Greensill

Rancher V2.0

Setup & Implementation

Guide

Contents

[Changes log 4](#_Toc518026188)

[Rancher Infrastructure 5](#_Toc518026189)

[Rancher V2 High Availability Install 5](#_Toc518026190)

[Windows PC Install 5](#_Toc518026191)

[Downloads 5](#_Toc518026192)

[Enable Scripts 6](#_Toc518026193)

[AWS Keys 6](#_Toc518026194)

[Setting Up the Terraform Environment 6](#_Toc518026195)

[Initialize the Terraform Environment 6](#_Toc518026196)

[Running the Setup 7](#_Toc518026197)

[Logging onto Rancher 7](#_Toc518026198)

[TLS Certificates 8](#_Toc518026199)

[Create a Cluster 9](#_Toc518026200)

[Single Node Cluster 9](#_Toc518026201)

[Multi-node Cluster 9](#_Toc518026202)

[Monitoring 10](#_Toc518026203)

[Prometheus – Stats Gathering 10](#_Toc518026204)

[Grafana- Graphics 10](#_Toc518026205)

[High-Availability Web Guide 11](#_Toc518026206)

[Useful Commands De-Bugging 13](#_Toc518026207)

[Terraform Commands 13](#_Toc518026208)

[Check setup code - Plan 13](#_Toc518026209)

[Run the code - Apply 13](#_Toc518026210)

[Remove the AWS environment - Destroy 13](#_Toc518026211)

[Cluster Setup Commands 13](#_Toc518026212)

[Start the HA cluster 13](#_Toc518026213)

[Remove the cluster 13](#_Toc518026214)

[Kubernetes Cluster Commands 14](#_Toc518026215)

[Show elements and logs 14](#_Toc518026216)

[Show the FQDNS name 14](#_Toc518026217)

[Display errors from the cattle system 14](#_Toc518026218)

[Display network Logs 14](#_Toc518026219)

[Display Ingress 14](#_Toc518026220)

[Display Rancher Logs 14](#_Toc518026221)

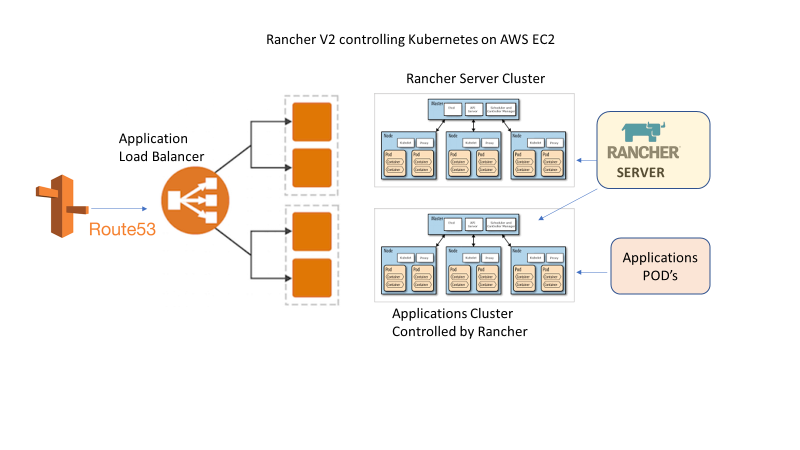
# Changes log

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Who | Description |
| V01.1 | 08/06/2018 | G.Cann | Draft version |
| V01.2 | 25/06/2018 | G.Cann | Added Variable update section |
| V01.3 | 29/06/2018 | G.Cann | Converted to Terraform running the Kubernetes setup |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Rancher Infrastructure

This section details the setup for Rancher V2 on AWS.

The Rancher control server is run as a pod on a separate Kubernetes cluster which provides a high availability environment for the server. The Rancher server controls the applications Kubernetes cluster which deploys the required applications.



# Rancher V2 High Availability Install

Rancher HA is installed in a pod on its own Kubernetes cluster. The cluster must have 3 nodes in separate availability zones in order to provide fail over. The Rancher HA environment is setup using a Kubernetes product called RKE. The RKE setup can be run from a PC via windows or via an EC2 bastion host.

## Windows PC Install

### Downloads

* Download the appropriate RKE application from: https://github.com/rancher/rke
* Use the Windows version and install and update the path information to access the RKE application.
* Install Terraform : https://www.terraform.io/downloads.html
* Install KUBECTL: https://kubernetes.io/docs/tasks/tools/install-kubectl/
* Install the AWS CLI.
* Install the RKE application. https://github.com/rancher/rke/releases/

### Enable Scripts

ensure scripts can run on your PC:

List the execution policy for your machine:

Get-ExecutionPolicy -List

Set the policy for your account:

Set-ExecutionPolicy Unrestricted -Scope CurrentUser

## AWS Keys

Create an AWS key pair to use for the AWS instance setup.

Download the private key into a file called **rancherv2.pem** into the directory with the terraform scripts and other files from the repo.

This will be used by terraform to ssh into the instances.

## Setting Up the Terraform Environment

The terraform scripts will setup the environment that will run the Rancher V2 installation. Before the scripts are run the Terraform variables must be altered as required so that the environment does not clash with a currently running installation.

Use an editor to open the variables.tf file this file contains all of the variable definitions used in the other scripts.

e.g: Check through the names used , the subnet CIDR blocks , VPC id and all the variables used in the setup.

## Initialize the Terraform Environment

Before running the setup script the terraform environment must be initialized on the PC that will be running the scripts. Open a command line window and change the directory to the directory containing the TF scripts.

Enter the following command: ***terraform init***

This will initialize the Terraform files and load the required AWS modules.

## Running the Setup

Clone the repository to the PC running the install.

Find out the external IP address of the PC running the install.

*(from windows command prompt*: **IPCONFIG** ) or

(*from AWS edit the input of a security group and select my\_IP this will give your IP address in AWS*)

In a CMD window change directory to the directory where the files were copied. Then enter the following command:

**Terraform plan**

If this runs without any errors then enter the following command:

**Terraform apply**

Enter the AWS KEY , PC IP Address , secret KEY when prompted.

The setup will create the 3 AWS EC2 instances and the environment required to run Rancher.

The script then tailors the “rancher-cluster.yaml” file with the IP addresses from the Terraform setup and the DNS name used for the configuration. Terraform the executes the Kubernetes RKE application is then run which uses the “rancher-cluster.yaml” file as input and then sets up the Kubernetes cluster on the EC2 instances and installs Rancher V2 in the cluster.

## Logging onto Rancher

From a browser enter the DNS name used in the setup script:

eg https://rancher.greensill.cloud

Set the admin password on the first start up and keep the DNS name the same.

If LDAP has been setup then the Windows logon credentials can be used.

# TLS Certificates

The Rancher 2 environment uses the default \*.greensill.cloud TLS from the ALB (application load balancer) to the browser and a self signed certificate is used within the rancher cluster.

The certificate is contained within AWS Certificate manager and applies to \*.greensill.cloud. The certificate is attached to the ALB used for Rancher.

# Create a Cluster

Do NOT Check the Amazon button under cluster options . The cluster will fail to initialize if this is checked.

Define 3 nodes in the cluster each node in a separate availability zone. Three node templates will have to be created if not already present on for each availability zone.

**Node Template**

Create a Node Template which defines the subnet , instance type , and the AMI to use.

* Choose the region to run this in eg: eu-central-1
* Setup or choose a subnet for the cluster that has access to a public IP. Usually the subnet will be Rancher-1a or 1b or 1c
* Security group – **Rancher-Nodes**
* Choose and instance type to run the cluster. Click the link under the AMI box and select the appropriate AMI number for the region.
* Change the user ID to **rancher**
* Set the IAM role to EC2-S3

Select a name for the cluster.

## Single Node Cluster

All of the components can run on one node for testing . Check all of the boxes (Control,Worker,ETCD)

Select create to create the cluster.

## Multi-node Cluster

A high workload cluster may require the worker node to be placed in a separate pool.

* Select add nodes
* Add a name and use the same template as for the control nodes.

## Add IP’s to the security group

Add the external IP’s of the Nodes created into the security group as all traffic , the IP address suffixed by /32 e.g. ??.??.??.??/32

# Monitoring

In order to collect performance metrics from the Rancher cluster a monitoring tool must be started.

## Prometheus – Stats Gathering

Install the Prometheus reporter from the default catalogue.

## Grafana- Graphics

Deploy the grafana package from the default catalogue.

The load balancer tab will display the URL to logon.

Login with user id: admin , password :”rancher admin password”

From settings set Prometheus as the default data provider.

Copy in the Prometheus server url from the load balancer settings.

From Home select import dashboard

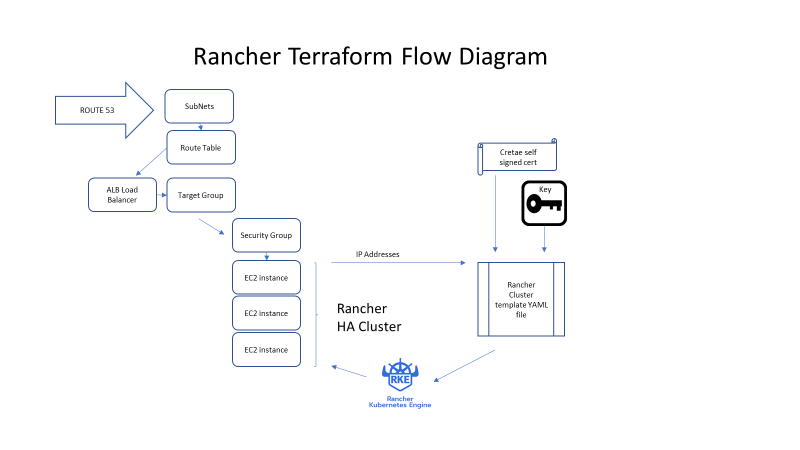
Dashboard: Kubernetes Node Exporter Full (dash-board ID: 3320 )

# High-Availability Web Guide

<https://rancher.com/docs/rancher/v2.x/en/installation/ha-server-install/>

# Terraform Scripts

## TF Scripts Diagram



The scripts are divided into two main functions, the first is to set up the AWS environment , the second is to create the certificate , private key and update the RKE YAML file with these values.

The IP addresses of the EC2 instances are substituted into the YAML file along with the certificate and the private key.

The RKE program is then run using the rancher cluster file as input. The RKE program creates a Kubernetes cluster on the 3 EC2 instances and installs Rancher into the cluster.

# LDAP Set-up

From the Security tab select Authentication.

Enter the following details.

-----------------------------------------------------------------------------------

LDAP server: 192.168.12.23

Service Accoutn ID with domain prefix: domain\id

Service account password: password

default domain: domain

Search Base=: search base

Authenticate with you ID and Password.

# Useful Commands De-Bugging

## Terraform Commands

### Check setup code - Plan

Check the current configuration using the plan command:

**Terraform plan**

### Run the code - Apply

Once the plan has run through correctly us the apply command to initiate the setup on AWS.

**Terrafrom apply**

### Remove the AWS environment - Destroy

Remove the entire setup from AWS using the destroy command. Use with care as it will remove all the infrastructure components for the cluster.

**Terraform destroy**

## Cluster Setup Commands

### Start the HA cluster

This is run by the setup script. If the Terraform scripts work but the RKE program fails then this command can be used to create the Kubernetes cluster for Rancher 2.

**rke up –config rancher-config.yaml**

### Remove the cluster

If the cluster setup fails the Kubernetes cluster can be removed by using the following command:

**rke remove –config rancher-config.yaml**

## Kubernetes Cluster Commands

### Show elements and logs

Display pods

kubectl --kubeconfig=kube\_config\_rancher-cluster.yaml get pods --all-namespaces

### Show the FQDNS name

The FQDNS name must match the Route53 “A” record for the network to initialize.

kubectl --kubeconfig kube\_config\_rancher-cluster.yaml get ingress -n cattle-system -o wide

### Display errors from the cattle system

kubectl --kubeconfig=kube\_config\_rancher-cluster.yaml log -l job-name=rke-user-addon-deploy-job -n kube-system

### Display network Logs

kubectl --kubeconfig=kube\_config\_rancher-cluster.yaml logs -l app=ingress-nginx -n ingress-nginx

### Display Ingress

kubectl --kubeconfig=kube\_config\_rancher-cluster.yaml describe ing -n cattle-system

### Display Rancher Logs

kubectl --kubeconfig=kube\_config\_rancher-cluster.yaml logs -l app=cattle -n cattle-system

kubectl --kubeconfig=kube\_config\_rancher-cluster.yaml logs canal-b6lvn -n kube-system -c kube-flannel

### Kubernetes Install Dashboard

kubectl --kubeconfig= kube\_config\_rancher-cluster.yaml --insecure-skip-tls-verify create -f https://raw.githubusercontent.com/kubernetes/dashboard/master/src/deploy/recommended/kubernetes-dashboard.yaml