**Tugas Day 14**

Instalasi dan Konfigurasi Ansible

pada Sistem Operasi CentOS

**Intro**

Ansible adalah perangkat lunak *open-source* yang dapat difungsikan sebagai *software provisioning*, *configuration management* dan *application deployment*. Ansible bersifat agentless yang mana tidak melibatkan instalasi agent di nodes target. Di sisi lain, ansible hanya menggunakan ssh untuk melakukan koneksi dari controlling machines ke nodes target.

**Skenario**

Pada skenario kali ini, kita diminta untuk membuat *virtual machine* yang difungsikan sebagai *ansible* client dan server. Kemudian, pada *ansible* server akan dijalankan module *ping* untuk mengecek konektivitas terhadap semua *ansible* *client*.

**Spesifikasi**

Berikut ini merupakan spesifikasi ansible client-server yang akan diimplementasikan

|  |
| --- |
| # Ansible server  OS : CentOS 7 Server  IP : 172.25.0.1/24  # Ansible client  OS : CentOS 7 Server  Node :  - node1 (172.25.0.2/24)  - node2 (172.25.0.3/24)  User : centos |

**Instalasi Ansible**

Untuk menginstal package *ansible*, terlebih dahulu tambahkan *repository* *epel*. Kemudian jalankan perintah yum seperti di bawah ini

|  |
| --- |
| $ sudo yum -y install epel-release  $ sudo yum -y install ansible  $ ansible --version |
|  |

**Konfigurasi Inventory dan Host**

Pada terminologi *ansible*, inventory didefinisikan sebagai sekumpulan list atau grup yang melibatkan operasi *task*, *command*, dan *module*. Inventory digunakan untuk mengatur dan mengontrol banyak *nodes* atau *host* sekaligus pada satu waktu.

Sebelum membuat ***inventory file***, kita buat konfigurasi file dasar untuk ansible. Di bawah ini, kita mengatur ***inventory path*** pada file **hosts**. Selain itu, kita juga menonaktifkan ***strict ssh key checking*** saat koneksi SSH pertama kali dilakukan.

|  |
| --- |
| $ vim ansible.cfg |
| [defaults]  inventory = hosts  host\_key\_checking = False |

Selanjutnya, kita buat inventory file bernama **hosts**. Di dalamnya, kita definisikan sebuah blok grup bernama **servers** yang berisikan hostname dari *ansible client* (node1, node2). Selain itu, kita juga mendeklarasikan variabel grup berupa explicit interpreter untuk menghindari *warning* saat ansible dijalankan

|  |
| --- |
| $ vim hosts |
| [servers]  node1  node2  [servers:vars]  ansible\_python\_interpreter=/usr/bin/python |

**Konfigurasi Autentikasi SSH**

Ansiblemenggunakan komunikasi SSH untuk menjalankan sekumpulan **tasks** pada remote device. Oleh karena itu, kita harus memastikan bahwa autentikasi antara host dan *remote device* dapat berjalan dengan baik, salah satunya adalah dengan men generate pasangan *public-private key*.

Untuk mempersingkat deklarasi yang nantinya akan kita digunakan, terlebih dahulu kita definisikan *alias* *name* untuk masing-masing SSH config

|  |
| --- |
| $ vim ~/.ssh/config |
| Host node1  HostName 172.25.0.2  User centos  Host node2  HostName 172.25.0.3  User centos |

Kemudian, kita generate pasangan *public-private key* pada *ansible server* dengan perintah **ssh-keygen**

|  |
| --- |
| $ ssh-keygen -t rsa -b 4096  $ ls ~/.ssh |
|  |

Terakhir, kita unggah file **~/.ssh/id\_rsa.pub** ke masing-masing *remote* *device* dengan perintah sebagai berikut

|  |
| --- |
| $ ssh-copy-id node1  $ ssh-copy-id node2 |
|  |

Apabila proses berjalan dengan benar, sistem akan menggunakan **id\_rsa** untuk melakukan autentikasi terhadap *remote* *device* sehingga kita tidak perlu memasukkan kata sandi saat melakukan SSH. Selain itu, kita juga akan melihat sebuah file bernama **authorized\_keys** yang merupakan pubkey dari *ansible server*

|  |
| --- |
| $ ssh node1 |
|  |

**Pengujian**

Pada tahap pengujian, kita diminta untuk mengecek konektivitas antara *ansible server*dengan *ansible client* menggunakan module **ping** seperti yang tertera di bawah ini:

|  |
| --- |
| $ ansible -m ping all |
|  |

Selain itu, kita bisa juga melakukan *command execution*pada masing-masing *machine*, menggunakan module **shell**

|  |
| --- |
| $ ansible -m shell -a "id" all |
|  |

Terakhir, kita bisa juga melihat manual dari *builtin module* yang disediakan oleh ansible

|  |
| --- |
| $ ansible-doc ping |
|  |