

# **PENGEMBANGAN SERVER SIAKAD UNIVERSITAS DEHASEN BENGKULU MENGGUNAKAN HIGH AVAILABILITY CLUSTERING DAN MYSQL DATABASE REPLICATION**

Hari Aspriyono

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dehasen Bengkulu, Bengkulu

Jl. Meranti Raya No. 32 Sawah Lebar Kec. Ratu Agung Kota Bengkulu, 38228  
hari.aspriyono@unived.ac.id

## **Abstrak**

Penelitian ini dilakukan untuk memahami konsep *high availability clustering* dan *mysql database replication* yang diimplementasikan dalam pengembangan *server* sistem informasi akademik Universitas Dehasen Bengkulu. Pengumpulan data dilakukan melalui observasi secara langsung dan wawancara terhadap pengelola *server* sistem informasi akademik yang ada di Universitas Dehasen Bengkulu. Selanjutnya dilakukan analisa terhadap kondisi *server* dan aplikasi sistem informasi akademik yang akan dikembangkan dengan mengimplementasikan metode *high availability clustering* dan *mysql database replication*. Pengembangan *server* ini dilakukan dengan menambahkan dua buah *server* menggunakan sistem operasi *Linux CentOS 6.6*. Pengujian dilakukan untuk mengukur tingkat ketersediaan (*availability*) dan konsistensi data. Untuk menguji ketersediaan *server* dilakukan monitoring dengan membandingkan *server* sebelum dan setelah pengembangan. Sedangkan untuk menguji konsistensi data dilakukan dengan cara melakukan perubahan data dari *server* yang sedang melayani akses ketika salah satu *server* mengalami kegagalan atau *down*. Dari hasil pengujian dapat disimpulkan bahwa *high availability clustering* dan *mysql database replication* dapat meningkatkan ketersediaan (*availability*) dan konsistensi data pada *server* sistem informasi akademik Universitas Dehasen Bengkulu.

**Kata kunci** : *server, high availability clustering, mysql database replication*

## **Pendahuluan**

*Server* adalah penyedia sumber daya dan layanan pada jaringan, termasuk *server* sistem informasi akademik Universitas Dehasen Bengkulu. Hal yang sangat perlu diperhatikan dalam memberikan layanan adalah ketersediaan akses dan data, dimana *server* sistem informasi akademik Universitas Dehasen Bengkulu harus dapat diakses kapan saja.

Kenyataan yang sering terjadi adalah sistem informasi akademik tidak dapat diakses karena terjadi gangguan pada sistem baik gangguan terencana maupun gangguan yang tidak terencana. Gangguan terencana dapat berupa pemeliharaan (*maintenance*) sistem berkala, sedangkan gangguan tidak terencana dapat berupa kerusakan komputer *server*, *storage*, *data corruption*, jaringan dan terjadinya bencana alam. Hal ini menyebabkan *server* tidak dapat diakses dan hilangnya data yang sangat

penting, tentu saja hal ini sangat tidak diinginkan.

*High availability cluster* yang sering disebut sebagai *failover cluster* dimana konsep dasar dari teknik ini adalah dengan menempatkan satu atau lebih *server* cadangan dalam modus siaga, yang dapat *online* secara otomatis dalam beberapa saat hanya setelah menemukan kegagalan pada *server* utama (*automatic failover*).

Konsistensi data pada masing-masing *server* baik *server* utama maupun *server* cadangan sangat diperlukan. Untuk itu dibutuhkan suatu teknologi yang mampu melakukan replikasi data pada masing-masing *server*. Replikasi adalah suatu teknik untuk melakukan *copy* dan pendistribusian data serta objek-objek *database* dari satu *database* ke *database* lain dan melaksanakan sinkronisasi antara *database* sehingga konsistensi data pada semua server dapat terjamin.

Beberapa penelitian terdahulu tentang implementasi *high availability* yakni oleh (Pribadi, 2013) yang mengimplementasikan *high availability* pada layanan *VPN Client* menyimpulkan bahwa dengan adanya sistem *high availability VPN Client*, layanan *VPN* tidak akan terganggu yang diakibatkan oleh kerusakan pada *primary server* karena *secondary server* akan mengambil alih tugas *primary server* dengan baik jika terjadi kegagalan.

Selain penelitian tentang *high availability clustering*, penelitian lain juga telah dilakukan dan dituliskan dalam jurnal atau karya ilmiah tentang *backup* data adalah (Safitri Wahyuningsih, 2011) menguji dua metode *backup* data yang terdapat pada *server database MySQL* antara lain *MySQLDump* dan replikasi. Kedua metode tersebut diuji dengan membandingkan pengukuran memori, penggunaan CPU dan waktu proses pada *server* saat proses *backup* terjadi hampir bersamaan dengan proses transaksi *input* data. Kesimpulannya adalah metode *backup* menjadikan replikasi lebih efisien dibandingkan dengan *MySQLDump* dari sisi penggunaan *memory* dan CPU.

Solusi yang ditawarkan pada penelitian ini adalah dengan mengimplementasikan *high availability clustering (ha-cluster)* pada *server* sistem informasi akademik Universitas Dehasen Bengkulu, sehingga *server* dapat melakukan *automatic failover* atau berpindah secara otomatis ke *server* redundan apabila terjadi kegagalan pada *server* utama. Untuk menjaga konsistensi data pada semua *server*, pada pengembangan *server* ini juga memanfaatkan fasilitas replikasi yang ada pada *MySQL database*. Dengan solusi ini *server* akan memiliki tingkat *availability* yang tinggi dengan waktu *fileover* yang cepat dan konsistensi data yang terjamin.

## Landasan Teori Server

Komputer *server* adalah komputer yang biasanya dikhususkan untuk menyimpan data yang akan digunakan bersama, atau berbagai basis data. Selain itu, jika menggunakan sistem operasi berbasis *network (Network Operating System)* maka komputer *server* berisi informasi daftar *user* yang diperbolehkan masuk ke *server*

tersebut, berikut otoritasnya yang dapat di-manage oleh *administrator*.(Zakarya et al., 2012)

### Cluster Computing

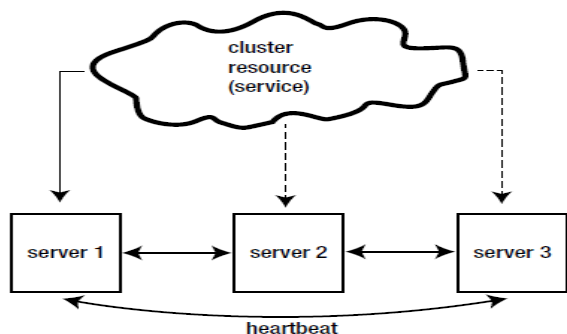
*Cluster computing* atau komputasi kluster adalah jenis komputasi di mana beberapa *node* yang dibuat untuk dijalankan sebagai satu kesatuan. Berbagai *node* yang terlibat dalam *cluster* biasanya terhubung satu sama lain menggunakan beberapa jaringan lokal yang cepat.(Rai A.K dan Kaur K., 2014)

### High Availability

*High Availability (HA)* merupakan konsep yang menawarkan tingkat *available* yang tinggi terhadap suatu sistem. Konsep ini biasanya berkaitan dengan kemampuan sistem mengatasi *system hang, crash/down* maupun kesalahan pada jaringan.(A B M Moniruzzaman, 2014) Solusi yang ditawarkan berupa *backup* data atau *failover* data yang dilakukan secara *real time*. Saat *server* utama berhenti berjalan, maka *server slave* akan mengambil alih peran *server* utama dengan kualitas penanganan input dan output yang sama dengan *server* utama. Sistem akan selalu melakukan sinkronisasi data diantara keduanya atau mungkin lebih untuk mendapatkan redundansi data.

### High Availability Clustering

*High availability* adalah ilmu untuk menciptakan redundansi dalam setiap sistem dan subsistem untuk memastikan bahwa layanan tetap *up* dan tersedia. *High availability* pada dasarnya menempatkan satu atau lebih *server* cadangan dalam modus siaga, yang bisa *online* dalam beberapa saat hanya setelah mereka menemukan kegagalan pada sistem utama.(Gadir et al., 2005)



Gambar 1

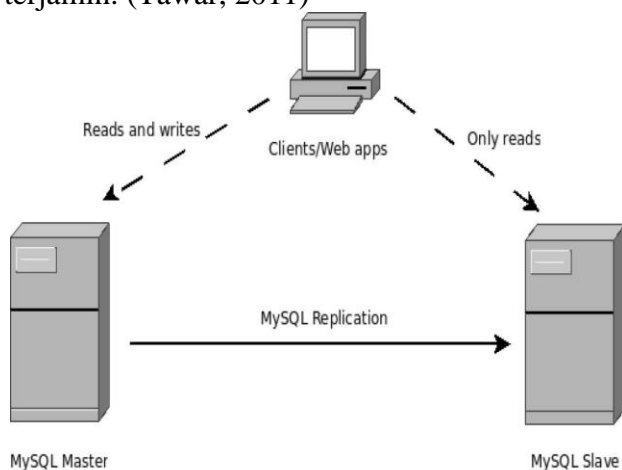
Konsep Dasar High Availability Cluster

### Heartbeat

*Heartbeat* adalah salah satu program yang terpisah atau termasuk dalam fungsi utama dari aplikasi cluster. *Heartbeat* bertujuan untuk terus mem-*polling* server dalam konfigurasi cluster untuk memastikan bahwa mereka sudah *up* dan merespon. (Priyadi, 2013)

### Replikasi

Replikasi adalah suatu teknik untuk melakukan *copy* dan pendistribusian data dan objek-objek database dari satu database ke database lain dan melakukan sinkronisasi antara database sehingga konsistensi data dapat terjamin. (Tawar, 2011)



Gambar 2

Ilustrasi Replikasi Database

### Metode Penelitian

Metodologi penelitian merupakan sekumpulan peraturan, kegiatan, dan prosedur yang digunakan untuk memperoleh informasi dan data yang lengkap, sehingga dalam hal penyelesaian masalah dapat lebih teratur dan mendapatkan hasil yang maksimal.

Penelitian adalah suatu cara untuk memahami sesuatu melalui penyelidikan atau usaha mencari bukti-bukti yang muncul sehubungan dengan masalah itu, yang dilakukan secara teliti sehingga diperoleh suatu pemecahan masalah. Penelitian ini bertujuan untuk memahami desain sistem server yang telah ada dan selanjutnya melakukan analisa masalah-masalah yang ada sebagai dasar pengembangan server untuk mengatasi masalah yang ada pada server yang telah ada. Analisa dan pengembangan tersebut kemudian diimplementasikan dan dilakukan pengujian untuk melihat optimalisasi dari hasil pengembangan.

Adapun tahapan yang dilakukan dalam penelitian ini adalah:

1. Identifikasi Permasalahan  
Identifikasi permasalahan dilakukan untuk menentukan masalah-masalah yang terjadi pada tempat penelitian, kemudian dirumuskan dan diberikan batasan-batasan permasalahan yang akan diteliti.
2. Studi Pustaka  
Studi Pustaka atau studi kepustakaan dilakukan untuk mendapatkan informasi sebagai referensi terkait dengan masalah dalam penelitian dan digunakan sebagai landasan pengembangan server. Dalam penelitian ini referensi yang dibutuhkan meliputi jaringan komputer, server, cluster computing, high availability clustering, dan MySQL database replication. Studi pustaka dilakukan supaya dalam penelitian ini tidak menyimpang dari teori-teori yang telah ada.
3. Pengumpulan Data  
Dalam penelitian ini proses pengumpulan data dilakukan dengan beberapa cara yaitu:
  - a) Pengamatan langsung ke lokasi penelitian (observasi) guna melihat kondisi server dan data yang berkaitan dengan penelitian, dapat juga dilakukan dengan mempelajari dokumentasi yang ada di tempat penelitian;
  - b) Wawancara, yaitu bagian dari teknik pengumpulan data yang dilakukan dengan menanyakan langsung terkait kondisi server dengan pegawai yang bertanggung jawab terhadap objek yang diteliti.

#### 4. Analisis

Analisis dilakukan pada *server* yang sedang berjalan untuk mengetahui spesifikasi *server*, topologi jaringan yang berjalan, kelemahan sistem *server*. Selain itu juga dilakukan pada sistem informasi akademik untuk mengetahui spesifikasi sistem, *system requirement* dan *database* yang digunakan. Hasil analisis akan digunakan untuk pengembangan *server* yang akan dilakukan.

#### 5. Implementasi

Dalam proses implementasi dibagi menjadi beberapa tahapan yaitu:

- a) Persiapan *hardware* dan *software*
- b) *Re-design* topologi jaringan baru
- c) Instalasi *software* pada *server*

Setelah selesai instalasi, sistem informasi akademik diakses melalui jaringan *intranet* dan *internet*. Sistem informasi akademik yang berada pada sistem *server* yang menerapkan *ha-cluster* dan *MySQL database replication* harus berfungsi dengan baik sesuai dengan rencana pengembangan.

#### 6. Pengujian

Pengujian dilakukan untuk memastikan bahwa optimasi yang dilakukan melalui pengembangan *server* ini berjalan dengan baik. Parameter yang diuji adalah tingkat *availability* dan konsistensi data pada *database* masing-masing *server*. Adapun rencana pengujian dilakukan dengan cara sesuai dengan parameter yang diuji yaitu:

##### a) *Availability*

Tingkat *availability* atau ketersediaan diuji menggunakan *online website monitoring tool* yaitu dengan mendaftarkan alamat *url* atau alamat *domain* sistem informasi akademik Universitas Dehasen Bengkulu ke situs <http://www.pingdom.com> yang merupakan situs penyedia layanan *website monitoring*. Sehingga akan didapatkan nilai *uptime* dan *downtime*.

##### b) Konsistensi Data

Konsistensi data diuji dengan mencoba melakukan *query* (penambahan data, perubahan data, penghapusan data) pada *server* utama dan melihat hasilnya pada *backup server* dan sebaliknya.

### Hasil dan Pembahasan

#### Analisis Permasalahan

#### Spesifikasi Server Berjalan

Spesifikasi *server* sistem informasi akademik Universitas Dehasen Bengkulu terdiri dari spesifikasi perangkat keras (*hardware*) dan spesifikasi perangkat lunak (*software*).

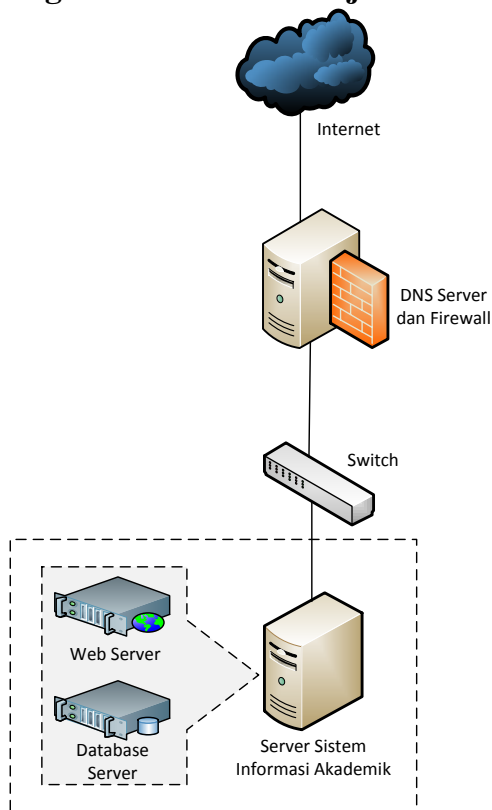
Tabel 1  
Spesifikasi Perangkat Keras Server

No	Alat	Unit	Spek
1	Processor	1	4 x Intel Xeon CPU E5506 @ 2.13GHz
2	Memory	1	4GB
3	Harddisk	1	120GB
4	Ethernet Card	4	Fast Ethernet Card 10/100 MBps

Tabel 2  
Spesifikasi Perangkat Lunak Server

No	Software	Spesifikasi
1	Sistem Operasi	CentOS Release 5.7 (Final)
2	Web Server	Apache 2.0.52
3	Programming Language	PHP Version 5.2.17
4	Database	MySQL Server 5.5

## Topologi Sistem Server Berjalan



Gambar 3

Topologi Sistem Server Berjalan

## Analisa Kelemahan Server yang Sedang Berjalan

Analisa kelemahan *server* dilakukan untuk mengetahui kondisi *server* sebelum implemantasi *high availability clustering* dan *mysql database replication*. Berikut ini beberapa analisis penulis berdasarkan literatur dan pengamatan pada *server* sistem informasi akademik Universitas Dehasen tanpa penerapan *high availability cluster*.

1. Sering terjadi kegagalan akses ke sistem informasi akademik, dikarenakan *server* sedang *down*, yang disebabkan oleh faktor-faktor berikut:
  - a. *Crash* pada *Harddisk*
  - b. Kerusakan pada *Network Interfaces*
  - c. *Server* mati total
2. Membutuhkan *administrator* untuk penanganan kegagalan sistem, di mana *system administrator* harus menangani secara langsung dengan merekonfigurasi layanan ke *server*.
3. Menimbulkan *downtime* yang cukup lama untuk penanganan kegagalan, berpotensi

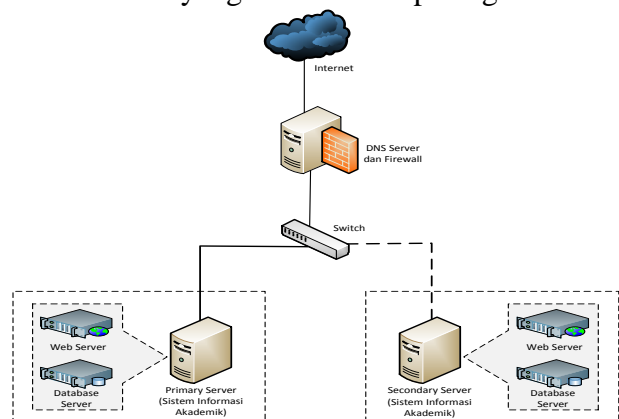
buruknya layanan sistem informasi akademik yang digunakan oleh dosen, staf dan mahasiswa Universitas Dehasen Bengkulu.

4. Konsistensi data antara data yang ada di *server* dengan data *backup* tidak terjamin, karena *backup* data tidak dilakukan secara *realtime*.

## Pemecahan Masalah

### Perancangan Topologi

Perancangan topologi ini dibuat sebagai desain atau gambar perancangan interkoneksi pada *server* sistem informasi akademik yang dikembangkan. Topologi sistem terdiri dari 3 buah *server* yaitu *DNS Server*, *Primary Server*, dan *Secondary Server*. *Secondary Server* merupakan *server* yang ditambahkan sebagai server redundan sesuai dengan konsep *high availability clustering*. Perancangan topologi sistem *server* yang baru terlihat pada gambar 4.

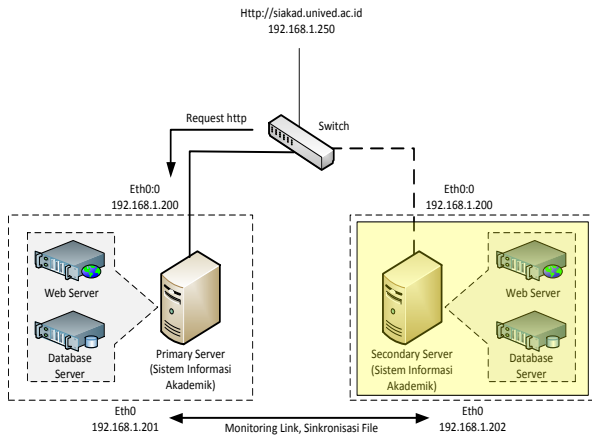


Gambar 4

Topologi Sistem Server Baru

## Skema Kerja

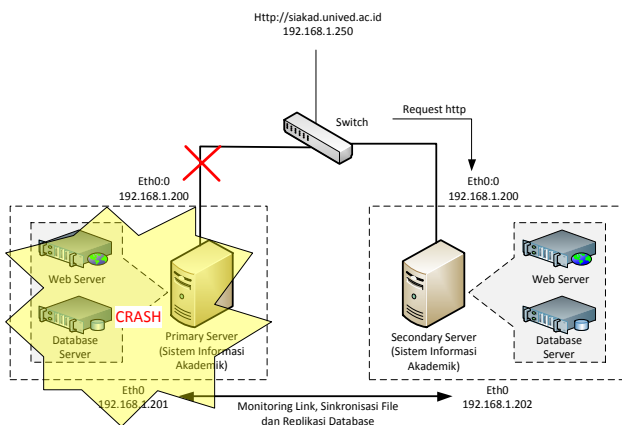
Pada kondisi normal *request* sistem informasi akademik dari *user* dilayani oleh *primary server*, sedangkan *secondary server* dalam posisi *standby*. *Primary server* selalu berkomunikasi dengan *secondary server* untuk melakukan monitoring *link*, sinkronisasi *file* dan replikasi *database* yang direpresentasikan oleh garis dengan tanda panah bolak-balik. *Primary server* akan selalu memberi tahu pada *secondary server* jika *primary server* dalam keadaan normal, sehingga *secondary server* hanya menerima sinkronisasi *file* dan replikasi *database*.



Gambar 5

Skema Kerja Server pada Kondisi Normal

Selain itu untuk skema kerja dalam kondisi *primary server down*. *Primary server* tidak dapat melakukan komunikasi dengan *secondary server* diakibatkan oleh kegagalan pada *primary server* yang direpresentasikan garis silang warna merah. Saat tidak ada komunikasi dari *primary server*, *secondary server* akan menyatakan *primary server* dalam kondisi *down* sistem, sehingga *secondary server* akan mengaktifkan *eth0:0* dan *service httpd* untuk melayani *user* yang mengakses *server* sistem informasi akademik.



Gambar 6

Skema Kerja Pada Kondisi Primary Server Down/Crash

Instalasi Perangkat Lunak Server  
Instalasi Linux Centos 6.6

Tahap awal dalam implementasi *ha-cluster* dan *MySQL database replication* adalah instalasi *CentOS* pada computer *server*. Adapun bagian penting dalam instalasi *CentOS* adalah:

1. Menentukan *Hostname*

*Hostname* merupakan nama komputer, pada kasus ini adalah nama pada masing-masing komputer *server* yaitu *server1* sebagai *primary server* dan *server2* sebagai *secondary server*.

2. Menentukan *Root Password*

*Root password* adalah kata kunci untuk masuk sebagai *user root* atau *user* dengan level tertinggi pada *CentOS*.

3. Menentukan Tipe Instalasi

Tipe instalasi yang dimaksud adalah *set of software* yang dalam kasus ini penulis memilih tipe *minimal*, dengan tujuan untuk mengoptimalkan kinerja *server* supaya tidak terbebani dengan paket-paket *software* yang tidak dibutuhkan.

Konfigurasi Network dan IP Address

Konfigurasi *network* dilakukan untuk menentukan *IP Address* dari masing-masing *server*. Konfigurasi pada *CentOS* tersimpan dalam bentuk *file*, dimana *file* konfigurasi untuk mengatur *IP Address* tersimpan pada folder */etc/sysconfig/network-scripts/* dengan nama file *ifcfg-eth0*. Adapun konfigurasi pada *primary server* terlihat pada gambar 7.

```

root@server1:~# cat /etc/sysconfig/network-scripts/ifcfg-eth0
DEVICE=eth0
TYPE=Ethernet
UUID=15ee3b3d-5c0a-4728-ab25-e7d41bb448db
ONBOOT=yes
NM_CONTROLLED=yes
BOOTPROTO=static
HWADDR=90:F6:52:04:E5:F3
IPADDR=192.168.1.201
NETMASK=255.255.255.0
GATEWAY=192.168.1.250
NETWORK=192.168.1.0
BROADCAST=192.168.1.255
DNS1=192.168.1.250
IPV6INIT=no
    
```

Gambar 7

Konfigurasi Primary Server

Sedangkan konfigurasi pada *secondary server* terlihat pada gambar 8.

```

root@server2:~# cat /etc/sysconfig/network-scripts/ifcfg-eth0
DEVICE=eth0
TYPE=Ethernet
UUID=3269fea7-1b96-458d-a76b-b940a42378e6
ONBOOT=yes
NM_CONTROLLED=yes
BOOTPROTO=static
HWADDR=08:69:95:F9:89:35
IPADDR=192.168.1.202
NETMASK=255.255.255.0
GATEWAY=192.168.1.250
NETWORK=192.168.1.0
BROADCAST=192.168.1.255
DNS1=192.168.1.250
IPV6INIT=no
    
```

Gambar 8

Konfigurasi Pada Secondary Server

Instalasi Web Server

*Apache web server* diinstal pada *primary server* dan *secondary server*. Setelah *web server* di-install, selanjutnya dilakukan konfigurasi

sesuai kebutuhan yang terlihat pada tabel di bawah ini.

Tabel 3  
Parameter Konfigurasi Web Server

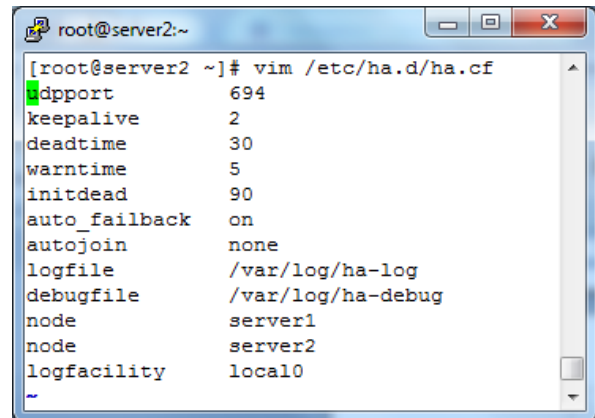
No	Param	Nilai	Keterangan
1	Listen	80,8080	Menentukan port yang dapat diterima oleh web server.
2	Server Name	192.168.1.200:80	Ip Address dan Port tempat web server dijalankan.
3	Doc. Root	/home/kruwell/public_html/app	Direktori dimana aplikasi sistem informasi akademik tersimpan.
4	UserDir	public_html	Folder user yang diaktifkan, sehingga dapat dibaca oleh web server.
5	Name Virtual Host	*:80	Mengaktifkan Virtual Host pada server

### Instalasi PHP

Instalasi *php* dilakukan untuk mendukung bahasa pemrograman yang digunakan pada aplikasi sistem informasi akademik, sehingga perlu juga di-*install* beberapa *module php* yang dibutuhkan.

### Implementasi Ha-Cluster

*Ha-Cluster* dibangun menggunakan *Heartbeat* untuk proses *failover server* pada metode *high availability clustering*. Konfigurasi *ha-cluster* menggunakan *heartbeat* terlihat pada gambar 9.

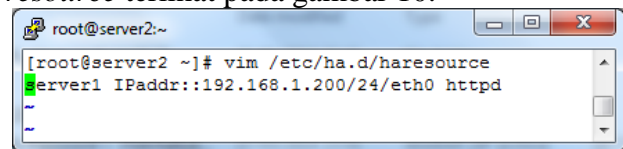


```

root@server2:~# vim /etc/ha.d/ha.cf
dpport        694
keepalive     2
deadtime     30
warntime     5
initdead     90
auto_failback on
autojoin     none
logfile      /var/log/ha-log
debugfile    /var/log/ha-debug
node         server1
node         server2
logfacility   local0
    
```

Gambar 9

Knfigurasi *ha-cluster* dengan *heartbeat* Untuk proses *failover* dibutuhkan konfigurasi *resource*, adapun konfigurasi *resource* terlihat pada gambar 10.



```

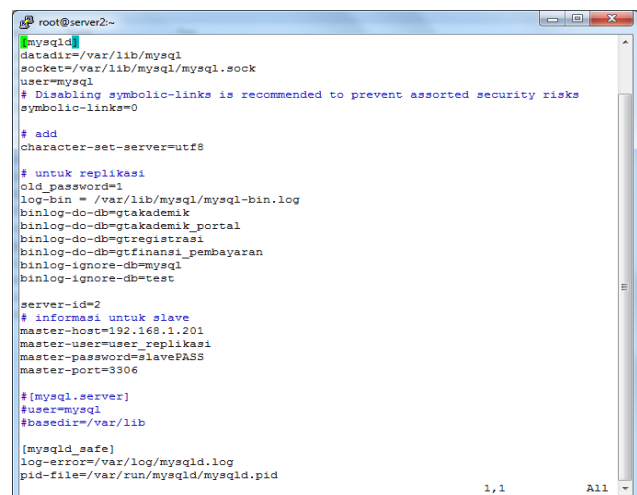
root@server2:~# vim /etc/ha.d/haresource
server1 IPaddr::192.168.1.200/24/eth0 httpd
    
```

Gambar 10

Konfigurasi *Resource Failover*

### Implementasi MySQL Database Replication

*MySQL Server* di-*install* pada masing-masing *server*, baik *primary server* maupun *secondary server*. Konfigurasi dilakukan untuk keperluan replikasi *database*. Replikasi *database* yang digunakan adalah tipe *master to master*. Adapun *database* yang akan direplikasi adalah 4 buah *database* yaitu *gtakademik*, *gtakademik\_portal*, *gregistrasi*, dan *gtfinansial\_pembayaran*.



```

root@server2:~# vim /etc/my.cnf
[mysqld]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
user=mysql
# Disabling symbolic-links is recommended to prevent assorted security risks
symbolic-links=0

# add
character-set-server=utf8

# untuk replikasi
old_password=1
log-bin = /var/lib/mysql/mysql-bin.log
binlog-do-db=gtakademik
binlog-do-db=gtakademik_portal
binlog-do-db=gregistrasi
binlog-do-db=gtfinansial_pembayaran
binlog-ignore-db=mysql
binlog-ignore-db=test

server-id=2

# informasi untuk slave
master-host=192.168.1.201
master-user=mysql_replikasi
master-password=SlavePASS
master-port=3306

[mysql.server]
#user=mysql
#basedir=/var/lib

[mysqld_safe]
log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid
    
```

Gambar 11

Konfigurasi Service MySQL

Hasil konfigurasi *mysql* replikasi dapat dilihat melalui status *master* dan status *slave* pada masing-masing *server*.

```
mysql> show master status;
+-----+-----+-----+-----+
| File           | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+-----+-----+-----+-----+
| mysql-bin.000002 | 5283    | qtakademik,qtakademik_portal,qtregistrasi,grfinansi_pembayaran | mysql,test |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

Gambar 12  
Status Master Pada Server Utama

```
mysql> show slave status \G
***** 1. row *****
Slave_IO_State: Waiting for master to send event
Master_Host: 192.168.1.201
Master_User: user_replikasi
Master_Port: 3306
Connect_Retry: 60
Master_Log_File: mysql-bin.000002
Read_Master_Log_Pos: 5283
Relay_Log_File: mysqld-relay-bin.000048
Relay_Log_Pos: 251
Relay_Master_Log_File: mysql-bin.000002
Slave_IO_Running: Yes
Slave_SQL_Running: Yes
```

Gambar 13  
Status Slave Pada Server Utama

```
mysql> show master status;
+-----+-----+-----+-----+
| File           | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+-----+-----+-----+-----+
| mysql-bin.000003 | 3507    | qtakademik,qtakademik_portal,qtregistrasi,grfinansi_pembayaran | mysql,test |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

Gambar 14  
Status Master Pada Server Redundan

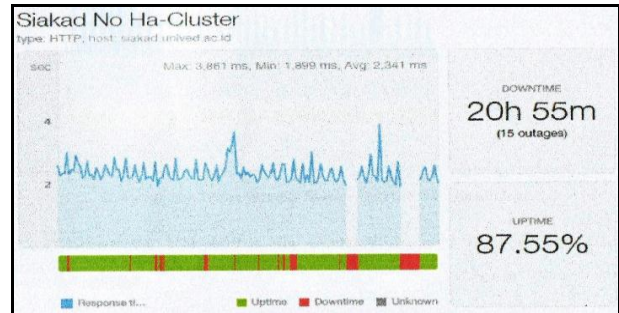
```
mysql> show slave status \G
***** 1. row *****
Slave_IO_State: Waiting for master to send event
Master_Host: 192.168.1.201
Master_User: user_replikasi
Master_Port: 3306
Connect_Retry: 60
Master_Log_File: mysql-bin.000002
Read_Master_Log_Pos: 5283
Relay_Log_File: mysqld-relay-bin.000048
Relay_Log_Pos: 251
Relay_Master_Log_File: mysql-bin.000002
Slave_IO_Running: Yes
Slave_SQL_Running: Yes
```

Gambar 15  
Status Slave Pada Server Redundan

## Pengujian

### Pengujian Availability (Ketersediaan)

Pengujian ini dilakukan menggunakan alat berupa *website monitoring*. Penulis menggunakan layanan dari <http://pingdom.com> sebagai *website monitoring*. Pengujian dilakukan dengan cara mendaftarkan url sistem informasi akademik sebelum pengembangan dan setelah pengembangan ke situs <http://pingdom.com>. Monitoring dilakukan selama 1 bulan. Tingkat ketersediaan *server* sebelum pengembangan dapat dilihat pada gambar 16.



Gambar 16  
Tingkat Availability Server Sebelum Pengembangan  
Sedangkan tingkat availability server setelah pengembangan terlihat pada gambar 17.



Gambar 17  
Tingkat Availability Server Setelah Pengembangan

Selain dengan menggunakan tool [pingdom.com](http://pingdom.com) untuk menguji failover server dengan membuat skenario dengan mengakses *server* dalam kondisi normal. Sehingga akses siakad dilayani server utama (Server1). Tampilan utama sistem informasi akademik terlihat pada gambar 18.

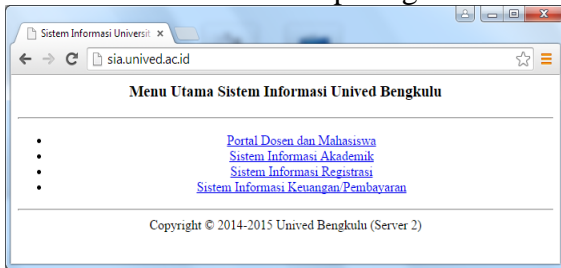


Gambar 18  
Halaman Menu Siakad (Dilayani Server 1)

Dalam kondisi normal, url <http://sia.unived.ac.id> dilayani oleh *primary server*. Setelah itu diskenarioikan *primary server* crash dengan cara mematikan *service heartbeat* atau dapat dilakukan *shutdown* pada *primary*



server, sehingga tampilan menu utama sistem informasi akademik terlihat pada gambar 19.



Gambar 19  
Halaman Menu Siakad (Dilayani Server 2)

### Pengujian Konsistensi Data Skenario I

Melakukan *input* nilai pada server utama melalui aplikasi sistem portal akademik, kemudian membuat server utama *down*, sehingga *request* aplikasi dilayani oleh server *backup*. Setelah itu penulis mengamati data nilai yang ada pada server *backup*. Dari hasil pengujian pada skenario I, terlihat data yang ada pada server *backup* sama dengan data yang ada pada server utama.

Perbandingan data masing-masing server pada skenario I terlihat pada gambar 20.

NO	AKSI	NIM	NAMA	KE	NILAI	BOBOT	KODE	APLIKASI	PENYUBUR	NAMA	PENYUBUR	TANGGAL
1	Ubah nilai	12110001	PERJUTRI PUTRA	1	4.00		A	Portal	HARI ASPRIYONO	HARI ASPRIYONO	27-02-2015	22-02-11
2	Ubah nilai	13110001	TRVENA KARTIKA WP	1	3.00		B	Portal	HARI ASPRIYONO	HARI ASPRIYONO	27-02-2015	22-02-11
3	Ubah nilai	13110002	DIAN WISNU SAPUTRA	1	2.00		C	Portal	HARI ASPRIYONO	HARI ASPRIYONO	27-02-2015	22-02-11
4	Ubah nilai	13110003	MESI HARTATI	1	4.00		A	Portal	HARI ASPRIYONO	HARI ASPRIYONO	27-02-2015	22-02-11
5	Ubah nilai	13110004	ZULFA ISTIGOMAH	1	2.00		C	Portal	HARI ASPRIYONO	HARI ASPRIYONO	27-02-2015	22-02-11

Gambar 20  
Data Nilai Pada Server Utama

Sedangkan data pada server *backup* terlihat pada gambar 21.

NO	AKSI	NIM	NAMA	KE	NILAI	BOBOT	KODE	APLIKASI	PENYUBUR	NAMA	PENYUBUR	TANGGAL
1	Ubah nilai	12110001	PERJUTRI PUTRA	1	4.00		A	Portal	HARI ASPRIYONO	HARI ASPRIYONO	27-02-2015	22-02-11
2	Ubah nilai	13110001	TRVENA KARTIKA WP	1	3.00		B	Portal	HARI ASPRIYONO	HARI ASPRIYONO	27-02-2015	22-02-11
3	Ubah nilai	13110002	DIAN WISNU SAPUTRA	1	2.00		C	Portal	HARI ASPRIYONO	HARI ASPRIYONO	27-02-2015	22-02-11
4	Ubah nilai	13110003	MESI HARTATI	1	4.00		A	Portal	HARI ASPRIYONO	HARI ASPRIYONO	27-02-2015	22-02-11
5	Ubah nilai	13110004	ZULFA ISTIGOMAH	1	2.00		C	Portal	HARI ASPRIYONO	HARI ASPRIYONO	27-02-2015	22-02-11

Gambar 21  
Data Nilai Pada Server Backup

Dari gambar 20 dan gambar 21 menunjukkan bahwa data yang ada pada server utama dan server *backup* adalah sama,

kesamaan data juga terlihat pada kolom tanggal perubahan. Kesamaan tanggal perubahan ini terjadi karena *query* yang dilakukan pada database server utama disimpan dalam log yang kemudian dikirimkan ke database pada server *backup*.

### Skenario II

Skenario ini dilakukan pada saat kondisi server utama *down*, dan *request* aplikasi dilayani oleh server *backup*. Dilakukan perubahan nilai pada database melalui aplikasi di server *backup*, kemudian dalam waktu 1 jam server utama diaktifkan sehingga *request* aplikasi dilayani kembali oleh server utama. Selanjutnya penulis mengamati data nilai pada aplikasi pada server utama, hasil pengamatan penulis menunjukkan data nilai di server utama juga telah berubah seperti pada server *backup*. Adapun data nilai setelah server utama aktif kembali terlihat pada gambar 22

NO	AKSI	NIM	NAMA	KE	NILAI	BOBOT	KODE	APLIKASI	PENYUBUR	NAMA	PENYUBUR	TANGGAL
1	Ubah nilai	12110001	PERJUTRI PUTRA	1	4.00		A	Portal	HARI ASPRIYONO	HARI ASPRIYONO	27-02-2015	22-02-11
2	Ubah nilai	13110001	TRVENA KARTIKA WP	1	3.00		B	Portal	HARI ASPRIYONO	HARI ASPRIYONO	27-02-2015	22-02-11
3	Ubah nilai	13110002	DIAN WISNU SAPUTRA	1	2.00		C	Portal	HARI ASPRIYONO	HARI ASPRIYONO	27-02-2015	22-02-11
4	Ubah nilai	13110003	MESI HARTATI	1	4.00		A	Portal	HARI ASPRIYONO	HARI ASPRIYONO	27-02-2015	22-02-11
5	Ubah nilai	13110004	ZULFA ISTIGOMAH	1	2.00		C	Portal	HARI ASPRIYONO	HARI ASPRIYONO	27-02-2015	22-02-11

Gambar 22  
Data Nilai Setelah Perubahan

### Kesimpulan

Berdasarkan pembahasan dan pengujian yang dilakukan pada implementasi *high availability clustering (ha-cluster)* dan *mysql database replication* pada server siakad Universitas Dehasen Bengkulu, maka dapat ditarik kesimpulan sebagai berikut:

1. Penerapan *high availability clustering* dapat meningkatkan ketersediaan (*availability*) pada server sistem informasi akademik.
2. Sistem dapat menjalankan fungsinya baik dalam keadaan normal dan melakukan failover ketika terjadi *crash/down* pada server utama (*primary server*) dan sebaliknya.
3. Penerapan *mysql database replication* dapat mempertahankan konsistensi data pada kedua server sehingga data pada *primary server* sama dengan data pada *secondary server*.

4. Penerapan *high availability clustering* dan *mysql database replication* menjadi salah satu alternatif untuk mengatasi masalah ketersediaan akses (*availability*) terhadap sistem informasi akademik dan *backup* data untuk mengatasi kehilangan data pada *server*. *Engineering and Technology*, 4, 723-729.

#### **Daftar Pustaka**

- A B M Moniruzzaman, M. W., MD. Sadekur Rahman 2014. A High Availability Clusters Model Combined with Load Balancing and Shared Storage Technologies for Web Servers. 5.
- Gadir, O. M., Subbanna, K., Vayyala, A. R., Shanmugam, H., Bodas, A. P., Tripathy, T. K., Indurkar, R. S. & RAO, K. H. 2005. High-availability cluster virtual server system. Google Patents.
- Pribadi, P. T. 2013. Implementasi High-Availability VPN Client Pada Jaringan Komputer Fakultas Hukum Universitas Udayana. *Jurnal Ilmu Komputer, Program Studi Teknik Informatika, Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Udayana*, 6.
- Rai, A.K., Kaur, K. (2014). "A Comparative Analysis: Grid, Cluster and Cloud Computing." *International Journal of Advanced Research in Computer and Communication Engineering*. M.Tech Scholar, Department of Computer Science and Engineering. Lovely Professional University. Phagwara. India.
- Tawar, S. Wahyuningsih. 2011. Perbandingan Metode Backup Database MySQL antara Replikasi dan MySQLDump. *Program Studi Sistem Informasi, Universitas Ahmad Dahlan*, 1.
- Zakarya, M., Ali Khattak, M., Ur Rahman, I. & Ali Khan, A. 2012. Grid High Availability and Service Security Issues with Solutions. *International Journal of*