BAB 1
PENDAHULUAN

1.1 Latar Belakang

Transformasi digital memanfaatkan integrasi teknologi informasi (TI) guna menunjang proses internal dan layanan eksternal yang efisien, responsif, serta adaptif terhadap perubahan kebutuhan dan perkembangan teknologi (Pratama & Putri, 2025). Transformasi ini mendorong peningkatan kebutuhan pemanfaatan infrastruktur virtualisasi privat untuk mendukung layanan strategis institusi. Infrastruktur virtualisasi privat menjadi pilihan utama bagi institusi karena memungkinkan kendali penuh atas sumber daya dan data, sekaligus mendukung fleksibilitas penyediaan layanan TI.

Namun demikian, pada banyak infrastruktur virtualisasi privat, proses *provisioning virtual machine* (VM) masih dilakukan secara manual dan belum mengikuti standar konfigurasi yang konsisten. Hal ini menyebabkan waktu *provisioning* yang lama, inkonsistensi konfigurasi antar *virtual machine*, serta meningkatnya risiko kesalahan manusia, terutama pada infrastruktur berskala besar dan kompleks (Joshi, 2024) (Koneru, 2025). Sementara itu, percepatan transformasi digital menuntut layanan TI yang adaptif, responsif, dan berkelanjutan, sehingga membutuhkan mekanisme *provisioning* VM yang cepat dan andal (Moodley, 2025).

Selain aspek efisiensi, keamanan siber merupakan tantangan krusial dalam pengelolaan VM pada infrastruktur virtualisasi privat (Uddin dkk., 2021). Penggunaan mekanisme autentikasi berbasis kata sandi menjadikan infrastruktur rentan terhadap serangan siber berbasis kredensial, seperti *brute force* dan *credential stuffing* (Matzen dkk., 2025). Kerentanan tersebut tidak hanya meningkatkan risiko masuknya pengguna tanpa hak akses ke layanan TI, tetapi juga berpotensi memperlambat pemulihan layanan serta menurunkan citra institusi ketika terjadi insiden keamanan siber (Araujo dkk., 2024). Dalam lingkungan virtualisasi privat yang digunakan secara bersama oleh banyak pengguna, ketiadaan mekanisme pengelolaan hak akses yang terstruktur juga berpotensi menimbulkan konflik penggunaan sumber daya serta meningkatkan risiko pelanggaran

keamanan, sehingga diperlukan pengaturan akses dan pemisahan sumber daya yang jelas antar pengguna.

Latar belakang masalah terkait kecepatan, konsistensi, dan keamanan pada *provisioning* VM memerlukan solusi yang mengintegrasikan otomasi dan autentikasi dalam pemanfaatan infrastruktur virtualisasi privat. Salah satu solusi integrasi adalah dengan pendekatan *Infrastructure as Code* (IaC). IaC menawarkan pemanfaatan infrastruktur virtualisasi privat dalam bentuk kode yang terdokumentasi, tervalidasi, dan dapat direplikasi secara konsisten (Nadagouda, 2025) (Morris, 2020), khususnya pada lingkungan virtualisasi privat berbasis Proxmox VE yang umum digunakan oleh institusi dalam pengelolaan sumber daya *virtual machine*. Melalui pendekatan ini, *provisioning* VM dapat dilakukan secara otomatis berdasarkan spesifikasi standar, sehingga mampu menurunkan waktu *provisioning*, meminimalkan kesalahan manusia, dan menjamin konsistensi konfigurasi (Chauhan, 2024) (Vaggu, 2025).

Namun demikian, sebagian besar implementasi IaC masih berfokus pada pengguna dengan kompetensi teknis yang kuat, seperti *cloud engineer* atau *system administrator* (Pospisil, 2025) (Valtanen, 2023). Implementasi yang mengharuskan pengguna menulis konfigurasi *HashiCorp Configuration Language* (HCL) menjadikan IaC sulit diadopsi oleh pengguna nonteknis. Dalam konteks penelitian ini, pengguna nonteknis didefinisikan sebagai pengguna layanan infrastruktur TI (seperti mahasiswa, dosen, dan peneliti di lingkungan Politeknik Caltex Riau) yang membutuhkan penyediaan mesin virtual secara cepat untuk keperluan praktikum maupun pengujian, namun tidak memiliki keahlian teknis dalam menyusun skrip IaC serta tidak memiliki hak istimewa administratif (*administrative privileges*) untuk melakukan konfigurasi langsung pada tingkat *hypervisor*. Kondisi ini dapat meningkatkan tingkat frustrasi pengguna serta mendorong praktik *workaround* yang berpotensi melemahkan efektivitas keamanan siber. Oleh karena itu, diperlukan sebuah mekanisme penyediaan *virtual machine* yang tidak hanya mengandalkan otomasi berbasis IAC, tetapi juga menyediakan antarmuka terpusat yang mudah digunakan oleh pengguna non-teknis melalui konsep *dashboard web self-service*.

1.2 Perumusan Masalah

Berdasarkan latar belakang di atas, rumusan masalah dalam penelitian ini adalah:

1. Bagaimana merancang dan mengimplementasikan *dashboard self-service* berbasis *Infrastructure as Code* (IaC) menggunakan Terraform yang dapat membuat *virtual machine* (VM) secara lebih cepat pada lingkungan Proxmox VE?
2. Bagaimana memfasilitasi pengguna dalam memilih dan menyediakan *virtual machine* yang lebih aman (melalui mekaniseme isolasi *role*, otentikasi kunci SSH, dan pengkonfigurasian otomatis via Cloud-init)?
3. Bagaimana efisiensi waktu eksekusi *provisioning* yang dihasilkan oleh sistem otomatisasi berbasis *dashboard* jika dibandingkan dengan proses *provisioning* konvensional (manual) pada Proxmox VE?

1.3 Batasan Masalah

Sejalan dengan rumusan masalah yang telah ditetapkan, tujuan utama dari penelitian ini adalah sebagai berikut:

1. Objek penelitian dibatasi secara eksklusif pada platform Proxmox Virtual Environment (VE), tanpa melakukan perbandingan terhadap *hypervisor* atau platform *cloud* lainnya.
2. Otomatisasi penyediaan infrastruktur (*provisioning*) dibatasi menggunakan teknologi Terraform.
3. Spesifikasi sumber daya VM dibatasi secara statis pada 2 vCPU, 1 GB RAM, dan 10 GB penyimpanan dengan sistem operasi Ubuntu Server 22.04 LTS berbasis cloud-init.
4. Mitigasi keamanan difokuskan pada otomasi injeksi *SSH Public key* dan isolasi hak akses melalui *Role-Based Access Control* (RBAC).
5. Pengelolaan pasca-*provisioning* dibatasi pada kendali siklus hidup mesin virtual (*start, stop, restart, destroy*) tanpa menyediakan akses konsol atau VNC.
6. Skala pemanfaatan infrastruktur dibatasi pada lingkungan nonproduksi dengan alokasi maksimal satu mesin virtual per pengguna.

7. Evaluasi keberhasilan dibatasi pada pengukuran kinerja operasional, validasi keamanan, dan tingkat kebergunaan berdasarkan instrumen *System Usability Scale* (SUS) dengan melibatkan 30 responden.
8. Pengembangan *backend* dan *Application Programming Interface* (API) dibatasi secara eksklusif menggunakan bahasa pemrograman Python dan kerangka kerja Flask.

1.4 Tujuan dan Manfaat

1.4.1 Tujuan

Sejalan dengan rumusan masalah yang telah ditetapkan, tujuan utama dari penelitian ini adalah sebagai berikut:

1) Mengotomatisasi proses penyediaan VM (*provisioning*) di lingkungan Proxmox VE menggunakan pendekatan *Infrastructure as Code* (IAC) dengan menggunakan Terraform guna memangkas inefisiensi waktu operasional, menjaga konsistensi konfigurasi, dan meminimalkan tingkat kesalahan manusia (*human error*).
2) Merancang dan membangun *dashboard self-service* sebagai lapisan abstraksi visual guna memfasilitasi pengguna nonteknis dalam melakukan penyediaan infrastruktur secara mandiri dan intuitif, tanpa memerlukan interaksi langsung dengan antarmuka baris perintah (CLI).
3) Mengimplementasikan arsitektur keamanan preventif terintegrasi berbasis *Role-Based Access Control* (RBAC) dan otomasi injeksi *SSH Public key* guna memitigasi kerentanan autentikasi kata sandi serta mencegah konflik manajemen hak akses pada infrastruktur virtualisasi privat multipengguna.

1.4.2 Manfaat

Hasil dari penelitian ini memberikan manfaat sebagai berikut:

1. Bagi Pengguna NonTeknis

   Meningkatkan kemandirian operasional pengguna dalam melakukan *provisioning* mesin virtual secara instan dan terstandarisasi. Keberadaan antarmuka visual (*dashboard*) menyederhanakan proses interaksi pengguna

dengan sistem, sehingga pengguna tidak perlu menguasai skrip *Infrastructure as Code* (IaC) maupun antarmuka baris perintah (CLI).

2. Bagi Administrator dan Pengelola Infrastruktur TI

   Menurunkan beban kerja repetitif yang berasal dari *provisioning* manual serta meminimalkan kesalahan manusia (*human error)*. Selain itu, sistem ini memperkuat keamanan infrastruktur privat melalui otomatisasi injeksi *SSH Public key* dan penerapan *Role-Based Access Control* (RBAC) untuk pengelolaan akses.

3. Bagi Institusi

   Mendukung akselerasi transformasi digital institusi melalui penyediaan layanan TI yang adaptif, efisien, dan aman. Efisiensi operasional yang dihasilkan dapat mendukung penyediaan lingkungan komputasi nonproduksi untuk kegiatan pengembangan dan pengujian secara lebih cepat dan responsif tanpa mengorbankan stabilitas sistem utama.

4. Bagi Pengembangan Ilmu Pengetahuan

   Memberikan kontribusi literatur dan kerangka kerja empiris terkait implementasi *Infrastructure as Code* (IaC) pada lingkungan virtualisasi privat (*Private Cloud*). Penelitian ini memberikan bukti bahwa penerapan lapisan abstraksi (*dashboard self-service*) mampu menyederhanakan kompleksitas teknis IaC, sekaligus mengintegrasikan arsitektur keamanan preventif sejak awal melalui *secure by default*.