Faysal Aziz

Terraform Bespoke

Building Enterprise Infrastructure-as-Code with Terraform and Ansible in VMware and Azure - 2 days

FA-0438 · 2 days

DevOps, Cloud & Infrastructure · Automation, RPA & Power Platform

---

Introduction

Modern infrastructure teams are expected to provision, scale, secure, and maintain environments at a speed that traditional administration methods can no longer support. Infrastructure-as-Code has evolved from a niche DevOps capability into a core operational requirement for virtualization, private cloud, and hybrid cloud environments. Organizations running VMware vSphere alongside Azure are increasingly adopting Terraform and Ansible to standardize deployments, reduce configuration drift, and create repeatable infrastructure workflows across multiple platforms.

This intensive two-day hands-on course focuses primarily on Terraform automation within VMware vSphere environments while also introducing Azure provisioning workflows and operational integration with Ansible. Participants will gain practical exposure to enterprise Infrastructure-as-Code methodologies, state management strategies, reusable module design, automation workflows, and hybrid infrastructure deployment patterns aligned with current industry practices. The course is designed around real-world operational use cases rather than purely academic demonstrations, and is delivered by an instructor with over 30 years of industry experience using production-driven content and enterprise deployment methodologies. Recent enterprise trends emphasize combining Terraform for infrastructure provisioning with Ansible for configuration management and orchestration in hybrid cloud environments.

Learning outcomes

By the end of this course, participants will be able to:

- Understand Infrastructure-as-Code principles and enterprise automation workflows
- Deploy and manage VMware infrastructure using Terraform
- Configure Terraform providers, resources, variables, outputs, and modules

- Build reusable Terraform configurations for enterprise virtualization environments
- Manage Terraform state files and remote backend strategies
- Automate Azure infrastructure provisioning using Terraform
- Integrate Terraform workflows with Ansible automation
- Execute post-provisioning configuration management using Ansible playbooks
- Understand hybrid infrastructure deployment patterns across VMware and Azure
- Apply current Infrastructure-as-Code operational best practices
- Troubleshoot common Terraform and Ansible deployment issues

Prerequisites

- Basic understanding of virtualization concepts
- Familiarity with VMware vSphere administration
- Basic Linux command-line knowledge
- Basic networking knowledge
- General understanding of cloud computing concepts
- Basic SSH and remote administration familiarity
- Prior scripting experience is beneficial but not mandatory
- Laptop capable of running virtualization or remote lab access tools
- Administrative access to a Linux environment for lab exercises

Training outline

1. Introduction to Infrastructure-as-Code and Enterprise Automation

Understanding Infrastructure-as-Code

1. Evolution of infrastructure management
2. Traditional provisioning versus Infrastructure-as-Code
3. Declarative versus imperative automation
4. Enterprise automation drivers
5. Infrastructure consistency and repeatability
6. Automation maturity models
7. Hybrid cloud operational requirements

Terraform in Modern Enterprise Environments

1. Terraform architecture overview
2. Terraform workflow lifecycle
3. HashiCorp Configuration Language fundamentals
4. Providers, resources, data sources, and modules
5. Infrastructure state management concepts
6. Enterprise adoption trends
7. Multi-platform automation strategies
8. Terraform and OpenTofu ecosystem overview

Ansible in Infrastructure Automation

1. Configuration management concepts
2. Agentless automation architecture
3. YAML fundamentals
4. Inventory management concepts
5. Playbooks and task execution
6. Idempotent automation principles
7. Terraform and Ansible integration patterns
8. Infrastructure provisioning versus configuration management responsibilities

2. Environment Preparation and Toolchain Setup

Linux Automation Environment Preparation

1. Linux package management
2. Environment configuration
3. CLI tooling requirements
4. Secure administrative access
5. SSH key management
6. Directory structures and workspace organization

Terraform Installation and Configuration

1. Terraform installation procedures
2. Binary verification

3. Environment path configuration
4. Terraform CLI initialization
5. Terraform version management
6. Terraform formatting and validation tools

Visual Studio Code Integration

1. VS Code installation and configuration
2. Terraform extensions and plugins
3. Ansible extensions
4. Remote SSH integration
5. Formatting and linting support
6. Workspace organization strategies

Ansible Installation and Preparation

1. Ansible installation
2. Python dependencies
3. Inventory configuration
4. SSH authentication setup
5. Ansible configuration file structure
6. Validation and connectivity testing

3. Terraform Core Concepts and Workflow Operations

Terraform Configuration Development

1. Terraform block structure
2. Provider configuration
3. Resource definitions
4. Variables and local values
5. Outputs and dependency management
6. Resource interpolation
7. Conditional expressions
8. Functions and expressions

Terraform Workflow Lifecycle

1. terraform init
2. terraform validate
3. terraform fmt
4. terraform plan
5. terraform apply
6. terraform destroy
7. Incremental infrastructure modifications
8. Resource dependency processing

Terraform State Management

1. Understanding terraform.tfstate
2. State dependency relationships
3. State locking concepts
4. Local versus remote state
5. Risks of manual state modification
6. State backup considerations
7. State security concerns
8. Team collaboration workflows

Terraform Modularization

1. Introduction to modules
2. Root modules and child modules
3. Variable passing strategies
4. Output sharing between modules
5. Reusable infrastructure design
6. Environment abstraction concepts
7. Module versioning considerations
8. Enterprise module organization

4. VMware vSphere Automation with Terraform

VMware vSphere Provider Fundamentals

1. VMware provider architecture
2. vCenter integration concepts
3. Provider authentication methods
4. API communication workflows
5. Provider version management
6. VMware inventory concepts

VMware Environment Discovery and Preparation

1. Datacenter discovery
2. Cluster identification
3. Resource pool management
4. Datastore selection
5. Network inventory discovery
6. VM template preparation
7. VMware Tools considerations
8. Permission and role requirements

Building VMware Terraform Configurations

1. Provider block configuration
2. Using VMware data sources
3. Datacenter resource selection
4. Cluster and host targeting
5. Datastore assignment
6. Virtual networking configuration
7. Template selection workflows

Virtual Machine Provisioning with Terraform

1. Deploying virtual machines from templates
2. VM cloning workflows
3. CPU allocation

4. Memory allocation
5. Disk provisioning strategies
6. Thin versus thick provisioning
7. Multi-network interface configuration
8. Guest OS customization
9. Hostname and IP assignment
10. VMware customization specifications

Managing VMware Infrastructure Changes

1. Scaling VM resources
2. Updating VM configurations
3. Storage expansion workflows
4. Network reassignment
5. Controlled infrastructure modifications
6. Resource lifecycle management
7. Safe infrastructure destruction
8. Drift detection concepts

VMware Operational Best Practices

1. Naming standards
2. Tagging strategies
3. Resource organization
4. Template lifecycle management
5. State management strategies
6. Infrastructure governance
7. Environment separation
8. Production operational workflows

5. Terraform Automation on Microsoft Azure

Azure Provider Fundamentals

1. Azure provider architecture
2. Azure authentication methods
3. Service principals and managed identities

4. Subscription management concepts
5. Provider version management

Azure Resource Provisioning

1. Resource groups
2. Virtual networks
3. Subnets and routing
4. Network security groups
5. Virtual machine provisioning
6. Public and private IP configuration
7. Storage account deployment
8. Tagging and governance practices

Hybrid VMware and Azure Concepts

1. Hybrid infrastructure models
2. Workload portability concepts
3. Infrastructure standardization
4. Multi-environment deployment workflows
5. Consistent automation methodologies
6. Shared module strategies
7. Governance across platforms

Azure State and Collaboration Concepts

1. Remote state storage
2. Azure backend configuration
3. Team collaboration workflows
4. Environment separation
5. Infrastructure validation workflows
6. Security and compliance considerations

6. Integrating Terraform and Ansible

Terraform and Ansible Architectural Roles

1. Infrastructure provisioning responsibilities
2. Configuration management responsibilities
3. Separation of concerns
4. Enterprise automation layering
5. Provisioning versus orchestration workflows
6. Hybrid automation patterns

Passing Terraform Outputs to Ansible

1. Terraform output generation
2. Dynamic inventory concepts
3. Inventory automation workflows
4. Infrastructure discovery
5. Host variable management
6. Environment abstraction

Ansible Playbook Development

1. Playbook structure
2. Task execution workflows
3. Variables and templates
4. Roles and reusable automation
5. Service installation automation
6. Linux configuration management
7. Patch and update workflows

Post-Provisioning Automation

1. Operating system hardening
2. User and access management
3. Package deployment
4. Service configuration
5. Middleware installation concepts

6. Environment standardization
7. Compliance automation concepts

Terraform and Ansible Operational Best Practices

1. Maintaining clear automation boundaries
2. Avoiding configuration drift
3. Git-based workflow management
4. CI/CD integration concepts
5. Dynamic inventory strategies
6. Secure credential handling
7. Enterprise automation governance

7. Infrastructure Lifecycle Management and Enterprise Operations

Enterprise Infrastructure Governance

1. Change management workflows
2. Approval and validation processes
3. Infrastructure auditability
4. Policy-as-code concepts
5. Compliance considerations
6. Operational risk reduction

CI/CD and Automation Pipelines

1. Infrastructure pipeline concepts
2. Validation and testing workflows
3. Automated deployment strategies
4. Infrastructure promotion models
5. GitOps concepts
6. Deployment orchestration workflows

Disclaimer

This training outline is intended as a high-level guideline for course delivery and content coverage. The instructor reserves the right to amend, reorganize, expand, condense, or substitute topics, labs, technologies, and delivery sequences as deemed appropriate based on participant skill levels, operational requirements, platform availability, technological developments, and time constraints without prior notice.

© 2026 Faysal Aziz · Kuala Lumpur