

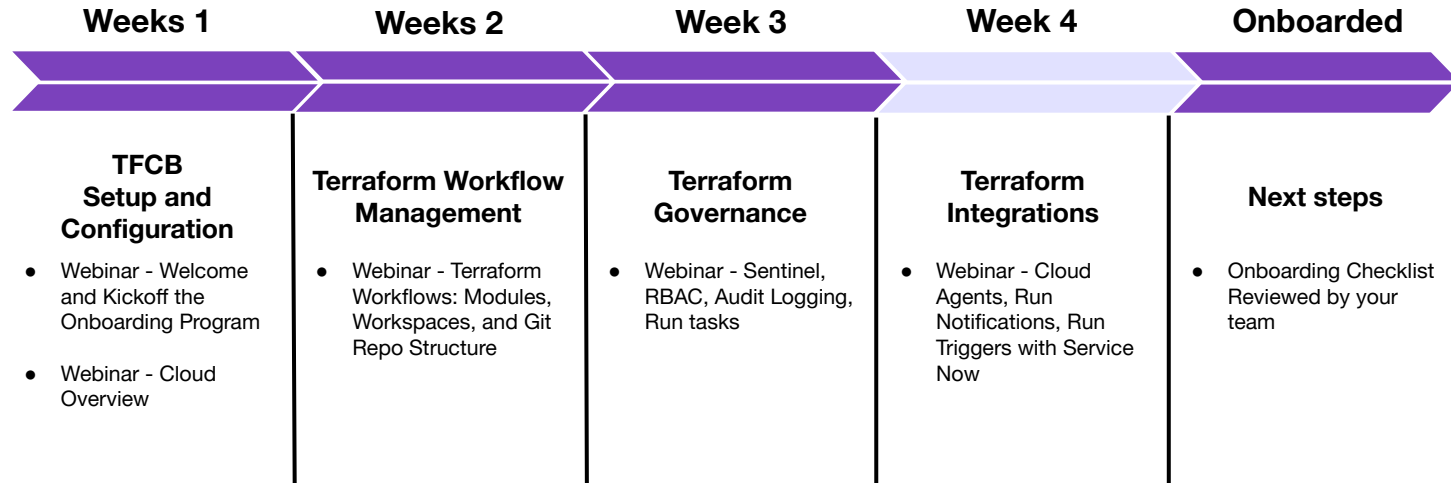
Terraform Integrations & Program Closing



Agenda

1. Cloud Agents
2. Kubernetes Integration with TFC
3. ServiceNow Integration
4. Splunk Integration
5. Run Triggers
6. Run Notifications
7. Production Readiness
8. Closing Resources

TFCB Path to Production



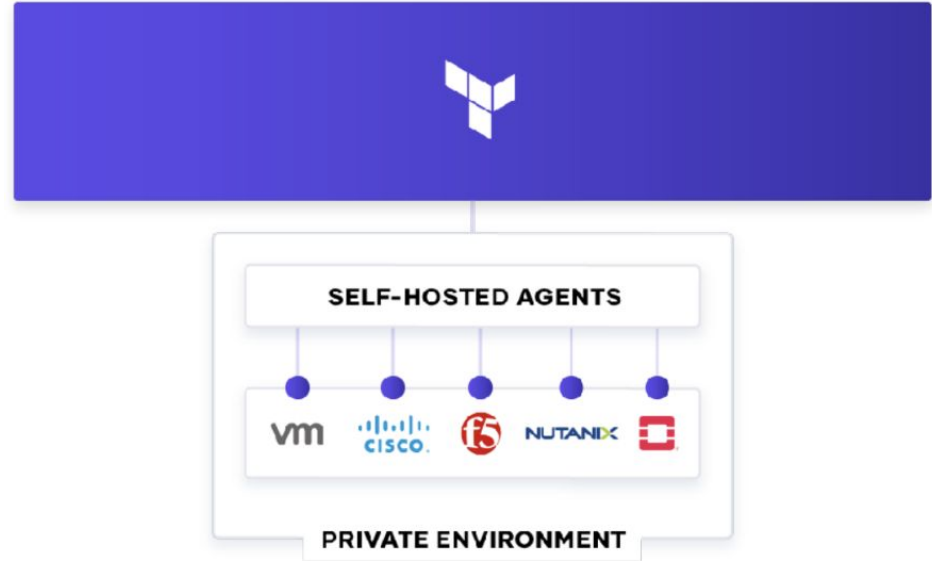
01

Cloud Agents

Terraform Cloud Agents



- Communicate with isolated, private, on-premises infrastructure, such as vSphere, Nutanix, & OpenStack, or across multiple cloud accounts
- The Cloud Agent is an x86-based Golang binary
- Deployable on bare metal, in a VM, as a Docker container, or in a Kubernetes cluster



Architecture

- No inbound public internet connectivity is required
- Supports cloud provider IAM systems for security credentials on demand
- Supports multi account, multi-environment strategy

The screenshot displays the Terraform Cloud interface for a workspace named 'aws-demo-app'. The top navigation bar includes links for 'hashicorp', 'Workspaces', 'Modules', 'Usage', 'Settings', and 'HCP'. The breadcrumb trail shows the path: 'hashicorp / Workspaces / aws-demo-app / Runs / run-6x7HQ6XeVmeNDYmy'. The main content area shows a run triggered manually in Terraform Cloud. Key details include: Run ID 'run-6x7HQ6XeVmeNDYmy', Configuration from GitHub, Branch 'master', Repo 'aws-demo-app', Commit '67ba093: Update main.tf', and Trigger 'Run manually triggered'. The 'Execution Mode' is set to 'Agent'. The run status is 'Plan finished' and 'Apply finished', both completed a minute ago. The 'Agent Pool' is 'my-first-pool' and the 'Agent' is 'agent_01'. The 'Cost estimation' shows 0 resources estimated at \$0.00/mo. The 'Apply' status shows 1 resource added, 0 changed, and 0 destroyed. The 'View raw log' button is visible. The log content shows Terraform v0.13.5 initializing plugins and modules, creating the 'random_id.random' resource, and applying the configuration. The output shows the 'random' ID as 'a15a3d81b42c5d3f'.





Requirements

Supported Platforms

- Baremetal
- Docker
- Kubernetes (K8S)
- VMware VM
- AWS EC2 VM, EKS, ECS, Fargate EKS, Fargate ECS
- Azure VM, Container Service, AKS
- GCP Compute Engine VM, GKE

Hardware Requirements

- x86-based Linux host
- 2 GB of RAM
- 4 GB of disk space

Networking Requirements

- Public Egress, outbound network connections to `app.terraform.io` over HTTPS (443)
- [Terraform Cloud IP Ranges](#)

Agents

An agent pool represents a group of agents that can be used to allow Terraform Cloud to communicate with isolated, private, or on-premises infrastructure. Each agent pool has its own set of tokens which are not shared across pools. When a workspace is configured to execute runs using agents, any available agent in that workspace's associated agent pool is eligible to complete the run.

[Read more in our documentation.](#) 

Create your first agent pool

Agents and agent tokens are organized into agent pools, and cannot be shared among multiple agent pools. Once an agent pool is created, you can generate an agent token to allow your agents to securely communicate with Terraform Cloud.

[Create agent pool](#)

[Learn more about Terraform Agents](#) 

Create an agent pool

1 Name agent pool

2 Token management

An agent pool represents a group of agents that can be used to allow Terraform Cloud to communicate with isolated, private, or on-premises infrastructure. When a workspace is configured to execute runs using agents, any available agent in that workspace's associated agent pool is eligible to complete the run. Learn more about [agents and agent pools](#) [↗](#)

Agent pool names must be unique, and will be used by workspace administrators when linking workspaces to a specific agent pool.

Agent Pool Name

Dashes, underscores, and alphanumeric characters are permitted.

Cancel

Continue



Create an agent pool



Name agent pool



Token management

Token management

Each agent pool has its own set of tokens which are not shared across pools. These tokens allow agents to communicate securely with Terraform Cloud.

Configure your initial tokens for **test** below. Tokens can be created and revoked tokens later, as well.

Tokens

Token description	Created	Last used
No tokens to display		

Add a new token

Choose a description to help you identify this token later.

Description

Create token

Cancel

Finish



Token created

Your new agent token, **test**, is displayed below.

U2VABqmFKk7U0w.atlasv1.4KqCoYqe5AqpDvF0TsDVPfwa0WS3x4ECsvUCKB6oyFy6KgZLW4ZD5txSae3E0mk1S3o [🔗](#)



Warning

This token **will not be displayed again**, so make sure to save it to a safe place.

Set up your agents

Connect to your Docker host and set the following environment variables. `TFC_AGENT_NAME` is optional.

```
$ export TFC_AGENT_TOKEN=U2VABqmFKk7U0w.atlasv1.4KqCoYqe5AqpDvF0TsDVPfwa0WS3x4ECsvUCKB6oyFy6KgZLW4ZD5txSae3E0mk1S3o  
$ export TFC_AGENT_NAME=<my_agent_name>
```

[🔗](#)

Once the environment is configured, run the Docker container with the following command **or** [download the agent file](#). [🔗](#)

```
$ docker run -e TFC_AGENT_TOKEN -e TFC_AGENT_NAME hashicorp/tfc-agent:latest
```

[🔗](#)

[Read more in our documentation](#). [🔗](#)

Cancel

Finish



Resources: Cloud Agent

- [Releases](#)
- [Docker container](#)
- [Terraform module](#) for Kubernetes deployment
- HashiCorp Blog - [Introduction to Terraform Cloud Agents](#)
- [Manage Private Environments with Terraform Cloud Agents](#)

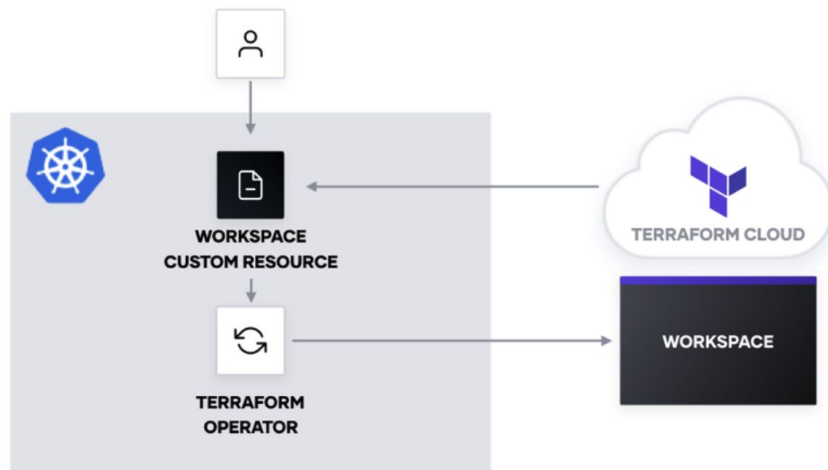
02

Kubernetes Integration

Terraform integration with K8s



Terraform Cloud customers can integrate with Kubernetes to provision infrastructure directly from the K8s control plane. By using the **Terraform Cloud Operator** for Kubernetes to provision infrastructure, you can dynamically create TFC workspaces and variables directly from the K8s control plane.



Use-case



1. Manage the lifecycle of cloud and on-prem infrastructure through a single kubernetes custom resource
2. Provision and manage infra from any cloud provider and other Terraform providers to use them with your existing applications
3. Deploy and manage the Kubernetes resources in a single git repo, or directly from a module in the TF registry.

Steps to Install and Configure



1. Install terraform-k8s via Helm Chart
2. Create the namespace where you will deploy the operator, secrets and workspace resources.
3. Authenticate to TFC via Team API Token and insert it as a TF credentials
4. Create K8s secret named terraformmrc in the namespace. Reference the creds file in the namespace.
5. It is best practice to create a separate team for the operator with “Manage Workspaces” access.

Resources: Kubernetes



- [Deployment Tutorial](#)
- HashiCorp Blog - [Creating Workspaces with the Operator](#)
- [Terraform Cloud Operator for Kubernetes](#)
- [Terraform Cloud Operator Helm Chart](#)
- [Syncing Kubernetes & Terraform Cloud Workspaces](#)

03

ServiceNow Integration

Terraform Integration with ServiceNow



- The Terraform ServiceNow Service Catalog integration enables end-users to provision self-serve infrastructure via ServiceNow
- Connecting ServiceNow to Terraform Cloud lets users:
 - order Service Items
 - create workspaces
 - perform Terraform runs using prepared Terraform configurations hosted in VCS repositories
- [Terraform ServiceNow Service Catalog Integration Setup Instructions](#)

Workflow



Terraform Admin	
Prepare an organization for use with the ServiceNow Catalog	ServiceNow Admin
Create a team that can manage workspaces in that organization	
Create a Team API so the integration can use that team's permission	
Retrieve the oAuth token ID's and repository identifiers for TFC to identify your VCS	
	Install the Terraform Integration application from the ServiceNow App Store
	Connect the integration application with TFC
	Add the Terraform Service Catalog to ServiceNow
	Configure the VCS repositories in ServiceNow
	Configure the Variable Sets for use with the VCS

— 04

Splunk Integration

Splunk for Terraform

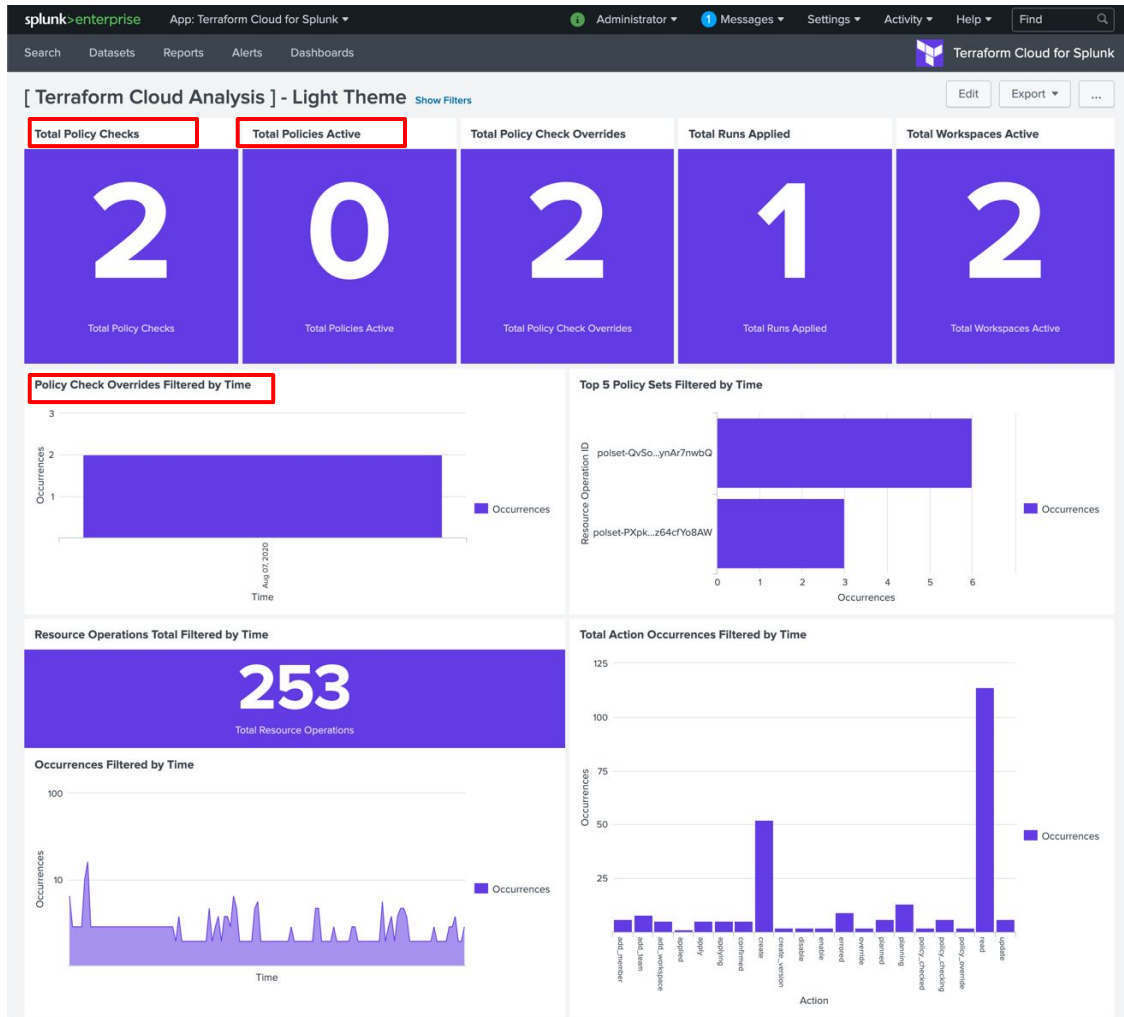


HashiCorp has an official Splunk Dashboard for ingestion of TFC Audit Logs

- Terraform Cloud retains 14 days of audit log information
- Connectivity to Splunk requires HTTPS outbound connectivity to the TFC API

Network Requirements:

Hostname	Port/Protocol	Directionality	Purpose
app.terraform.io	tcp/443, HTTPS	Outbound	Polling for new audit log events via the TFC API





splunk>enterprise

App: Terraform Cloud for Splunk

Messages

Settings

Activity

Help

Find

SearchDatasetsReportsAlertsDashboards

Terraform Cloud for Splunk

New Search

Save As

Close

source="terraform_cloud" sourcetype="terraform_cloud" resource.action="override" | table auth.description, resource.id, resource.type, resource.action, auth.type, timestamp

Last 24 hours

6 events (9/3/20 8:00:00.000 PM to 9/4/20 8:16:43.000 PM)

No Event Sampling

Job

Smart Mode

Events

Patterns

Statistics (6)

Visualization

20 Per Page

Format

Preview

auth.description	resource.id	resource.type	resource.action	auth.type	timestamp
kruddy	polchk-MQwsR84Qo61DXWFH	policy_check	override	Client	2020-09-04T14:39:13.000Z none
kruddy	polchk-MQwsR84Qo61DXWFH	policy_check	override	Client	2020-09-04T14:39:13.000Z none
kruddy	polchk-d27EuPcArUtd67Us	policy_check	override	Client	2020-09-04T14:40:52.000Z none



Resources: Splunk

- Splunkbase: [Terraform Cloud for Splunk](#)
- HashiCorp Blog: [Audit Logging with Splunk](#)
- [Terraform Installation Documentation](#)
- [Splunk Installation Documentation](#)

05

Run Triggers

Run Triggers

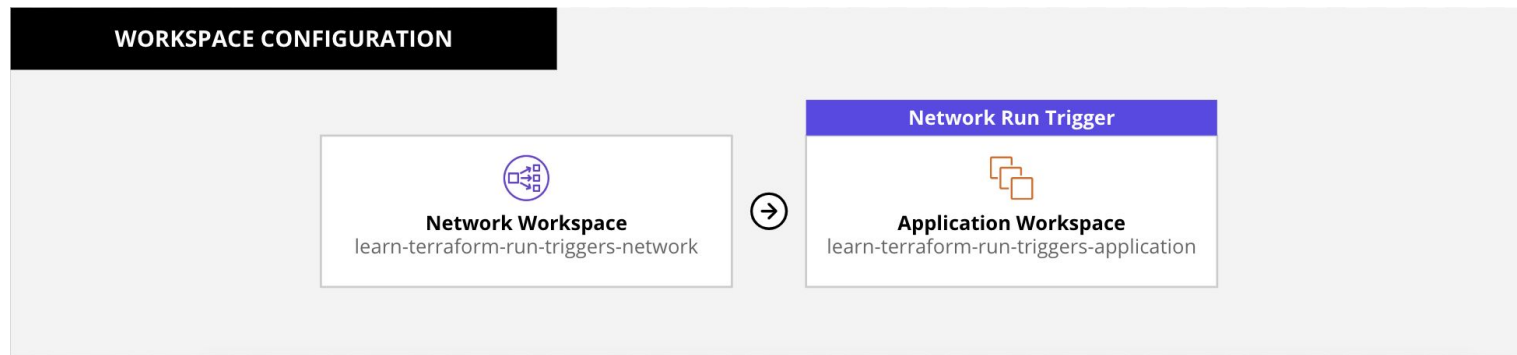


- Create infrastructure pipelines in TFCB
- Allow teams to manage complex infrastructure in TFCB by creating infrastructure pipelines between multiple workspaces
- When a source workspace is selected, multiple dependent workspaces can be linked
- When a successful apply is executed in the source workspace, the dependent workspaces have runs triggered and can be configured to auto-apply their configurations

Use-case - Application Configuration Management



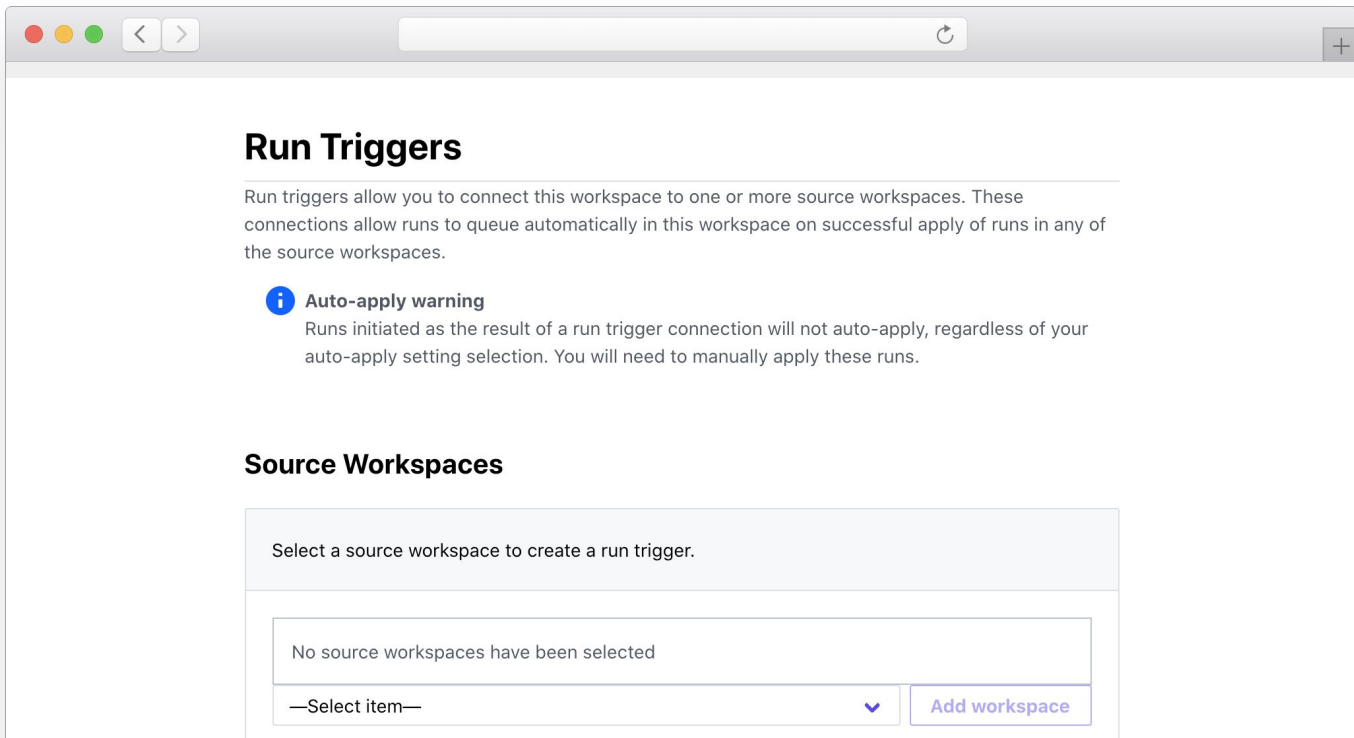
Run triggers automatically trigger updates to application configuration to rebalance servers across new subnets once they are successfully provisioned in the network workspace



Create Run Triggers



Workspace Settings → Run Triggers → Select Source Workspace





Resources: Run Triggers

- [Tutorial: Connect Workspaces with Run Triggers](#)
- [Run Triggers Documentation](#)
- [Terraform Registry - tfe run trigger](#)

06

Run Notifications

Run Notifications



- Run Notifications send updates/notifications to external services with details on run progress
- Notifications can be sent to up to 20 destinations
- Each workspace can be configured with it's own notification settings
- Can send either POST message to any URL via webhook, email message, or sent to Slack & post updates in channels

Notification Triggers



	Trigger	Description
Created	“run:created”	When a run is created and enters the "Pending" state.
Planning	“run:planning”	When a run acquires the lock and starts to execute.
Needs Attention	“run:needs_attention”	Human decision required. When a plan has changes and is not auto-applied, or requires a policy override.
Applying	“run:applying”	When a run begins the apply stage, after a plan is confirmed or auto-applied.
Completed	“run:completed”	When the run has completed on a happy path and can't go any further.
Errored	“run:errored”	When the run has terminated early due to error or cancellation.



Sample Notification Payload

```
CODE EDITOR

{
  "payload_version": 1,
  "notification_configuration_id": "nc-AeUQ2zfKZzW9TiGZ",
  "run_url":
  "https://app.terraform.io/app/acme-org/my-workspace/runs/run-FwnENkvDnrpyFC7M",
  "run_id": "run-FwnENkvDnrpyFC7M",
  "run_message": "Add five new queue workers",
  "run_created_at": "2019-01-25T18:34:00.000Z",
  "run_created_by": "sample-user",
  "workspace_id": "ws-XdeUVMWShTesDMME",
  "workspace_name": "my-workspace",
  "organization_name": "acme-org",
  "notifications": [
    {
      "message": "Run Canceled",
      "trigger": "run:errored",
      "run_status": "canceled",
      "run_updated_at": "2019-01-25T18:37:04.000Z",
      "run_updated_by": "sample-user"
    }
  ]
}
```



Create Notification Trigger

Workspace → Settings → Notifications

The screenshot shows a web application interface for creating a notification trigger. The top navigation bar is dark blue with a logo on the left and a user profile on the right. The main navigation menu includes 'email-notifications', 'Workspaces' (highlighted with a red box), 'Modules', and 'Settings'. Below the navigation bar, the breadcrumb trail reads 'email-notifications / Workspaces / demo_workspace / Settings / Notifications / New'. The main content area is titled 'demo_workspace' and has tabs for 'Runs', 'States', 'Variables', and 'Settings' (highlighted with a red box). A 'Queue plan' dropdown is also visible. The 'Create a Notification' section includes a description: 'Notifications allow you to send messages to other applications based on Run events.' Under the 'Destination' heading, there are three options: 'Webhook' (selected with a blue dot), 'Email', and 'Slack'. Each option has a description and a radio button. Below the destination selection, there are three input fields: 'Name' (with a placeholder 'e.g. My Notification'), 'Webhook URL' (with a placeholder 'https://example.com/...'), and 'Token' (with a placeholder 'Encrypted - write only'). A link to the documentation is provided at the bottom.

email-notifications / Workspaces / demo_workspace / Settings / Notifications / New


demo_workspace


Runs States Variables Settings Queue plan


Create a Notification

Notifications allow you to send messages to other applications based on Run events.

Destination


Webhook
POST messages to any URL
☒


Email
Send messages to users via Email
☐


Slack
Send messages to a Slack Channel
☐

Name

e.g. My Notification

Webhook URL

https://example.com/...

Token

Encrypted - write only

Used to generate the HMAC on the notification request. [Read more in the documentation](#).



Resources: Run Notifications

- [Documentation: Run Notifications](#)
- [Notification Configurations API](#)
- [Terraform Registry - tfe_notification_configuration](#)

07

Terraform Cloud Production Readiness



Production Readiness



Join security & vulnerability announcements list

<https://discuss.hashicorp.com/c/security/52>






Discuss

Sign in



Now open: CFP & Registration for HashiTalks 2022. Sign up today to join us for 24-hours of knowledge-sharing: hashi.co/hashitalks-2022-discuss

Security ▾ Security-Vault ▾ | Latest Top

Topic		Replies	Views	Activity
HCSEC-2021-34 - Vault, Consul, Boundary, and Waypoint Affected By Denial of Service in Golang's net/http (CVE-2021-44716)		0	487	Dec '21
Security security-vault, security-consul, security-waypoint, security-boundary				
HCSEC-2021-33 - Vault's KV Secrets Engine With Integrated Storage Exposed to Authenticated Denial of Service		0	487	Dec '21
Security security-vault				
HCSEC-2021-30 - Vault's Templated ACL Policies Matched First-Created Alias Per Entity and Auth Backend		1	983	19d
Security security-vault				



Production Readiness



Bookmark the Terraform Cloud Agent Changelog

<https://www.terraform.io/cloud-docs/agents/changelog>



Overview

Use Cases ▾

Editions ▾

Registry

Tutorials

Docs ▾

Community



Terraform Cloud

Download

Filter...

Search Terraform documentation



Terraform Cloud Agents

Overview

Telemetry

Monitoring

Hooks

• Changelog

Back to Cloud and Enterprise

Terraform Cloud Agent Changelog

JUMP TO SECTION ▾

These are the release notes from the Terraform Cloud Agent application. Changes within each release are categorized into one or more of the following labels:

- **FEATURES** - Used for net-new features being added to the agent.
- **BUG FIXES** - Backward-compatible fixes for buggy functionality.
- **IMPROVEMENTS** - Functional improvements to performance, efficiency, etc.
- **SECURITY FIXES** - Fixes for security-related issues.
- **BREAKING CHANGES** - Reserved for changes which break previous functionality.

Each version below corresponds to a release artifact available for download on the official [releases website](#).

1.2.1 (05/10/2022)

BUG FIXES:



— Production Readiness

- Determine key workflows for teams
 - API, CLI, or VCS driven
- SSO and/or MFA configured
- VCS repo standard current and future
- Minimum Terraform code version established
- Internal consumption and training plan created

08

Post-Program Support

TFCB Onboarding Journey Content

- AMER Terraform Cloud June '22 Program Kickoff - password: u7d2gZjq39
- Terraform Cloud Onboarding Workflows June 22' - password: a7d2gZjq44
- Terraform Cloud Onboarding Program Governance June '22 - password: P8ajf3uQ
- AMER TFCB Integrations and Program Closing June 22' - password: C7d2gZjq39

[All program materials](#)

TFCB Onboarding Journey Content

[Custom Learn Collections](#)

- Week 1 - Learn the Terraform Programming Language
- Week 2 - TFCB basics and an introduction to Infrastructure as Code
- Week 3 - Explore the basics of TFCB and learn how to collaborate on infrastructure with Terraform Cloud
- Week 4 - Use Cases
- Week 5 - Modules, Workspaces, Runs, Git Repo Structure
- Week 6 - Terraform Workspaces, VCS, and Terraform State
- Week 7 - Cloud Agents, Single-sign On (SSO), Role-Based Access Control (RBAC), Sentinel, and Audit Logging

Recommended additional resources



We strongly urge you to subscribe to the Terraform Cloud status web page, this can be done here

<https://status.hashicorp.com/>

Support will continue to be your resource for resolving technical challenges

support.hashicorp.com



Additional Training Resources

Wanting to Learn More? A Topic Not Covered? See these



Developer.hashicorp.com/terraform/tutorials

Tutorials on just about everything



HashiCorp Events- <https://www.hashicorp.com/events?type=all>

Great place for find HashiCorp events, conferences, webinars



HashiCorp User Groups (HUGs) - <https://www.meetup.com/pro/hugs/>

With over 50 countries, and 155 user groups, find a HUG located near you



HashiCorp Terraform Certification

Even if you don't want the certification, the [Study Guide](#) is a relevant and useful curriculum to follow especially for team members new to Terraform



HashiCorp Instruct Labs

Want more hands-on experience? Visit our Instruct page

<https://play.instruct.com/hashicorp>

Tutorials

<https://developer.hashicorp.com/terraform/tutorials>



Step-by-step guides to accelerate deployment of Terraform Cloud

The screenshot displays the Terraform Developer website interface. At the top is a dark navigation bar with the Terraform logo, navigation links (Home, Documentation, Tutorials, Install, Registry, Try Cloud), a search bar, and a user profile icon. The left sidebar contains a 'Terraform Home' link and a list of categories: Tutorials, Overview, Get Started, AWS, Azure, Docker, GCP, OCI, Terraform Cloud (highlighted), Fundamentals, CLI, Configuration Language, and Modules. The main content area shows the breadcrumb 'Developer / Terraform / Tutorials / Terraform Cloud' and a document icon. The title 'Get Started - Terraform Cloud' is followed by a paragraph about collaborating on version-controlled configuration. Below this is a 'Start' button and a box indicating '10 tutorials'. The first tutorial card, 'What is Terraform Cloud - Intro and Sign Up', shows a 5min duration and a brief description of Terraform Cloud's features. A second card for 'Log in to Terraform Cloud from the CLI' (3min) is partially visible at the bottom.

Developer / Terraform / Tutorials / Terraform Cloud

Get Started - Terraform Cloud

Collaborate on version-controlled configuration using Terraform Cloud. Follow this track to build, change, and destroy infrastructure using remote runs and state.

[Create an account](#) to track your progress.

[Start](#) 10 tutorials

5min

What is Terraform Cloud - Intro and Sign Up

Sign up for Terraform Cloud, which provides free remote state storage, a stable run environment, version control system (VCS) driven plans and applies, a collaborative web GUI, and more. Create your first organization.

3min

Log in to Terraform Cloud from the CLI

Need Additional Help?



Customer Success

Contact our Customer Success Management team with any questions. We will help coordinate the right resources for you to get your questions answered.

customer.success@hashicorp.com

Discuss

Engage with the HashiCorp Cloud community including HashiCorp Architects and Engineers

discuss.hashicorp.com

Technical Support

Something not working quite right? Engage with HashiCorp Technical Support by opening a ticket for your issue at support.hashicorp.com.

HashiCorp Academy

Terraform [Enterprise Academy](#) classes are virtual and delivered by a live instructor with in-depth Terraform knowledge and implementation expertise.

Academy courses include a sandbox environment for hand-on experience in the 10 labs throughout the 3-day course.



Thank You

customer.success@hashicorp.com
www.hashicorp.com/customer-success