



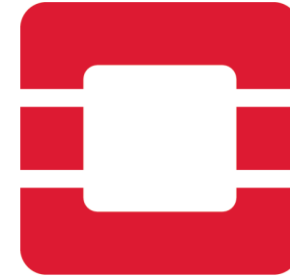
Rancher & Terraform

Christian Frank
December 14th
HashiTalks: DACH

Cover artwork © Pascal Campion



Experience



Experience





Cluster Creation



Options

There are three options to create a Kubernetes cluster with Rancher on Amazon Web Services:

- Managed Kubernetes (EKS)
- Rancher Node-Driver
- Custom nodes

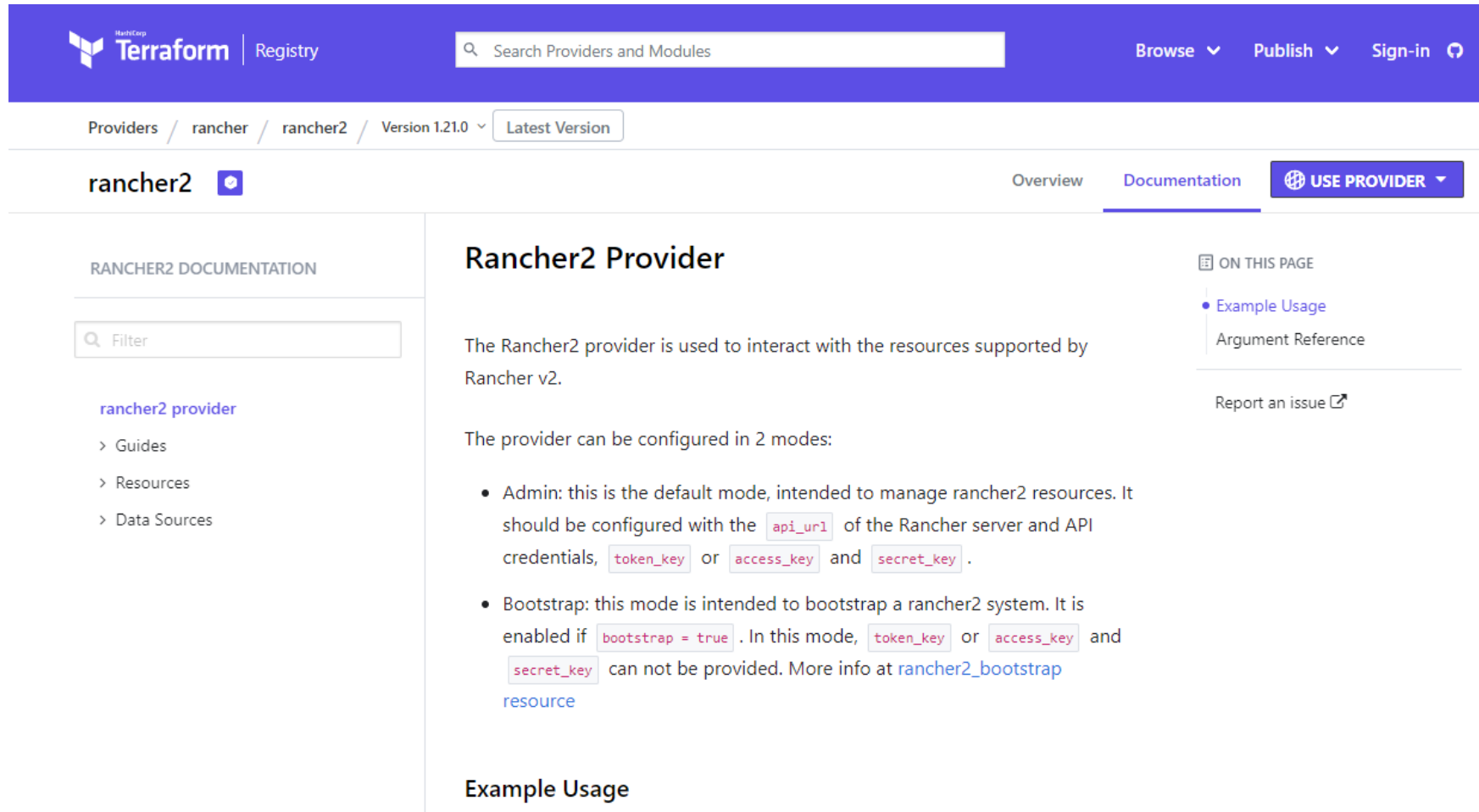


Rancher Node-Driver

- Within a Terraform plan, define the Rancher provider
- Optional: In the Rancher cluster definition, define the AWS cloud provider within the RKE cluster options
- Optional: Create the necessary IAM policies
- Create appropriate cloud credentials
- Create node templates
- Create the cluster (using rancher-machine)
- Note: The creation of RKE2 / k3s Clusters will be using the Cluster API and Fleet (currently in tech preview)



Terraform Provider



The screenshot shows the Terraform Registry interface for the Rancher2 Provider. The top navigation bar is purple with the Terraform logo, a search bar, and links for Browse, Publish, and Sign-in. Below the navigation bar, the breadcrumb trail reads: Providers / rancher / rancher2 / Version 1.21.0 (Latest Version). The main header for the rancher2 provider includes tabs for Overview and Documentation, and a prominent 'USE PROVIDER' button. The left sidebar, titled 'RANCHER2 DOCUMENTATION', contains a search filter and a list of links: rancher2 provider, Guides, Resources, and Data Sources. The main content area is titled 'Rancher2 Provider' and contains the following text: 'The Rancher2 provider is used to interact with the resources supported by Rancher v2.' and 'The provider can be configured in 2 modes:'. A bulleted list describes the 'Admin' mode (default, uses api_url, token_key, access_key, secret_key) and the 'Bootstrap' mode (intended for bootstrapping, uses bootstrap = true, and does not require token_key, access_key, or secret_key). A link to 'rancher2_bootstrap resource' is provided. On the right side of the main content, there is a section 'ON THIS PAGE' with links for 'Example Usage' and 'Argument Reference', and a 'Report an issue' link with an external icon.

Rancher2 Provider

The Rancher2 provider is used to interact with the resources supported by Rancher v2.

The provider can be configured in 2 modes:

- Admin: this is the default mode, intended to manage rancher2 resources. It should be configured with the `api_url` of the Rancher server and API credentials, `token_key` or `access_key` and `secret_key`.
- Bootstrap: this mode is intended to bootstrap a rancher2 system. It is enabled if `bootstrap = true`. In this mode, `token_key` or `access_key` and `secret_key` can not be provided. More info at [rancher2_bootstrap resource](#)

Example Usage



Rancher Provider

```
# Rancher
provider "rancher2" {
  api_url = var.rancher-url
  token_key = var.rancher-token
}
```



EC2 Credentials

```
# Rancher cloud credentials
resource "rancher2_cloud_credential" "credential_ec2" {
  name = "EC2 Credentials"
  amazonec2_credential_config {
    access_key = var.ec2-access-key
    secret_key = var.ec2-secret-key
  }
}
```



EC2 Credentials

Cloud Credentials

Create

Delete

Filter

☐ ID ↕	Name ↕	API Key ↕	Description ↕	Age ↕
Amazon				
☐ cc-tlh6s	EC2 Credentials	AKIAS62FX745QUIO6MXE		59 mins ⋮



Node Templates

```
# Rancher node template
resource "rancher2_node_template" "template_ec2" {
  name = "EC2 Node Template"
  cloud_credential_id = rancher2_cloud_credential.credential_ec2.id
  engine_install_url = var.dockerurl

  amazonec2_config {
    ami = var.image
    region = var.ec2-region
    security_group = [var.ec2-secgroup]
    subnet_id = var.ec2-subnet
    vpc_id = var.ec2-vpc
    zone = var.ec2-zone
    root_size = var.disksize
    instance_type = var.type
  }
}
```



Node Templates

Node Templates

Add Template

☐	State ▾	Name ▾	Owner ▾	Provider ▾	Location ▾	Size ▾	
Owner: Default Admin							
☐	Active	EC2 Node Template	Default Admin	Amazon EC2	eu-central-1a	m3.xlarge	⋮



Node Pool

```
# Rancher node pool
resource "rancher2_node_pool" "nodepool_ec2" {
  cluster_id = rancher2_cluster.cluster_ec2.id
  name = "nodepool"
  hostname_prefix = "rke-${random_id.instance_id.hex}-"
  node_template_id = rancher2_node_template.template_ec2.id
  quantity = var.numnodes
  control_plane = true
  etcd = true
  worker = true
}
```



Node Pool

Instances (1/3) [Info](#)

Filter Instances

Instance state: running X Clear filters

Refresh

Connect

Instance state ▼

Actions ▼

Launch instances ▼

< 1 > ⚙

	Name ▼	Instance ID	Instance state ▼	Instance type ▼	Status check	Alarm status	Availability Zone ▼	Public IPv4 DNS ▼	Pul
☐	rke-ecbb4c-3	I-0c2551623c3f0501a	✓ Running 🔍	m3.xlarge	✓ 2/2 checks passed	No alarms +	eu-central-1a	ec2-54-93-73-173.eu-c...	54.
☐	rke-ecbb4c-2	I-013f629566a1a214e	✓ Running 🔍	m3.xlarge	✓ 2/2 checks passed	No alarms +	eu-central-1a	ec2-3-122-253-42.eu-c...	3.1
☒	rke-ecbb4c-1	I-028c0400be4d781be	✓ Running 🔍	m3.xlarge	✓ 2/2 checks passed	No alarms +	eu-central-1a	ec2-52-59-252-128.eu-...	52.

Instance: i-028c0400be4d781be (rke-ecbb4c-1)

Details

Security

Networking

Storage

Status checks

Monitoring

Tags

▼ Instance summary [Info](#)

Instance ID

I-028c0400be4d781be (rke-ecbb4c-1)

IPv6 address

-

Private IPv4 DNS

ip-172-31-19-238.eu-central-1.compute.internal

Public IPv4 address

52.59.252.128 | [open address](#)

Instance state

✓ Running

Instance type

m3.xlarge

Private IPv4 addresses

172.31.19.238

Public IPv4 DNS

ec2-52-59-252-128.eu-central-1.compute.amazonaws.com | [open address](#)

Elastic IP addresses

-



Kubernetes Cluster

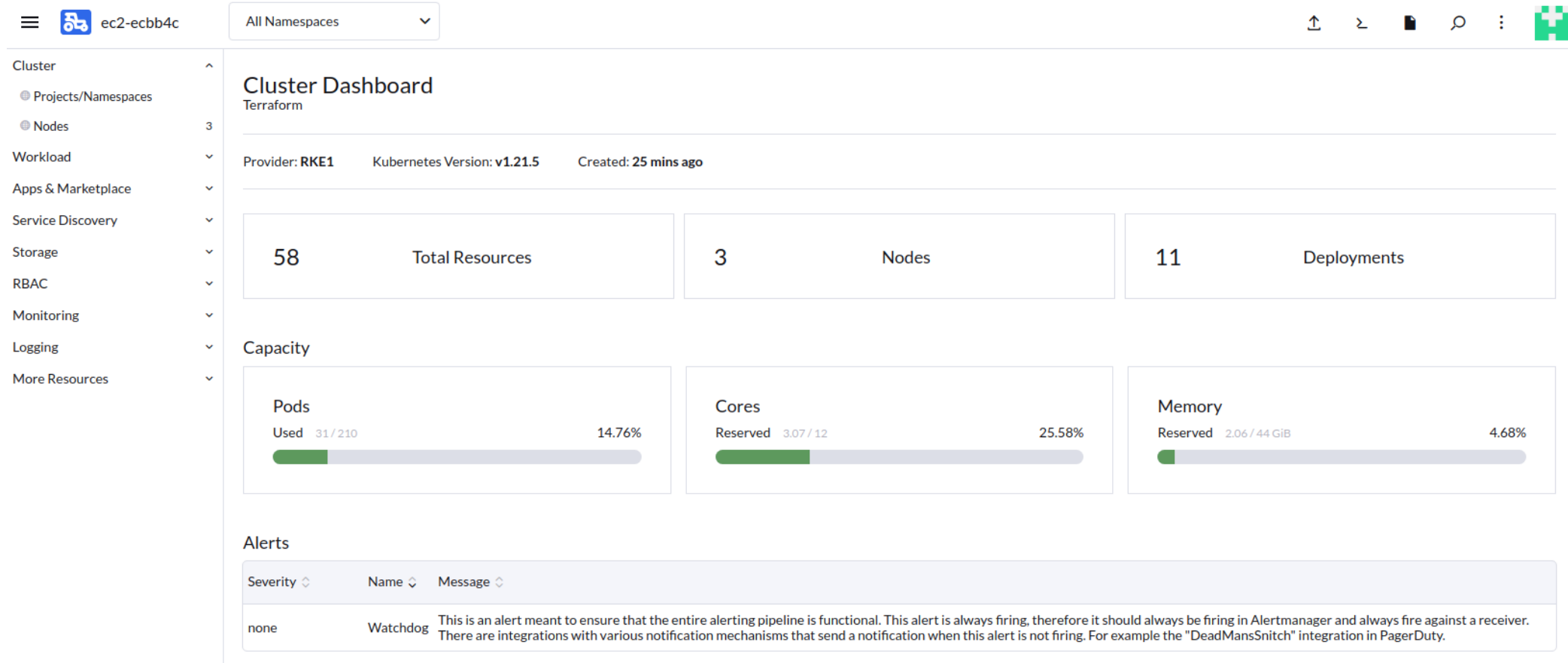
```
# Rancher cluster
resource "rancher2_cluster" "cluster_ec2" {
  name          = "ec2-${random_id.instance_id.hex}"
  description    = "Terraform"

  rke_config {
    kubernetes_version = var.k8version
    ignore_docker_version = false
    network {
      plugin = "flannel"
    }
  }
}
```

...



Kubernetes Cluster



Kubernetes Cluster

 chfrank-cgn

Workspaces

 Registry Settings HashiCorp Cloud Platform [🔗](#)

Upgrade Now

?



chfrank-cgn / Workspaces / compute-test-eu-central / Overview

compute-test-eu-central

ec-cluster-1

Resources 14 Terraform version 0.12.29 Updated 4 minutes ago

Overview States Settings

🔓 Unlocked Actions

▼

Resources 14 Outputs 0

Current as of the most recent state version.

Filter resources

🔍

NAME	PROVIDER	TYPE	MODULE	UPDATED ↓
before	hashicorp/null	null_resourc...	root	Oct 21 2021
cluster_ec2	rancher/rancher...	rancher2_clu...	root	Oct 21 2021
credential_ec2	rancher/rancher...	rancher2_clo...	root	Oct 21 2021
delay	hashicorp/null	null_resourc...	root	Oct 21 2021
instance_id	hashicorp/rando...	random_id	root	Oct 21 2021

⚡ Execution mode: Local

⚙️ Auto apply: Off

Metrics

Metrics will appear once your next run is applied.

Tags (0)

Add a tag

▼

Tags have not been added to this workspace.

Run triggers

📘



Sample Terraform plans

chfrank-cgn / Rancher

Public

Notifications

Star

14

Fork

17

<> CodeIssuesPull requestsActionsProjectsWikiSecurityInsights

master

1 branch

0 tags

Go to file

Code

 chfrank-cgn EKS Cluster	a8633b5 10 minutes ago	44 commits
aks-cluster	Azure update to Rancher 2.5.3	11 months ago
az-cluster-1	Alert Manager	17 days ago
az-cluster-2	Alert Manager	17 days ago
ec2-cluster-1	EKS Cluster	10 minutes ago
eks-cluster	EKS Cluster	10 minutes ago
gcp-cluster	Rancher 2.6.0, Terraform 1.0.7	last month
gcp-nfs-helm2	NFS and Helm	2 years ago
gcp-nfs-helm3	Terraform 1.0	3 months ago
lc-cluster	OpenStack on leaf.cloud	10 months ago
openstack	OpenStack config files	2 years ago
ovirt-cluster	oVirt Cluster	2 years ago
vm-cluster	VMware cluster	8 months ago
.gitattributes	Initialized	2 years ago
CODE_OF_CONDUCT.md	Initialized	2 years ago
LICENSE	Initialized	2 years ago
README.md	OpenStack on leaf.cloud	10 months ago
README.md.backup	AKS Cluster	2 years ago

About

Rancher

Readme

GPL-3.0 License

Releases

No releases published

Packages

No packages published

Languages

HCL 97.0%

Shell 3.0%

<https://github.com/chfrank-cgn/Rancher>





Thank you

cfrank@chfrank.net