Serf is a tool for cluster membership, failure detection, and orchestration that is decentralized, fault-tolerant and highly available.

It is primarily used for web server.

Requirements:

Linux, Mac OS X, and Windows.

5 to 10 MB of resident memory

Must be installed on each cluster member (node)

Flow:

Serf communicates servers in a cluster using a gossip protocol (UDP) to provide

1. Server membership management

Includes automatic tasks on severs’ changes, LB, HA

1. Failure detection and recovery

Executes a handler script when a cluster member is down

1. Custom event propagation

Propagate configuration, deploys

Installing Serf

-Download and run the installer

-run the executable to test

Running the Agent

It is possible to run multiple agents in a server, and make the server be a node on multiple clusters

$ serf agent

//\* it will automatically join the cluster

//\* The terminal windows or console must be left open with serf running, it does not run as a service.

Cluster Members

$ serf members

Stopping the Agent

Use <CTRL>-C on the running terminal or console to stop the agent. It will notify the cluster that is left, avoiding a failed state.

Join a Cluster

To join an existing cluster, Serf only needs to know about a single existing member. After it joins, the agent will gossip with this member and quickly discover the other members in the cluster.

Start 1st member of the New Cluster

$ serf agent -node agent-node-1 -bind=172.20.20.10

//\* node is the name of the server in the cluster

//\* bind is the address that Serf listen on. Normally localhost.

Join a 2nd Member to the Cluster

$ serf agent -node agent-node-test -bind=172.20.20.11

//\* node is the name of the server in the cluster

//\* bind is the address that Serf listen on. Normally localhost.

$ serf join 172.20.20.10

Event Handlers and more

See https://www.serf.io/intro/getting-started/event-handlers.html