

Cara Menjalankan Program

1. Siapkan Beberapa Server yang berbeda di virtual box, server tersebut yang akan digunakan untuk menjalankan proses recovery nantinya.
2. Siapkan juga server yang akan digunakan untuk memmanagement server-server yang ada di virtual box.
3. Setelah semua server telah siap kemudian hubungkan server menggunakan ansible, sebelum menghubungkan install dulu ansiblenya dengan perintah apt-get install ansible
4. Setelah ansible terinstall kemudian hubungkan dengan ssh keygen dan beri akses ke masing-masing server

```
root@DESKTOP-ICCB079:/home/lum# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
/root/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:hzilfxbnpu34Yb+z8Dj/+R9+Y1YC02ZopHA8GtpY0gs root@DESKTOP-ICCB079
The key's randomart image is:
+----[RSA 3072]-----+
|
|   . .
|  E = + .
|   B.* + o
|  o++.. = +
| + S o..=
|   o . + . .
|   . o * o.
|   o *. *o++
|   oo=oOBo
|
+-----[SHA256]-----+
root@DESKTOP-ICCB079:/home/lum#
```

Setelah itu beri akses ke masing-masing ip server

```
root@DESKTOP-ICCB079:/home/lum# ssh-copy-id -i root@192.168.1.15
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: ERROR: ssh: connect to host 192.168.1.15 port 22: Resource temporarily unavailable
root@DESKTOP-ICCB079:/home/lum# ssh-copy-id -i root@192.168.1.18
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: ERROR: ssh: connect to host 192.168.1.18 port 22: Resource temporarily unavailable
```

5. Setelah itu cek menggunakan ansible untuk mengetahui apakah masing-masing server sudah terhubung dengan ansible atau belum dengan perintah `ansible -m ping all`

```
server01 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": false,
  "ping": "pong"
}
```

```
server03 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": false,
  "ping": "pong"
}
```

6. Setelah masing-masing server terhubung dengan ansible, selanjutnya install aplikasi yang di butuhkan seperti PHP, Mysql, Python dan lain-lain.
7. Setelah aplikasi yang dibutuhkan sudah terinstall selanjutnya jalankan proses recovery yang meliputi backup, copy dan import baik databasenya maupun web servernya

8. Proses *Backup* Database merupakan proses *ansible-Playbook* dimana didalam konfigurasinya terdapat beberapa perintah untuk melakukan suatu proses *backup* database dari server 1 ke server host. Proses *backup* database ditunjukkan pada gambar dibawah ini

```
root@DESKTOP-ICCB079:/etc/ansible# ansible-playbook backupdb.yml
PLAY [server01] *****
TASK [Gathering Facts] *****
[DEPRECATION WARNING]: Distribution Ubuntu 16.04 on host server01 should use /usr/bin/python3, but is using
/usr/bin/python for backward compatibility with prior Ansible releases. A future Ansible release will default to using
the discovered platform python for this host. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information. This feature
will be removed in version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in
ansible.cfg.
ok: [server01]
TASK [create a backup] *****
changed: [server01]
PLAY RECAP *****
server01 : ok=2 changed=1 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
root@DESKTOP-ICCB079:/etc/ansible#
```

9. Proses *backup* web server merupakan proses *ansible Playbook* dimana didalam konfigurasinya terdapat beberapa perintah untuk melakukan suatu proses mem-*backup* sekumpulan data web server dari server 1 ke server host. Proses *backup* web server ditunjukkan pada gambar dibawah ini :

```
root@DESKTOP-ICCB079:/etc/ansible# ansible-playbook backupweb.yml
PLAY [server01] *****
TASK [Gathering Facts] *****
[DEPRECATION WARNING]: Distribution Ubuntu 16.04 on host server01 should use /usr/bin/python3, but is using
/usr/bin/python for backward compatibility with prior Ansible releases. A future Ansible release will default to using
the discovered platform python for this host. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information. This feature
will be removed in version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in
ansible.cfg.
ok: [server01]
TASK [Ansible zip directory example] *****
ok: [server01]
TASK [Copy dari Remote ke Host] *****
changed: [server01]
PLAY RECAP *****
server01 : ok=3 changed=1 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
```

Setelah itu dapat dilihat data dari web server sudah ter *backup* pada direktori yang diinginkan. Dapat dilihat pada gambar dibawah ini :

```
root@DESKTOP-ICCB079:/etc/ansible# cd /tmp/
root@DESKTOP-ICCB079:/tmp# ls
backup.zip server01 tmpn9elzblp
root@DESKTOP-ICCB079:/tmp#
```

10. Selanjutnya proses *recovery* web server merupakan proses *ansible-Playbook* dimana didalam konfigurasinya terdapat beberapa perintah untuk melakukan suatu proses *recovery* web server dari server host ke server 2. Setelah proses *recovery* web server selesai kemudian data dari wes server tersebut akan di *compress* secara otomatis. Proses *recovery* web server ditunjukkan pada gambar dibawah ini :

```
root@DESKTOP-ICCB079:/etc/ansible# ansible-playbook recoveryweb.yml
PLAY [server03] *****
TASK [Gathering Facts] *****
[DEPRECATION WARNING]: Distribution Ubuntu 16.04 on host server03 should use /usr/bin/python3, but is using
/usr/bin/python for backward compatibility with prior Ansible releases. A future Ansible release will default to using
the discovered platform python for this host. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information. This feature
will be removed in version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in
ansible.cfg.
ok: [server03]
TASK [Install unzip] *****
ok: [server03]
TASK [Copy File .zip] *****
ok: [server03]
TASK [Ansible zip directory example] *****
ok: [server03]
PLAY RECAP *****
server03 : ok=4  changed=0  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
```

Dan gambar dibawah ini merupakan hasil dari *recovery* tersebut akan di *compress* secara otomatis pada server 2.

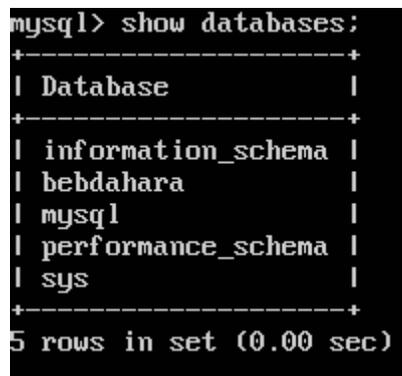
```
root@lum:/# cd /var/www/
root@lum:/var/www# ls
backup.zip html lum.com
root@lum:/var/www# _
```

11. Proses terakhir yaitu *import* database ke mysql menggunakan ansible *Playbooks* juga.

Proses tersebut dilakukan secara otomatis sehingga tinggal menunggu prosesnya selesai dan web nya dapat dijalankan. Proses *import* database ditunjukkan pada gambar dibawah ini :

```
root@DESKTOP-ICCB079:/etc/ansible# ansible-playbook importdb.yml
PLAY [server03] *****
TASK [Gathering Facts] *****
[DEPRECATION WARNING]: Distribution Ubuntu 16.04 on host server03 should use /usr/bin/python3, but is using
/usr/bin/python for backward compatibility with prior Ansible releases. A future Ansible release will default to using
the discovered platform python for this host. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information. This feature
will be removed in version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in
ansible.cfg.
ok: [server03]
TASK [Copy file sql] *****
changed: [server03]
TASK [import the database] *****
changed: [server03]
PLAY RECAP *****
server03 : ok=3  changed=2  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
root@DESKTOP-ICCB079:/etc/ansible#
```

Setelah itu dapat dilihat pada server 2 database nya sudah ada dan web nya dapat kita jalankan. Hasil database dapat dilihat pada gambar dibawah ini :



```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| bebdahara |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.00 sec)
```

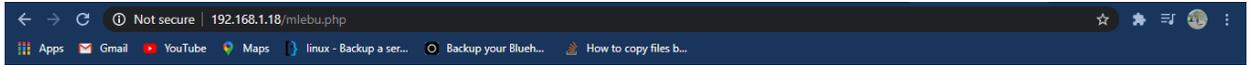
12. Lalu untuk Proses *backup* juga dapat dilakukan secara otomatis sesuai jam yang di tentukan jadi setiap jam yang telah di tentukan system akan secara teratur melakukan proses *backup* dengan menggunakan konfigurasi yang ada di cron job. Proses *backup* otomatis dapat dilihat pada gambar dibawah ini :

```
root@DESKTOP-ICC8079:/etc/ansible# ansible-playbook cronbackup.yml
PLAY [server03] *****
TASK [Gathering Facts] *****
[DEPRECATION WARNING]: Distribution Ubuntu 16.04 on host server03 should use /usr/bin/python3, but is using
/usr/bin/python for backward compatibility with prior Ansible releases. A future Ansible release will default to using
the discovered platform python for this host. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information. This feature
will be removed in version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in
ansible.cfg.
ok: [server03]
TASK [Copy Web Server] *****
changed: [server03]
TASK [Tentukan Waktu] *****
changed: [server03]
PLAY RECAP *****
server03 : ok=3  changed=2  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
```

Setelah itu dapat dilihat pada server 2 file yang telah di *backup* sudah ada dan secara terus menerus akan melakukan *backup* pada server 2 sesuai waktu yang ditentukan. Hasil *backup* otomatis dapat dilihat paad gambar dibawah ini :

```
root@lum:/var/www# ls
backup.zip  html  lum.com
root@lum:/var/www#
```

13. Selanjutnya masukkan ip server 2 yaitu 192.168.1.18 dan dapat dilihat hasilnya seperti dibawah ini menandakan wesb servernya sudah dapat di akses di server 2.



Login

Username

Password

LOG IN

14. Lalu gambar dibawah ini merupakan tampilan setelah login pada web diatas

