

SUPFile-PROJECT

***Class* Group Project**

***Storage & HA***

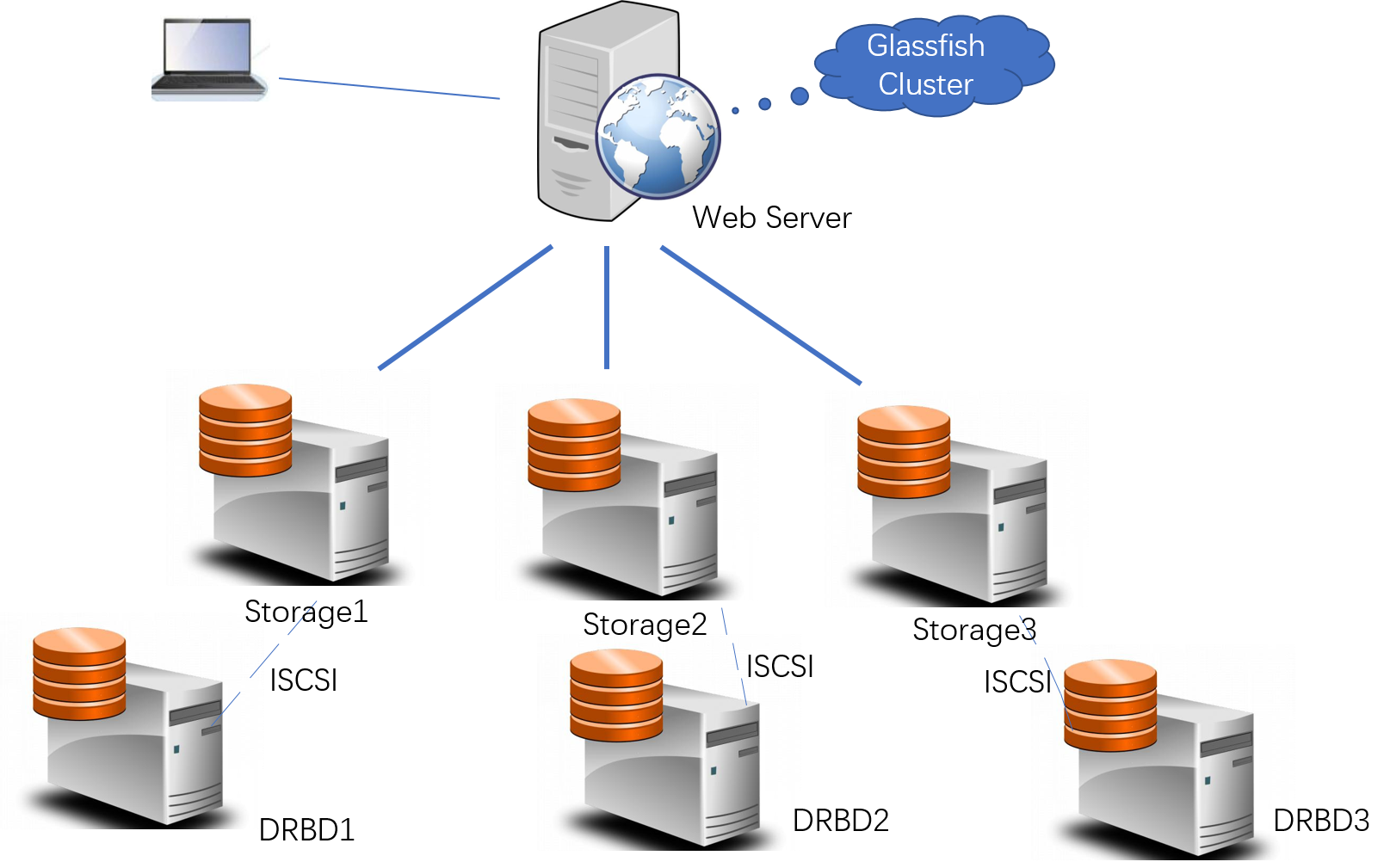
Delivery

Group members

Documentation

# 

# 1 Architecture Overview



## **1.1 Storage**

For the storage, we use samba to create a shared storage on each of the 3 storage servers, we use Centos operating system. We use SMB protocal to access server’s resources.

## **1.2 High Availablity**

For the High availability, we use DRBD and ISCSI for the storage, and Glassfish Cluster for HA of glassfish.

# 2 Implementation

## **2.1 Storage**

These are procedures we used to implement share function.

1. Check samba server

**Rpm –qa | grep samba** to check our samba environment.

**Yum install samba** to install samba server side.

1. Shut down firewall and selinux.

**systemctl status firewalld**

**systemctl stop firewalld**

**getenforce**

**nano /etc/selinux/config**

**SELINUX=disabled**

1. Configure samba

Change **smb.n** configuration file : **setsebool –P samba \_enable\_home\_dirs on**.

**smbpasswd –a root** to add a samba user.

Then we can access the remote directory as it is on the local disk.

For the shared storage, we used 3 storage servers, configure samba on each of them, then we can access them on our web server. When a user registered, he wil assign 30GB of storage on the least used storage server.

## **2.2 High Availablity**

### 2.2.1 Storage Cluster (DRBD + ISCSI)

In case of disks failure and any other unexpected situation, we store all user files and data in there storage servers. We use DRBD and ISCSI to back up those data into 2 nodes, one primary, and one secondary.

|  |  |  |
| --- | --- | --- |
| server | Interface | IP |
| DRBD1 | Eth0 | 192.168.100.101 |
| DRBD2 | Eth0 | 192.168.100.102 |
| DRBD3 | Eth0 | 192.168.100.103 |

1.network interface

/etc/network/interfaces:

auto lo

iface lo inet loopback

allow-hotplug eth0

iface eth0 inet static

address 192.168.100.101

netmask 255.255.255.0

2.drbd configure

/etc/drbd.d/storage.res:

resource storage{

device /dev/drbd1;

disk /dev/sdb1;

meta-disk internal;

on drbd1{

address 192.168.100.101:7788;

}

on drbd2{

address 192.168.100.102:7788;

}

}

3.iscsi target configure

/etc/iet/ietd.conf:

Target iqn.2016.12.evilcorp.fr:storage

Lun 0 Path=/dev/drbd1,Type=fileio

4.high available configure

/etc/corosync/corosync.conf:

interface {

ringnumber: 0

bindnetaddr: 192.168.100.0

mcastaddr: 239.12.13.14

mcastport: 5405

}

logging{

to\_logfile: yes

logfile: /var/log/corosync.log

}

Highly Ability in DRBD1 && DRBD2:

crm

crm(live)#configure

crm(live)configure# property stonith-enabled=”false” crm(live)configure# no-quorum-policy=”ignore”

crm(live)configure# rsc\_defaults resource-stickiness=”1000”

crm(live)configure# primitive drbd-ip ocf:heartbeat:IPaddr params ip=”192.168.100.200” cidr\_netmask=”24”

crm(live)configure# primitive drbd-service heartbeat:drbddisk

crm(live)configure# primitive iscsi-service lsb:iscsitarget

crm(live)configure# group drbd drbd-ip drbd-service iscsi-service

crm(live)configure# commit

### 2.2.2 Glassfish Cluster

Clustering and HA support includes the creation of multiple clusters per domain and multiple instances per cluster with up to 100 instances per domain using the web-based Admin Console or the command line interface.

Environment : linux, Glassfish Server, JDK.

Implementation :

-Install JDK

-Glassfish :

#java -Xmx256m -jar glassfish-installer-v2.1.1-b31g-linux.jar

#ant -f setup-cluster.xml

#bin/asadmin start-domain

#bin/asadmin create-cluster –host localhost –port 4848 supfile

#bin/asadmin create-node-agent –host 192.168.1.114 –port 4848

agent01

# bin/asadmin create-node-agent –host 192.168.1.114 –port 4848

Agent02

#bin/asadmin create-instance –host localhost –prot –nodeagent agent01 –cluster supfile node1

#bin/asadmin create-instance –host localhost –prot –nodeagent agent01 –cluster supfile node2