



Standardizing Hybrid Cloud Environments with HashiCorp Terraform and Cisco Intersight



WHITEPAPER

Contents

Introduction.	.3
Different Tools Create Silos in Private and Public Infrastructure	.3
Hybrid Environments Often Lack a Central Control Plane for Management	5
HashiCorp Terraform Cloud and Cisco Intersight: A Solution for Hybrid Cloud Automation	6
How It Works	.7
The Advantages of a Cisco Intersight and Terraform Cloud Solution	11
Shifting Gears: Bringing Infrastructure as Code to Private Datacenters	12

Introduction

An estimated 87% of enterprises have a hybrid cloud strategy combining public cloud providers and infrastructure from their own private datacenters. This provides the best of both worlds, leveraging the agility and potential cost savings of public cloud, while continuing to host applications that have specific security and data retention requirements or are otherwise better suited to private datacenters.



Public clouds have introduced a very different paradigm for managing infrastructure. Not only is cloud infrastructure more dynamic and scalable — it can also be configured and provisioned as code. Codifying virtual servers, networking, databases, etc., enables them to be provisioned through automation as part of a CI/CD pipeline. This white paper outlines the implications of this tectonic shift, as well as how this new cloud operating model can be applied to both public clouds and private infrastructure.

Different Tools Create Silos in Private and Public Infrastructure

Many enterprises maintain private datacenters in order to ensure absolute compliance and security for certain applications. Organizations generally manage their private datacenters with vendorspecific utilities or external tools specially designed to manage individual infrastructure components. Often management is specific to types of infrastructure, such as network switches, server hardware, storage arrays, and so on. But this can lead to inconsistent tool standardization amongst various teams. Automation workflows are often imperative-based, which aren't very flexible and impose a heavy lift for building logic and other validation.

The DevOps movement has embraced public clouds and cloud services that offer robust APIs. By consuming cloud infrastructure as a service, DevOps teams can define infrastructure using code, and then provision it using automated CI/CD workflows. Both Amazon Web Services and Microsoft Azure have dedicated services to enable practitioners to define and provision infrastructure as code (IaC). But teams using AWS CloudFormation or Azure ARM templates must invest significant time and energy to get a production environment up and running. This amounts to sunk costs just to get started. Any workflows built around them further contribute to vendor lock-in. Since the average enterprise leverages five different public clouds, using cloud-specific domain-specific languages (DSLs) can create even more silos and skills gaps, factors that technical leaders often cite as issues within their organizations.



Different kinds of infrastructure require different kinds of tooling and processes, and create silos across an organization.

Hybrid Environments Often Lack a Central Control Plane for Management

While third-party tools may provide some visibility into both public and private infrastructure, they don't do anything to break apart the silos that often prevent this visibility from becoming actionable. Examples of issues that arise with hybrid infrastructure include:

- 1 There is no means to set up governance and security across environments in a meaningful way. To effectively set up governance, a management layer must be acutely aware of the sensitive information available within each cloud service or private datacenter, such as outputs, credentials, and permissions.
- 2 Managing costs across a growing number of public clouds becomes a challenge. While there are tools for viewing costs across clouds, these tools don't provide a common paradigm for managing these costs, especially across shadow IT implementations that are often provisioned by individual teams.
- ³ There is no common language or workflow for automation. Private infrastructure can be made consumable via a service. But if each switch, server, network, or public cloud has its own model for management and automation, teams will be hard pressed to stay on top of them all. The specificity of required skills for each tool requires significant training, often negating the productivity gains that led teams to choose these tools in the first place.
- 4 **Cloud services are dynamic in nature, while management tools for them are not.** Cloud platforms like AWS and Azure add and take away services and API endpoints all the time. Any centralized control plane must be aware of these changes at any given point to accurately manage them.
- ⁵ There is no consistent method for disaster recovery. Preparing for an outage on public clouds requires practitioners to set up high availability and failover processes to different zones. Private infrastructure requires much more granular oversight of each component, such as servers, switches, networks, etc. Setting up a disaster recovery plan across hybrid clouds requires two different skill sets and tooling.

HashiCorp Terraform Cloud and Cisco Intersight: A Solution for Hybrid Cloud Automation

Cisco Intersight's Terraform provider with HashiCorp Terraform Cloud provides a centralized control plane for management and automation across public and private environments. Terraform provides a single workflow for provisioning resources across all major cloud providers and more than 1,000 different cloud services. Intersight ties in on-premises datacenter management, providing management for servers, networks, and even individual switches and firmware updates. With the Cisco Intersight provider for Terraform, organizations can use the same workflow they use to provision AWS and Azure resources to bring entire datacenters online.



With Cisco Intersight's Terraform provider and HashiCorp Terraform Cloud, teams can use their preferred infrastructure. The process for provisioning remains the same, even if it is on premises.

How It Works

The Terraform workflow manages over 1,000 different public clouds and services

HashiCorp Terraform is a cloud automation tool that provides a single workflow for provisioning all major clouds and cloud providers. Terraform relies on plugins called providers to interact with public clouds, SaaS providers, and other APIs. More than 1,000 infrastructure providers are available. Terraform infrastructure as code uses a modular domain-specific language called HashiCorp Configuration Language (HCL) that makes it human-readable, machine-executable, and easy to debug. Entire collections of infrastructure can be templatized using Terraform modules. The Terraform workflow also provides safety guardrails at each phase of infrastructure provisioning. When operators execute their Terraform configuration, they see a snapshot of their planned output for the resources they are provisioning. Then the operator executes the apply phase, when resources are actually provisioned. All resources provisioned are recorded in a state file that can be tracked, audited, and versioned, making it easy for operations teams to know the "who, what, when, and where" of any given provisioning run.





Cisco Intersight is the control plane for a private datacenter, and can be managed using a Terraform provider

HashiCorp maintains an ecosystem of 1000+ providers, though you may find a provider doesn't exist, or doesn't support a particular workflow for specific hardware and middleware. This is where Intersight comes in. Intersight Assist is a secure, turn-key appliance that helps facilitate the management, observability, and orchestration of on-prem datacenter resources directly from Cisco Intersight.



With Cisco Intersight's Terraform provider and HashiCorp Terraform Cloud, operators can use the same workflow for on-premises and cloud infrastructure.

Terraform Cloud provides centralized management for all provisioning, public and private

By default, Terraform open source runs on a practitioner's local machine, which doesn't scale across an organization. Teams who choose Terraform for infrastructure automation collaborate using Terraform Cloud. By putting Terraform operations in the cloud, administrators provide a centralized run environment that includes all of the security and governance rules for all of an organization's public and private infrastructure. And version control integration lets operators collaborate better on infrastructure as code.

Terraform Cloud can also execute all infrastructure changes behind an organization's firewall using a feature called cloud agents. Cisco Intersight Assist manages cloud agents, which reduces the manual work required to use them.



The Advantages of a Cisco Intersight and Terraform Cloud Solution

What are the advantages of leveraging Terraform Cloud across a hybrid cloud environment?

- Bringing the DevOps automation mindset to the private datacenter. By codifying private infrastructure through Intersight and Terraform, updates can be versioned and automated. Operators can collaborate and minimize mistakes by using version control. Teams spend less time on error-prone manual tasks, and more time supporting the business.
- Providing a single consistent workflow for operators. DevOps teams can manage datacenters, remote sites, branch offices, and edge environments with the same workflow that they use to provision resources on AWS, Azure, or other cloud services. This eliminates common silos that often occur in hybrid environments.
- Terraform Cloud offers a single control plane for infrastructure governance and security.
 Terraform Cloud gives administrators granular access to credentials, outputs, and access controls across all of an organization's public and private infrastructure. Administrators can also set up entire policy sets as code using Sentinel, which is a framework for defining policy as code.
 These policy sets define the conditions required for infrastructure changes, and can be enacted for specific workspaces or across an organization.
- Enabling teams to include IaC as part of their disaster recovery operations. By centralizing infrastructure provisioning across Terraform Cloud and Cisco Intersight, IT teams can use one infrastructure as code workflow to set up disaster recovery for any and all infrastructure, enhancing existing recovery point and recovery time objectives (RTO/RPO).

Shifting Gears: Bringing Infrastructure as Code to Private Datacenters

In the last few years, directors of infrastructure and CIOs at over 500 enterprises have opted to provision their infrastructure as code through HashiCorp Terraform commercial products. Some have trained entire teams of systems administrators to write infrastructure as code in order to increase productivity, reduce silos, and standardize processes and tools. Terraform has also been downloaded more than 100 million times, making it one of the most popular DevOps tools on the market.

With Terraform Cloud and Cisco Intersight, infrastructure teams don't have to sacrifice productivity gains for the sake of compliance. To learn more, talk to your representatives at Cisco and HashiCorp.

More resources:

- Blog: Cisco Intersight Service for HashiCorp Terraform At-a-Glance
- Video: Demo of Cisco Intersight Service for Terraform
- Webinar: Cisco Intersight and Terraform Cloud
- Sandbox: Cisco Intersight Terraform DevNet





