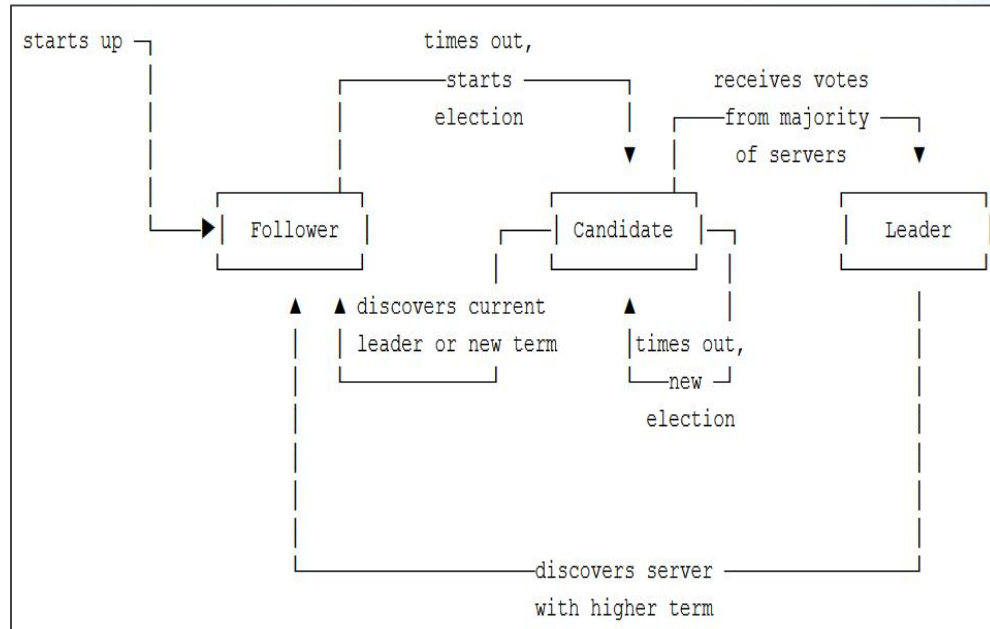


# Pembagian State

- Pada awalnya semua node atau web server menjadi follower state, masing-masing follower mencari apakah leader sudah ada atau belum. Jika belum, maka semua node menjadi candidate state, masing-masing candidate mengirimkan request dan menerima reply, node yang memiliki vote terbanyak dari semua follower akan menjadi leader.



# Leader Election



# Konfigurasi

## Vagrantfile

```
Vagrant.configure("2") do |config|
  (1..3).each do |i|
    config.vm.define "StorageServer#{i}" do |node|
      node.vm.hostname = "StorageServer#{i}"
      node.vm.box = "ubuntu/xenial64"
      node.vm.box_check_update = false
      node.vm.network "private_network", ip: "192.168.1.118:525#{i}"
      node.vm.network "public_network", bridge: "zat1"
      node.vm.provider "virtualbox" do |vb|
        vb.name = "storages#{i}"
        vb.gui = false
        vb.memory = "1024"
      end
      node.vm.provision "shell", path: "provision.sh", privileged: false
    end
  end
end
```

## zatt.conf

```
{
  "address": ["192.168.1.118", 5254],
  "storage": "zatt.persist",
  "cluster": [
    ["192.168.1.118", 5254],
    ["192.168.1.118", 5255],
    ["192.168.1.118", 5256],
    ["192.168.1.118", 5257]]
}
```

Cara menjalankan zatt pada storage service:

run\_zatt.sh

```
zatt -c zatt.conf --port 5254 -s zatt.0.persist --debug
zatt -c zatt.conf --port 5255 -s zatt.1.persist --debug
zatt -c zatt.conf --port 5256 -s zatt.2.persist --debug
```

# Repository github

<https://github.com/naufalpf/SistemTerdistribusi>





# Rencana Uji Coba Parameter

End user yakni client mengirimkan paket data pada storage utama, sebut saja storage W. storage W sebagai leader mendistribusikan data ke storage lainnya yaitu storage X,Y dan Z. hal ini disebut sebagai replikasi.

Parameter yang diuji adalah :

1. Konsistensi data yang terdistribusi pada masing masing storage
2. Performansi raft ketika storage ditambah
3. Performansi raft ketika storage dikurangi
4. Perbandingan waktu dengan acuan banyaknya data (ms)