DevSecOps

**Automasi Deployment API**

1. ***Otomatisasi Deployment***

Otomatisasi deployment adalah proses pengiriman aplikasi (seperti API) secara otomatis dari lingkungan pengembangan (development) ke lingkungan produksi (production) menggunakan *tool* seperti Ansible dan Jenkins.

***Tujuan***

* Meminimalkan kesalahan manusia.
* Mempercepat proses rilis.
* Menjaga konsistensi di seluruh server.

1. ***Pentingnya Otomatisasi Deployment***

***Deployment Manual :*** Rentan kesalahan (misalnya, salah konfigurasi, dependensi hilang), serta memakan waktu dan tidak scalable.

***Otomatisasi :*** Cepat dan dapat diulang (repeatable), dapat dilacak dan memiliki versi (versioned), serta aman untuk kolaborasi antar tim.

1. ***Tools***

***Ansible***

* Tool otomatisasi berbasis YAML.
* Agentless (menggunakan SSH).
* Sangat bagus untuk mengelola konfigurasi server dan deployment file.

***Jenkins***

* Tool CI/CD (Continuous Integration/Continuous Delivery) yang popular.
* Dapat membangun pipeline yang mengintegrasikan Git, Ansible, dan notifikasi.
* Visual dan dapat diskrip (scriptable).

1. ***Arsitekur Deployment***

Arsitektur deployment adalah sebagai berikut,

* Pengembang melakukan *commit* kode API terbaru ke Git (misalnya, GitHub).
* Jenkins mendeteksi perubahan dan memicu *pipeline* deployment.
* Jenkins mengeksekusi *Ansible playbook*.
* Ansible terhubung ke *server* API melalui SSH untuk.
  + - Mentransfer *file* aplikasi.
    - Menerapkan konfigurasi *server.*
    - Melakukan *restart* layanan API.

Jenkins Detect Change

GitHub Push

Jenkins Pipeline Executes

Ansible Playbook (API Server)

1. ***Contoh Ansible Playbook***

Berikut adalah contoh *Ansible Playbook* untuk deployment API,

*Playbook* ini,

* Mentransfer *file* JAR Anda ke *server.*
* Melakukan *restart* layanan API Spring Boot secara otomatis.

***Code***

*- name: Deploy Spring Boot API*

*hosts: api\_servers*

*become: yes*

*vars:*

*app\_name: my-api*

*jar\_file: my-api.jar*

*deploy\_path: /opt/apps/my-api*

*tasks:*

*- name: Copy JAR file*

*copy:*

*src: "{{ jar\_file }}"*

*dest: "{{ deploy\_path }}/{{ jar\_file }}"*

*- name: Restart API service*

*systemd:*

*name: my-api*

*state: restarted*

*enabled: yes*

***A screenshot of a computer

AI-generated content may be incorrect.***

1. ***Contoh Jenkins Pipeline***

Berikut adalah contoh *Jenkins Pipeline*,

* + - Mengkloning kode API terbaru dari GitHub.
    - Mengeksekusi *Ansible playbook* untuk melakukan deploy API.

***Code***

pipeline {

    agent any

    stages {

        stage('Checkout') {

            steps {

                git 'https://github.com/user/my-api.git'

            }

        }

        stage('Deploy with Ansible') {

            steps {

                sh 'ansible-playbook deploy.yml -i hosts.ini'

            }

        }

    }

}

**A computer screen with blue and white text

AI-generated content may be incorrect.**