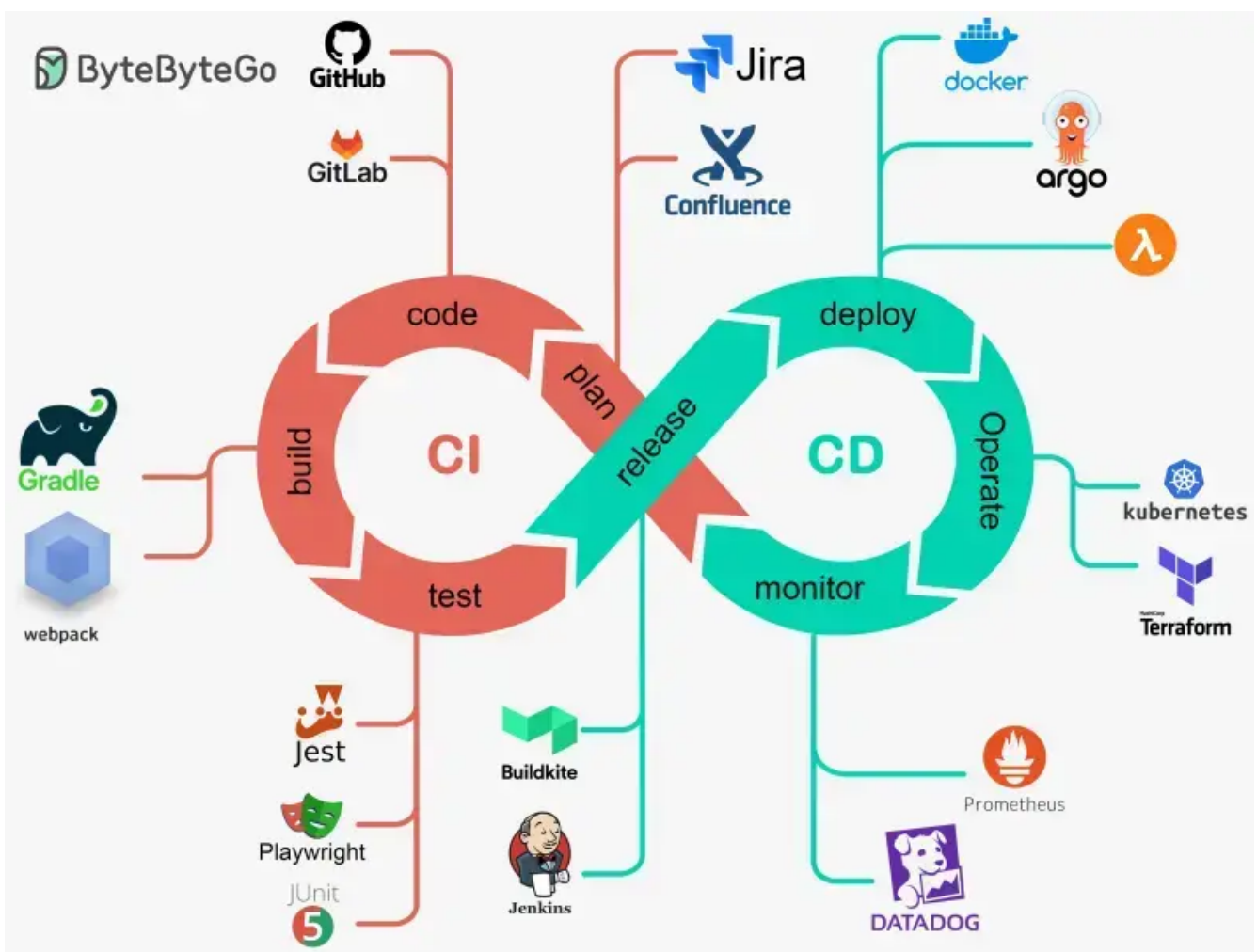


Apa Itu CI/CD dan Bagaimana Menggunakannya di VPS

Rabu, 18 Juni 2025 08:32 WITA | 88 views



CI/CD klikbyte.com

Dalam dunia pengembangan perangkat lunak modern, kecepatan dan akurasi dalam merilis fitur baru menjadi sangat penting. Untuk mendukung hal tersebut, dibutuhkan sistem otomatisasi yang disebut **CI/CD (Continuous Integration / Continuous Delivery** atau

Deployment). Dengan CI/CD, proses pengembangan, pengujian, hingga penyebaran aplikasi dapat dilakukan secara otomatis dan efisien.

Pada artikel ini, kita akan membahas pengertian CI/CD, cara kerjanya, manfaatnya, serta bagaimana Anda bisa menerapkannya di server **VPS (Virtual Private Server)**.

Apa Itu CI/CD?

1. Continuous Integration (CI)

CI adalah praktik di mana pengembang secara rutin melakukan commit kode ke repository bersama, biasanya setiap hari. Setiap perubahan kode akan langsung melalui proses *build* dan *testing* otomatis untuk memastikan tidak ada fitur yang rusak atau terganggu.

2. Continuous Delivery & Continuous Deployment (CD)

- **Continuous Delivery** memastikan kode selalu dalam kondisi siap rilis ke produksi, meskipun penyebarannya masih dilakukan secara manual.
- **Continuous Deployment** melangkah lebih jauh, di mana setiap perubahan kode yang lolos pengujian akan langsung di-*deploy* secara otomatis ke server produksi.

Manfaat Menggunakan CI/CD

- **Rilis Fitur Lebih Cepat** – Perubahan langsung sampai ke pengguna tanpa menunggu lama.
- **Mengurangi Risiko Error** – Proses pengujian otomatis menjaga kualitas kode.
- **Kolaborasi Lebih Efektif** – Semua perubahan tercatat dan mudah dilacak.

- **Otomatisasi Proses Repetitif** – Developer bisa fokus ke pengembangan, bukan operasional.
-

Menerapkan CI/CD di VPS

Menggunakan **VPS** sebagai host CI/CD memberi fleksibilitas tinggi dan kontrol penuh, tanpa harus bergantung pada layanan seperti GitHub Actions atau GitLab CI.

1. Persiapan Lingkungan Server

Pastikan VPS Anda sudah memiliki:

- Sistem Operasi (Ubuntu, Debian, CentOS, dll.)
- Web Server (Apache atau Nginx)
- Database (MySQL atau PostgreSQL)
- Runtime Environment (PHP, Node.js, Python, dll.)

2. Memilih Alat CI/CD

Berikut beberapa alat populer untuk diterapkan di VPS:

- **Jenkins**
 - **GitLab CI Runner**
 - **Drone CI** (bisa terintegrasi dengan Gitea)
 - **GitHub Actions** (dengan self-hosted runner)
-

Contoh Instalasi Jenkins di Ubuntu

```
sudo apt update
sudo apt install openjdk-11-jdk -y

wget -qO - https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo apt-key add -
sudo sh -c 'echo deb http://pkg.jenkins.io/debian-stable binary/ > /etc/apt/sources.list.d/jenkins.list'
sudo apt update
sudo apt install jenkins -y
sudo systemctl start jenkins
sudo systemctl enable jenkins
```

Setelah selesai, akses Jenkins melalui browser di `http://[IP_VPS]:8080`.

3. Buat Repository Git

Gunakan GitHub, GitLab, Bitbucket, atau host Git lokal seperti **Gitea** di VPS Anda sendiri untuk menyimpan kode sumber.

4. Konfigurasi Pipeline

Pipeline adalah urutan proses build ? test ? deploy. Contoh pipeline sederhana untuk aplikasi Node.js:

```
pipeline {
  agent any
  stages {
    stage('Build') {
      steps {
        echo "Building..."
        sh 'npm install'
      }
    }
    stage('Test') {
      steps {
        echo "Testing..."
        sh 'npm run test'
      }
    }
  }
}
```

```
    }
  }
  stage('Deploy') {
    steps {
      echo "Deploying..."
      sh 'scp -r . user@your_vps:/var/www/app'
      sh 'ssh user@your_vps "systemctl restart nginx"'
    }
  }
}
```

5. Integrasi Webhook

Aktifkan webhook pada repository agar setiap push/commit langsung memicu pipeline CI/CD secara otomatis.

6. Monitoring dan Logging

Pastikan alat CI/CD Anda memiliki fitur monitoring. Jenkins, misalnya, menyediakan dasbor untuk melihat status build dan log error jika terjadi kegagalan.

Tips Tambahan

- Gunakan **SSH Key** untuk autentikasi aman ke server produksi.
 - Pertimbangkan menggunakan **Docker** untuk menyederhanakan konfigurasi lingkungan.
 - Simpan konfigurasi pipeline di version control (misalnya file `.jenkinsfile` atau `.gitlab-ci.yml`).
-

Kesimpulan

CI/CD bukan lagi sekadar pilihan, tapi kebutuhan dalam pengembangan perangkat lunak modern. Dengan memanfaatkan VPS, Anda bisa membangun sistem CI/CD yang fleksibel, aman, dan efisien tanpa tergantung layanan cloud berbayar.

Baik Anda bekerja sendiri atau dalam tim, CI/CD akan mempercepat proses rilis dan meningkatkan kualitas aplikasi secara keseluruhan.

Link Artikel:

<https://www.klikbyte.com/read/14/apa-itu-cicd-dan-bagaimana-menggunakannya-di-vps>

www.klikbyte.com