

IMPLEMENTASI SISTEM BACKUP OTOMATIS VIRTUAL PRIVATE SERVER DENGAN CRONTAB

I Putu Eka Suparwita

Program Studi Teknik Informatika, Jurusan Ilmu Komputer,
Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Udayana
Email: iputu.eka@cs.unud.ac.id

ABSTRAK

Virtual Private Server merupakan suatu *environment* yang berupa suatu program atau sistem operasi yang tidak ada secara fisik namun dapat dijalankan dalam *environment* lain. Dalam konteks ini, *virtual server* disebut “*guest*” sementara *environment* yang menjalankannya disebut “*host*”. Permasalahan yang sering muncul dalam *virtual private server* yaitu hilang atau rusaknya data-data dan konfigurasi karena *server* mengalami kerusakan atau *down*. Sistem *backup* merupakan solusi dalam mencegah kehilangan data yang tersimpan dalam *virtual private server*. Karena dengan *backup*, data-data yang tersimpan dalam *server* dapat diduplikasi dan dicadangkan sehingga dapat disimpan dan diambil kembali saat diperlukan. *Backup* dapat dilakukan secara berkala pada *server*. Namun untuk mewujudkan hal tersebut admin secara berkala harus melakukan pengecekan serta backup pada *server*. Untuk mempermudah tugas admin dalam mem-backup server, sistem backup virtual private server yang terdapat pada *OpenVZ* dapat diintegrasikan dengan *crontab* yaitu suatu perintah yang mampu melaksanakan tugas secara berkala untuk melakukan *backup virtual private server* sesuai dengan waktu yang telah ditentukan.

Kata Kunci : *Virtual Private Server, OpenVZ, Crontab*

ABSTRACT

Virtual Private Server is an environment in the form of a program or operating system that does not exist physically but can be run in other environment. In this context, virtual server called "guest" while running environment called "hosts". Problems often arise in the virtual private server is lost or damaged data and configurations for servers damaged or down. System backup is a solution to prevent the loss of data stored in the virtual private server. Because of the backup, the data stored in the server can be duplicated and backed up so that it can be stored and retrieved when needed. Backups can be performed at regular intervals on the server. But to achieve this admin should regularly checks and backups on the server. To simplify admin tasks in backup servers, virtual private server backup system contained on OpenVZ crontab that can be integrated with a command that can carry out tasks to perform periodic backups of virtual private server according to the specified time.

Keywords : *Virtual Private Servers, OpenVZ, Crontab*

PENDAHULUAN

Dalam dunia teknologi yang berkembang sangat pesat saat ini, kebutuhan akan sumber daya terutama

perangkat keras (*hardware*) akan semakin bertambah. Dengan bertambahnya kebutuhan tersebut (*hardware*) maka akan mempengaruhi jumlah biaya yang harus

dikeluarkan untuk membangun sistem tersebut. Virtualisasi merupakan salah satu solusi yang dapat dipilih dalam mengatasi masalah tersebut. Dengan adanya teknologi virtualisasi sebuah *resources hardware* dapat digunakan oleh beberapa sistem operasi sekaligus secara bersamaan. Tentunya hal ini sangatlah membantu dalam pembuatan suatu sistem yang memerlukan banyak *host* ataupun *server*, namun tidak memerlukan alokasi biaya yang besar dalam penyediaan perangkat keras atau *hardware*. Namun permasalahan yang kerap kali terjadi yaitu hilang atau rusaknya data-data atau konfigurasi yang tersimpan pada mesin virtual pada saat server mengalami kerusakan atau down. Hal inilah yang mendasari diciptakannya sebuah solusi yang dapat mengatasi masalah tersebut.

Teknologi *backup* merupakan solusi yang dapat dipilih dalam mengatasi masalah tersebut. Dengan adanya teknologi backup, data-data dari sebuah sistem dapat di duplikasi dan kemudian dicadangkan sehingga nantinya data cadangan tersebut dapat digunakan kembali pada saat sistem utama mengalami kegagalan. Tentunya hal ini sangatlah membantu dalam penanganan bencana yang tidak diinginkan dalam waktu yang singkat. Tugas admin pun dapat dipermudah dengan adanya sistem backup otomatis secara berkala yang disebut dengan *crontab* yang dapat dikonfigurasi sesuai dengan kebutuhan.

Virtual Private Server yang diimplementasikan yaitu menggunakan VPS OpenVZ, dengan sistem backup *vzdump* yang terintegrasi dengan *crontab*. pemilihan OpenVZ sebagai solusi dalam penerapan teknologi virtualisasi karena OpenVZ merupakan software yang berlisensi *free*.

MATERI DAN METODE

2.1 Virtual Server

Virtualisasi merupakan metode untuk membuat sesuatu menjadi lepas dari ketergantungan secara fisik. Misalkan virtual server adalah komputer, yang sebenarnya hanya berupa file di hard disk.

Dengan virtualisasi, maka sebuah komputer (fisik) bisa menjalankan banyak komputer virtual sekaligus pada saat yang bersamaan.

Virtual server adalah suatu environment, biasanya sebuah program atau sistem operasi, yang tidak ada secara fisik tetapi dijalankan dalam environment lain. Dalam konteks ini, virtual server disebut "*guest*" sementara environment yang menjalankannya disebut "*host*". Ide dasar dari virtual server adalah mengabstraksi perangkat keras dari satu komputer (CPU, memori, disk, dst) ke beberapa environment eksekusi, sehingga menciptakan illusi bahwa masing-masing environment menjalankan komputernya (terpisah) sendiri. *Virtual Private Server* (VPS) muncul karena adanya keinginan untuk menjalankan banyak sistem operasi pada satu komputer. Teknologi *virtual server* memiliki banyak kegunaan seperti memungkinkan konsolidasi perangkat keras, memudahkan *recovery sistem*, dan menjalankan perangkat lunak terdahulu. Salah satu penerapan penting dari teknologi VM adalah integrasi lintas platform. Beberapa penerapan lainnya yang penting adalah:

1. Konsolidasi server

Jika beberapa server menjalankan aplikasi yang hanya memakan sedikit sumber daya, VPS dapat digunakan untuk menggabungkan aplikasi-aplikasi tersebut sehingga berjalan pada satu server saja, walaupun aplikasi tersebut memerlukan sistem operasi yang berbeda-beda. Otomasi dan konsolidasi lingkungan pengembangan dan testing. Setiap VPS dapat berperan sebagai lingkungan yang berbeda, ini memudahkan pengembang sehingga tidak perlu menyediakan lingkungan tersebut secara fisik.

2. Menjalankan perangkat lunak terdahulu

Sistem operasi dan perangkat lunak terdahulu dapat dijalankan pada sistem yang lebih baru. Memudahkan *recovery system*. Solusi virtualisasi dapat dipakai untuk rencana *recovery system* yang memerlukan portabilitas dan fleksibilitas antar platform.

2.2 OpenVZ

OpenVZ adalah sebuah otomasi server lengkap dan solusi virtualisasi yang dikembangkan oleh Swsoft. *OpenVZ* menciptakan berbagai *Virtual Private Server* (VPS) yang terisolasi pada satu server fisik, berbagi perangkat keras dan manajemen pengguna dengan efisiensi maksimum. Setiap VPS melakukan dan mengeksekusi persis seperti server yang berdiri sendiri, untuk para pengguna aplikasi dapat reboot secara independen dan memiliki akses root, user, alamat IP, memori, proses, file, aplikasi, sistem perpustakaan, dan file konfigurasi. Keunggulan dari desain yang efisien dari *OpenVZ* membuatnya menjadi virtualisasi pilihan yang tepat untuk *server* produksi dengan live aplikasi dan live data. Kemampuan VPS *OpenVZ* adalah :

1. Dynamic Real-time Partitioning – Partisi sebuah server fisik ke puluhan VPS, masing-masing dengan fungsi server penuh.
2. Resource Management – Menetapkan dan mengontrol sumber daya dan parameter VPS serta kembali mengalokasikan sumber daya secara real-time.
3. Mass Management – Mengatur banyak server fisik dan Virtual Private Server dalam satu cara.

OpenVZ memberikan solusi yang komprehensif untuk Hosting Service Provider yang memungkinkan untuk memiliki ratusan pelanggan dengan masing-masing fitur lengkap virtual private server (Virtual Private Server) berbagi satu server fisik. Memberikan setiap pelanggan dengan jaminan Quality of Service. Transparan memindahkan pelanggan dan lingkungan mereka antara server, tanpa konfigurasi manual ulang. Jika sejumlah dedicated server Linux digunakan dalam suatu perusahaan, yang masing-masing berjalan layanan tertentu, dapat digunakan

OpenVZ untuk mengkonsolidasikan semua server ini ke satu komputer tanpa kehilangan sedikit informasi berharga dan tanpa mengurangi performa. Virtual Private Server berperilaku seperti server yang berdiri sendiri dimana :

1. Setiap VPS memiliki proses sendiri, pengguna, file dan menyediakan akses shell penuh untuk root.
2. Setiap VPS memiliki alamat IP sendiri, nomor port, penyaringan dan aturan routing sendiri.
3. Setiap VPS dapat memiliki konfigurasi sendiri untuk sistem beserta aplikasi perangkat lunak sendiri, dan versi sendiri sistem terhadap perpustakaan.
4. Dimungkinkan untuk menginstall atau menyesuaikan perangkat lunak paket di dalam VPS independen dari VPS lain atau sistem host.

Template OS di *OpenVZ* pada dasarnya merupakan kumpulan paket dari beberapa distribusi Linux yang digunakan untuk mengisi satu atau lebih VPS. Dengan *OpenVZ*, distribusi yang berbeda dapat bersama-sama ada di satu hardware, sehingga beberapa OS template yang tersedia. Template OS terdiri dari sistem program, perpustakaan, dan *script* yang dibutuhkan untuk boot dan menjalankan sistem (VPS), serta beberapa sangat dasar dan utilitas aplikasi. Aplikasi seperti compiler dan SQL server biasanya tidak dimasukkan ke dalam template OS.

2.3 Crontab

Crontab adalah sebuah perintah yang sangat berguna untuk menjalankan tugas-tugas yang terjadwal, sehingga akan mengurangi waktu administrasi. Selain *crontab*, ada juga perintah lain seperti *anacron* dan *at*. *Anacron* digunakan untuk melakukan penjadwalan suatu perintah untuk komputer yang tidak selalu menyala terus menerus. *Anacron* menggunakan

interval waktu harian, mingguan, dan bulanan. Sedangkan perintah `at` menjalankan suatu tugas sekali pada satu waktu. Yang paling sering digunakan adalah *crontab*, karena lebih serba guna, dan dapat diatur untuk berjalan pada sembarang interval waktu.

Dalam melakukan administrasi sistem, pengaturan *cron* dilakukan melalui *file crontab*, yang berisi jadwal waktu dan *script* yang harus dieksekusi. *System Linux* memiliki *file crontab default*, yaitu */etc/crontab*, yang akan menjalankan beberapa *script* pada waktu yang telah ditentukan, misalnya setiap jam, harian, mingguan, dan bulanan.

Terdapat dua buah *file* yang menentukan user mana yang bisa menggunakan *crontab*: */etc/cron.allow* dan */etc/cron.deny*. Biasanya, hanya ada *file cron.deny* pada sistem, dan jika *file* ini ada, dan terdapat nama user di dalamnya (satu user per baris), maka user tersebut tidak diperbolehkan menggunakan perintah *crontab*. Jika terdapat *file cron.allow*, maka hanya *user* yang namanya terdapat pada *file* ini yang diperbolehkan menggunakan perintah *crontab*. Pada *file crontab*, terdapat enam *field* untuk setiap *entry*, dan masing-masing *field* dipisahkan oleh spasi atau tab. Lima *field* pertama menentukan kapan perintah akan dijalankan. *Field* ke-enam adalah perintah yang akan dijalankan. Sebagai contoh *script* yang digunakan dalam *crontab* adalah sebagai berikut :

```
# min(0-59) hours(0-23) day(1-31) month(1-12) dow(0-6) command
34 2 * * * sh /root/backup.sh
```

Pada contoh tersebut, perintah `“sh /root/backup.sh”` akan dijalankan pada jam 2:34 AM setiap hari. Tanda bintang pada contoh tersebut berarti “semua waktu”. Tanda bintang pada kolom “menit” berarti “jalankan setiap menit”. User bisa melakukan banyak kombinasi pengaturan waktu. Ada cara lain untuk pengaturan waktu: “20-27” menunjukkan *range*; “3,4,7,8” artinya hanya interval tersebut untuk suatu pilihan, dan `*/5` artinya setiap

interval ke lima. Fitur lainnya, setelah *cron* menyelesaikan suatu tugas, kita akan mendapatkan email berisi keluaran dari perintah yang dijalankan.

3. PEMBAHASAN

Dalam melakukan *backup*, admin seringkali direpotkan karena harus mengetikkan perintah *backup* setiap kali akan mem-*backup* suatu sistem yang dalam hal ini adalah *virtual private server* pada *OpenVZ*. *Crontab* merupakan salah satu perintah yang dapat digunakan untuk mengotomatisasi *backup virtual private server* yang dapat disesuaikan dengan kebutuhan. Berikut adalah *script* yang digunakan dalam mem-*backup virtual private server* yang terdapat pada server.

Dalam kasus ini beberapa *virtual private server* yang terdapat pada suatu server *primary* pada IP 172.16.122.11 akan di *backup* ke server *secondary* dengan IP 172.16.122.15. Langkah pertama yang dilakukan yaitu membuat sekumpulan perintah dari setiap *virtual private server* yang akan di *backup* kemudian dirangkum dalam satu *script* yang kemudian dijalankan dengan perintah *crontab* sebagai berikut :

1. Untuk menampilkan ID dari *virtual private server*; ketikkan perintah

```
#vzlist -a
```
2. Dilanjutkan dengan membuat *script backup* pada terminal, dalam hal ini *script* diberi nama *backup.sh*

```
#nano backup.sh
```

kemudian akan terbuka teks editor yang akan diisi *script backup* di dalamnya
3. Isikan *script backup* ke dalam teks editor yang telah dibuka tadi berdasarkan *virtual private server*

yang akan di *backup*. Misal *virtual private server* yang akan di *backup* dengan ID 141, 142, 143, 144, 145, 145, 146, 147, 148, 149, 150, 151, dan 153. Masukkan perintah berikut:

```
/usr/sbin/vzdump --
compress --suspend 141

/usr/sbin/vzdump --
compress --suspend 142

/usr/sbin/vzdump --
compress --suspend 143

/usr/sbin/vzdump --
compress --suspend 144

/usr/sbin/vzdump --
compress --suspend 145

/usr/sbin/vzdump --
compress --suspend 146

/usr/sbin/vzdump --
compress --suspend 147

/usr/sbin/vzdump --
compress --suspend 148

/usr/sbin/vzdump --
compress --suspend 149

/usr/sbin/vzdump --
compress --suspend 150

/usr/sbin/vzdump --
compress --suspend 151

/usr/sbin/vzdump --
compress --suspend 153
```

Dimana `/usr/sbin/` adalah lokasi dimana *script* `vzdump` berada.

4. Kemudian masukkan perintah *rsync* untuk mengirimkan hasil *backup* dan log yang berada di `/var/lib/vz/dump` ke server *secondary* dengan perintah

```
rsync -au
/var/lib/vz/dump/*.*
root@172.16.122.15:/home/b
ackup/ >
/root/backup/log/log-
$(date +%Y-%m-%d).txt

rsync -au
/root/backup/log/*.*
root@172.16.122.15:/home/a
ckup/log
```

Dengan hasil *rsync* dari *primary server* yang tersimpan di direktori `/home/backup/` dan log yang tersimpan di direktori `/root/backup/log`

5. Kemudian masukkan perintah hapus jika sistem sudah selesai melakukan *backup* dan *rsync* agar tidak banyak menggunakan *space* pada *primary server*

```
rm -rf /var/lib/vz/dump/*.*
```

Simpan file kemudian masukkan *script* tersebut ke dalam *crontab* dan sesuaikan dengan jadwal backup yang di tentukan. Buka *crontab* dengan mengetikkan perintah

```
#crontab -e
```

Maka akan terbuka editor *crontab* kemudian isikan dengan lokasi *script backup* yang telah dibuat tadi beserta tentukan waktu untuk menjalankan perintah *backup*

```
#backup open vz ke ip
172.16.122.15
00 22 * * * /root/salin.sh
```

Terlihat 5 deret tanda bintang dimana menandakan waktu yang di-*set* untuk menjalankan perintah *salin.sh* yang berada pada direktori `/root/` bintang pertama menandakan menit, bintang kedua menandakan jam, bintang ketiga menandakan setiap tanggal, bintang keempat menandakan bulan, dan bintang kelima menandakan minggu. Jadi dalam perintah di atas menandakan *script* akan dijalankan setiap pukul 22.00 setiap harinya.

6. Jika telah selesai memasukkan perintah ke dalam *crontab* simpan dengan menekan kombinasi *Ctrl+x*
7. Restart *crontab* dengan perintah pada terminal

```
#/etc/init.d/cron/ restart
```

Kemudian enter. Untuk melihat daftar perintah yang berjalan di *crontab*, dapat dilakukan dengan mengetikkan perintah

```
#crontab -l
```

Akan tampil seperti gambar berikut

```
root@FH-FILE:~# crontab -l
# m h dom mon dow   command
10 10 * * * rsync -av /home/vz/ 172.16.122.15:/home/backup/vz/
10 10 * * * rsync -av /home/nuffic/ 172.16.122.22:/home/backup/nuffic/
*/20 * * * /root/remove-route.sh

#backup open vz ke ip 172.16.122.15
32 15 * * * /root/salin.sh
00 22 * * * /root/salin2.sh

#15 14 * * * /root/test.sh
#15 09 * * * /root/backup1.sh
#10 20 * * * /root/backup2.sh

#50 11 * * * /root/mutt.sh
root@FH-FILE:~#
```

Gambar 1. Job Crontab

8. Crontab telah dijalankan dan hasil backup dapat dilihat pada *server secondary* setelah proses backup selesai.

SIMPULAN

Simpulan yang dapat diambil dari penelitian yang telah dilakukan adalah sebagai berikut:

1. *OpenVz* merupakan software mesin virtual yang di dalamnya terdapat *tools vzdump* yang mampu dalam mem-backup suatu *virtual server* pada *OpenVZ* dimana *tools* ini juga dapat dijalankan secara berkala dengan mengkombinasikannya dengan *Crontab* yang merupakan sebuah perintah yang sangat berguna untuk menjalankan tugas-tugas yang terjadwal.
2. *Backup Otomatis Virtual Private Server* yang telah penulis buat sudah dapat beroperasi sesuai tujuan pembuatannya yaitu mampu

melakukan *backup* secara otomatis pada *Virtual Private Server*.

KEPUSTAKAAN

- [1] Bayu, T.I., Widiyari, I.R., Chandra, D.W. 2010, Penerapan Teknologi *Virtualisasi Tingkat Sistem Operasi* pada *Server Linux Ubuntu 8.04* Menggunakan *OpenVZ*, Vol.7 Universitas Kristen Satya Wacana, Salatiga.
- [2] Joe, Putra. 2010. Penjadwalan perintah menggunakan Utilitas Cron [Online] Tersedia <http://slackware.kendari.linux.or.id/content/penjadwalan-perintah-menggunakan-utilitas-cron> [24 Agustus 2011]
- [3] Rogier Dittner and David Rule. 2007. *The Best Damn Server Virtualization Book Period: Including Vmware, Xen, and Microsoft Virtual Server*. Syngress Publishing.
- [4] Swsoft.inc. 2005. *OpenVZ User Guide Version 2.7.0-8* [Online] Tersedia <http://ftp.openvz.org/doc/OpenVZ-Users-Guide.pdf> [18 Agustus 2012]
- [5] Time, Falko. 2008. *Clone/Back Up/Restore OpenVZ VMs With vzdump* [Online] Tersedia <http://www.howtoforge.com/clone-back-up-restore-openvz-vm-with-vzdump> [24 Agustus 2012]