mme



v1.0

06-September-2022

USER MANUAL

AWS CFT TEMPLATE 3NIC-2VM-HA

# PATENT PROTECTION

A10 Networks, Inc. products are protected by patents in the U.S. and elsewhere. The following website is provided to satisfy the virtual patent marking provisions of various jurisdictions including the virtual patent marking provisions of the America Invents Act. A10 Networks, Inc. products, including all Thunder Series products, are protected by one or more of U.S. patents and patents pending listed at: a10-virtual-patent-marking.

# TRADEMARKS

A10 Networks, Inc. trademarks are listed at: a10-trademarks.

# CONFIDENTIALITY

This document contains confidential materials proprietary to A10 Networks, Inc. This document and information and ideas herein may not be disclosed, copied, reproduced, or distributed to anyone outside A10 Networks, Inc. without prior written consent of A10 Networks, Inc.

# DISCLAIMER

This document does not create any express or implied warranty about A10 Networks, Inc. or about its products or services, including but not limited to fitness for a particular use and non-infringement. A10 Networks, Inc. has made reasonable efforts to verify that the information contained herein is accurate, but A10 Networks, Inc. assumes no responsibility for its use. All information is provided "as-is." The product specifications and features described in this publication are based on the latest information available; however, specifications are subject to change without notice, and certain features may not be available upon initial product release. Contact A10 Networks, Inc. for current information regarding its products or services. A10 Networks, Inc. products and services are subject to A10 Networks, Inc. standard terms and conditions.

# ENVIRONMENTAL CONSIDERATIONS

Some electronic components may possibly contain dangerous substances. For information on specific component types, please contact the manufacturer of that component. Always consult local authorities for regulations regarding proper disposal of electronic components in your area.

# FURTHER INFORMATION

For additional information about A10 products, terms and conditions of delivery, and pricing, contact your nearest A10 Networks, Inc. location, which can be found by visiting [www.a10networks.com](http://www.a10networks.com/).

Contents

[PATENT PROTECTION 1](#_Toc123819542)

[TRADEMARKS 1](#_Toc123819543)

[CONFIDENTIALITY 1](#_Toc123819544)

[DISCLAIMER 1](#_Toc123819545)

[ENVIRONMENTAL CONSIDERATIONS 1](#_Toc123819546)

[FURTHER INFORMATION 2](#_Toc123819547)

[Introduction to Installing vThunder on AWS 5](#_Toc123819548)

[Overview of AWS 5](#_Toc123819549)

[Aws Terminology 6](#_Toc123819550)

[CloudFormation Template – 3NIC \_2VM \_HA 7](#_Toc123819551)

[Overview 7](#_Toc123819552)

[Prerequisites 7](#_Toc123819553)

[AWS Account and Environment Setup to Run CFT Template [One Time Step] 7](#_Toc123819554)

[AWS key-pairs to access ec2 instances. 9](#_Toc123819555)

[Steps to create a SSH key [Optional] 9](#_Toc123819556)

[System Requirements 10](#_Toc123819557)

[Stack 10](#_Toc123819558)

[Chapter 1 – Install vADC 14](#_Toc123819559)

[Install 14](#_Toc123819560)

[Chapter 2 Configure vThunder 20](#_Toc123819561)

[configure 20](#_Toc123819562)

[Chapter 3- Install Client/Server VM 30](#_Toc123819563)

[Install 30](#_Toc123819564)

[Chapter 4-HA Setup 31](#_Toc123819565)

[Configuration 31](#_Toc123819566)

[Install 32](#_Toc123819567)

[Chapter 5-Let us Verify 33](#_Toc123819568)

[Slb verification 34](#_Toc123819569)

[SSL verification 35](#_Toc123819570)

[HA verification 35](#_Toc123819571)

[Traffic Flow from Client to Server via vThunder 38](#_Toc123819572)

# Introduction to Installing vThunder on AWS

vThunder for Amazon Web Services is a fully operational, software-only version of the ACOS Series Server Load Balancer (SLB), or Application Delivery Controller (ADC) device. It is configurable by ACOS CLI, GUI, AXAPI, and Harmony Controller. For more information see Virtual Instances in Harmony Controller.

vThunder is a virtual appliance, yet it retains most of the functionality available on the hardware based ACOS appliances. Managing vThunder is the same as managing hardware based ACOS device, and vThunder has the same CLI configurations and GUI presentation.

The networking configuration for vThunder is also like hardware based ACOS devices.

A10 Networks brings Out-Of-Box template to deploy vThunder along with multiple features and functionality with pre-defined format into amazon cloud.

Please refer below section for more details.

* Three Network Card Interface (3NIC).
* TLS/SSL Certification (SSL).
* Server Load Balancer (SLB)

# Overview of AWS

Amazon Web Services offers a broad set of global cloud-based products including compute, storage, databases, analytics, networking, mobile, developer tools, management tools, IoT, security, and enterprise applications: on-demand, available in seconds, with pay-as-you-go pricing. From data warehousing to deployment tools, directories to content delivery, over 200 AWS services are available. New services can be provisioned quickly, without the upfront fixed expense.

AWS uses the following tools to create and manage resources:

**AWS Portal** - A web console to create and monitor AWS resources. For more information: https://aws.amazon.com/console/

**AWS CLI—** The AWS CLI enables you to start running commands that implement functionality equivalent to that provided by the browser-based AWS Management Console from the command prompt in your terminal program:

* **Linux shells** – Use common shell programs such as [bash](https://www.gnu.org/software/bash/), [zsh](http://www.zsh.org/" \t "_blank), and [tcsh](https://www.tcsh.org/) to run commands in Linux or macOS.
* **Windows command line** – On Windows, run commands at the Windows command prompt or in PowerShell.
* **Remotely** – Run commands on Amazon Elastic Compute Cloud (Amazon EC2) instances through a remote terminal program such as PuTTY or SSH, or with AWS Systems Manager. For more information: http//docs.aws.amazon.com/cli/index.html?nc2=h\_ql\_doc\_cli

Diagram

Description automatically generated

# Aws Terminology

**Access control list (ACL):** A firewall/security layer on the subnet level. For more information <https://docs.aws.amazon.com/AmazonS3/latest/userguide/acls.html>

**CloudWatch:** Service that allows you to monitor various elements of your AWS account. For more information: <https://docs.aws.amazon.com/cloudwatch/index.html>

**Lambda:** Serverless computing that will replace EC2 instances, for the most part. For more information: <https://docs.aws.amazon.com/lambda/latest/dg/welcome.html>

**Security group (SG):** Firewall/security layer on the server/instance level. For more information [https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-security- groups.html](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-security-%20groups.html)

**Subnet:** A subsection of a network and generally includes all the computers in a specific location. For more information: <https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-resource-ec2-subnet.html>

**Virtual Private Cloud (VPC):** A private subsection of AWS you control and in which you can place AWS resources. For more information: <https://docs.aws.amazon.com/vpc/latest/userguide/what-is-amazon-vpc.html>

# CloudFormation Template – 3NIC \_2VM \_HA

## Overview

CloudFormation template to create 3NIC 2 vThunder instances on AWS portal.

## Prerequisites

Generate access\_key\_id and secret\_access\_key if you don’t have it. For more details visit:

<https://docs.aws.amazon.com/powershell/latest/userguide/pstools-appendix-sign-up.html>

### AWS Account and Environment Setup to Run CFT Template [One Time Step]

1. Download and install python setup using following link:

<https://www.python.org/ftp/python/3.8.5/python-3.8.5-amd64.exe>

1. To verify that Python is available on our local machine, we need to open the command line (in Windows search, type cmd and press Enter to open Command Prompt or right-click on the Start button and select Windows Command Prompt), type python, and press Enter.
2. If Python is properly installed, we will see a notification like the one below:

*Python 3.8.x (tags/v3.10.2:a58ebcc, Jan 17 2022, 14:12:15) [MSC v.1929 64 bit (AMD64)] on win32 Type "help," "copyright," "credits," or "license" for more information.*

1. To check if PIP is already installed on Windows, we should open the command line again, type pip, and press Enter.
2. If PIP is installed, we will receive a long notification explaining the program usage, all the available commands and options. Otherwise, if PIP is not installed, the output will be:

*'pip' is not recognized as an internal or external command, operable program or batch file.*

1. To install pip on window visit:

[https://pip.pypa.io/en/stable/installation](https://pip.pypa.io/en/stable/installation/)

1. Install all dependencies go to current working directory and use following command:

pip install -r requirements.txt

1. Locate and open /credentials in current working directory.
2. Change the access key as well as secret access key as per your aws account.

Text, letter

Description automatically generated

*Copy credentials file to C:\Users\<USERNAME>\.aws*

1. Locate and open /config in current working directory.
2. Change aws region details.

Graphical user interface, text

Description automatically generated with medium confidence

1. Copy AWS config file on your local system, located at:

*C:\Users\<USERNAME>\.aws*

**For linux, macOS, Unix**:

1. Locate and open /credentials.
2. Change the access key as well as secret access key as per your aws account.

Text, letter

Description automatically generated

*Copy credentials file to ~/.aws*

1. Locate and open /config
2. Change aws region details.

Graphical user interface, text

Description automatically generated with medium confidence

1. Copy AWS config file on your local system, located at:

*~/.aws*

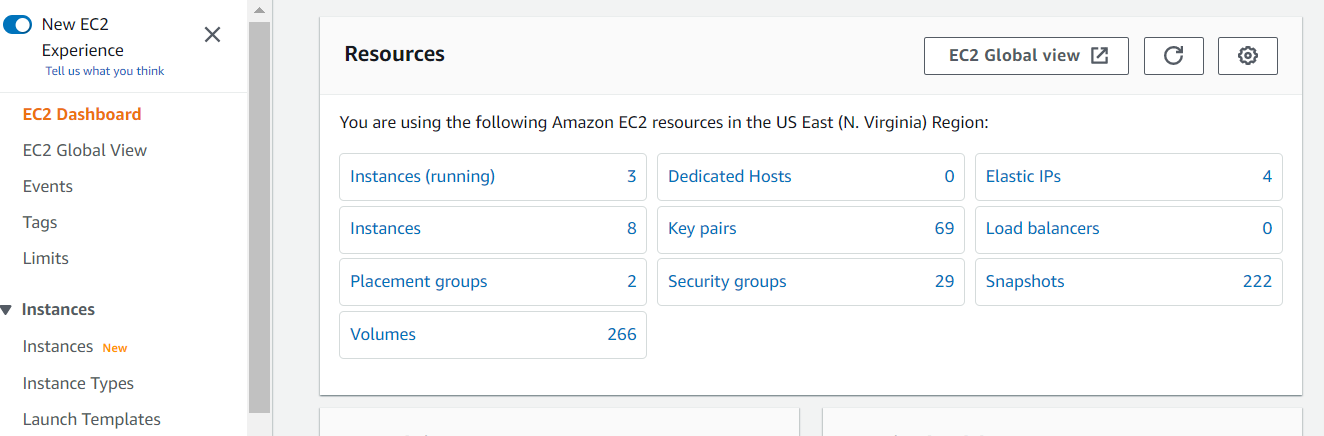
For more information: https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-credentials.html

### AWS key-pairs to access ec2 instances.

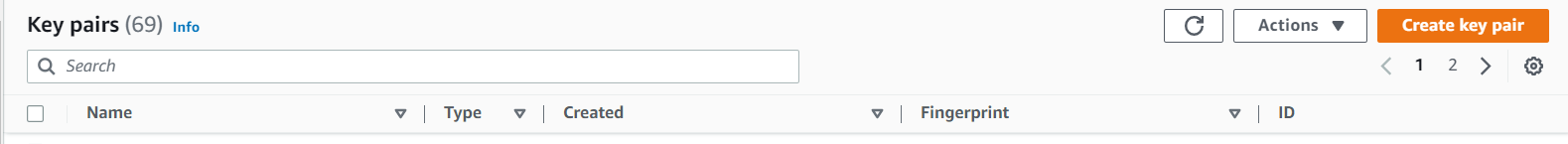
## Steps to create a SSH key [Optional]

Note: You can use existing key pair if you have key pair already created.

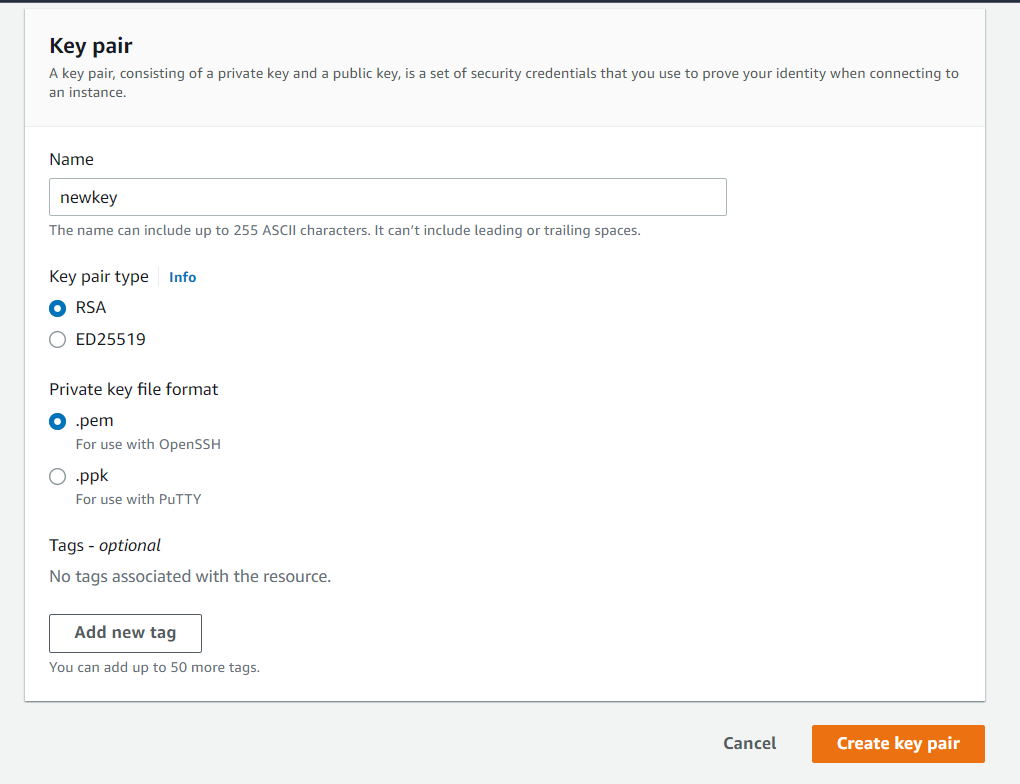
1. Go to the EC2 dashboard and click on key pairs.



1. Now click on create key pair at the top right.



1. Now name the key and select the key pair type as RSA and key file format as .pem.



## System Requirements

Below all AWS cloud resources will be created.

All templates come with default value it can be change while execution.

Stack

A new stack is created with the specified name and location.

Interfaces

For both vThunder 1 management and 2 data interfaces will be created.

Default names:

|  |
| --- |
| *<vth> -inst1-mgmt-nic1* |
| *<vth> -inst1-data-nic2* |
| *<vth> -inst1-data-nic3* |
| *<vth> -inst2-mgmt-nic1* |
| *<vth> -inst2-data-nic2* |
| *<vth> -inst2-data-nic3* |

Subnets

Total 3 subnets will be created.

Default names:

*<vth> -vpc-mgmt-sub1*

*<vth> -vpc-data-sub1*

*<vth> -vpc-data-sub2*

Virtual Private Network

A virtual Private network will be created. Address prefix is 10.0.0.0/16.

Default name*: <vth> -vpc*

Elastic Public Ip

Elastic Public Ip will be created and attached to management interface of vThunder instance.

Default name:

*<vth> -inst1-mgmt-nic1-ip*

*<vth> -inst2-mgmt-nic1-ip*

Security Group

Two Security Groups will be created and attached to management interface.

Default name: *<vth> -sg-mgmt*

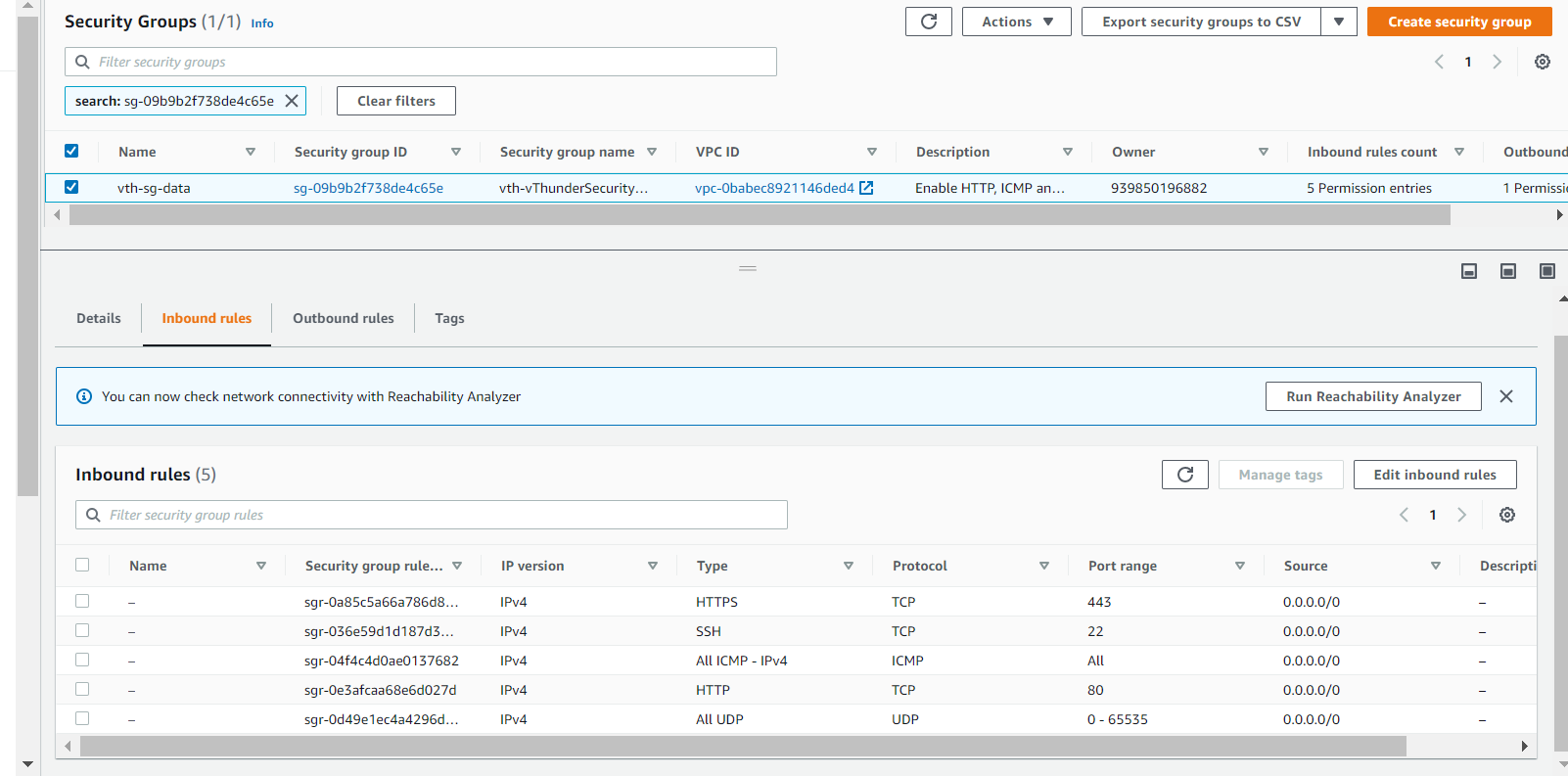
*<vth> -sg-data*

Path: *Stacks>> vth>> Resources>> vThunderSecurityGroupMgmt/ vThunderSecurityGroupData*

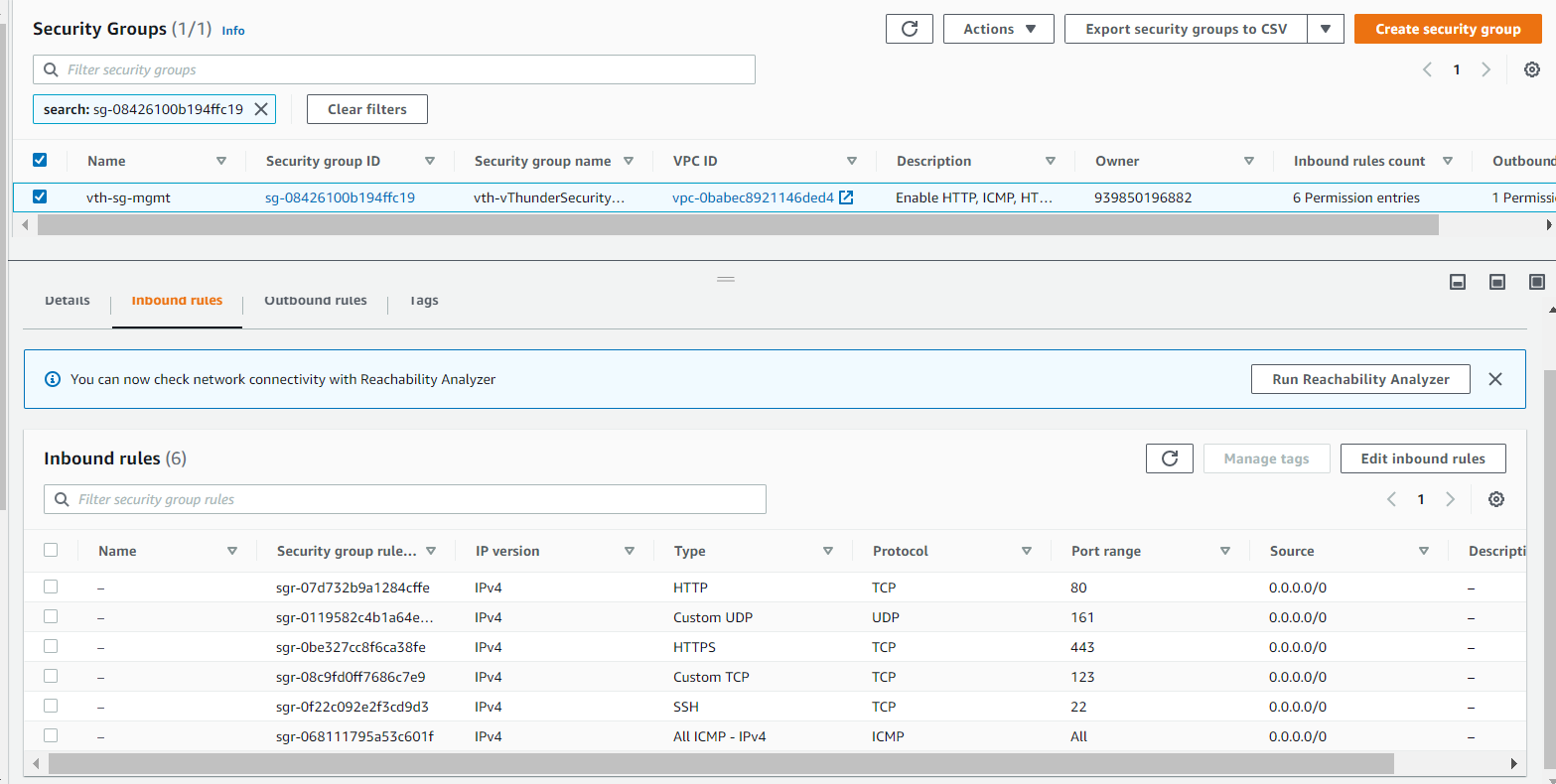
Note: If you want to add new rule to security group then select security group and then edit inbound rule and add rule and save.

Security Rules:

vth-sg-data



vth-sg-mgmt



vThunder Instance

2 vThunder instances will be created. (one active and one standby)

**Default Size**: m4.xlarge (40 Gb memory)

**Default name**: *<vth> -inst1, <vth> -inst2*

Ubuntu Server Instance

One Ubuntu Server instances will be created.

**Default Size**: t2.micro

**Default name**: *<vth>-server*

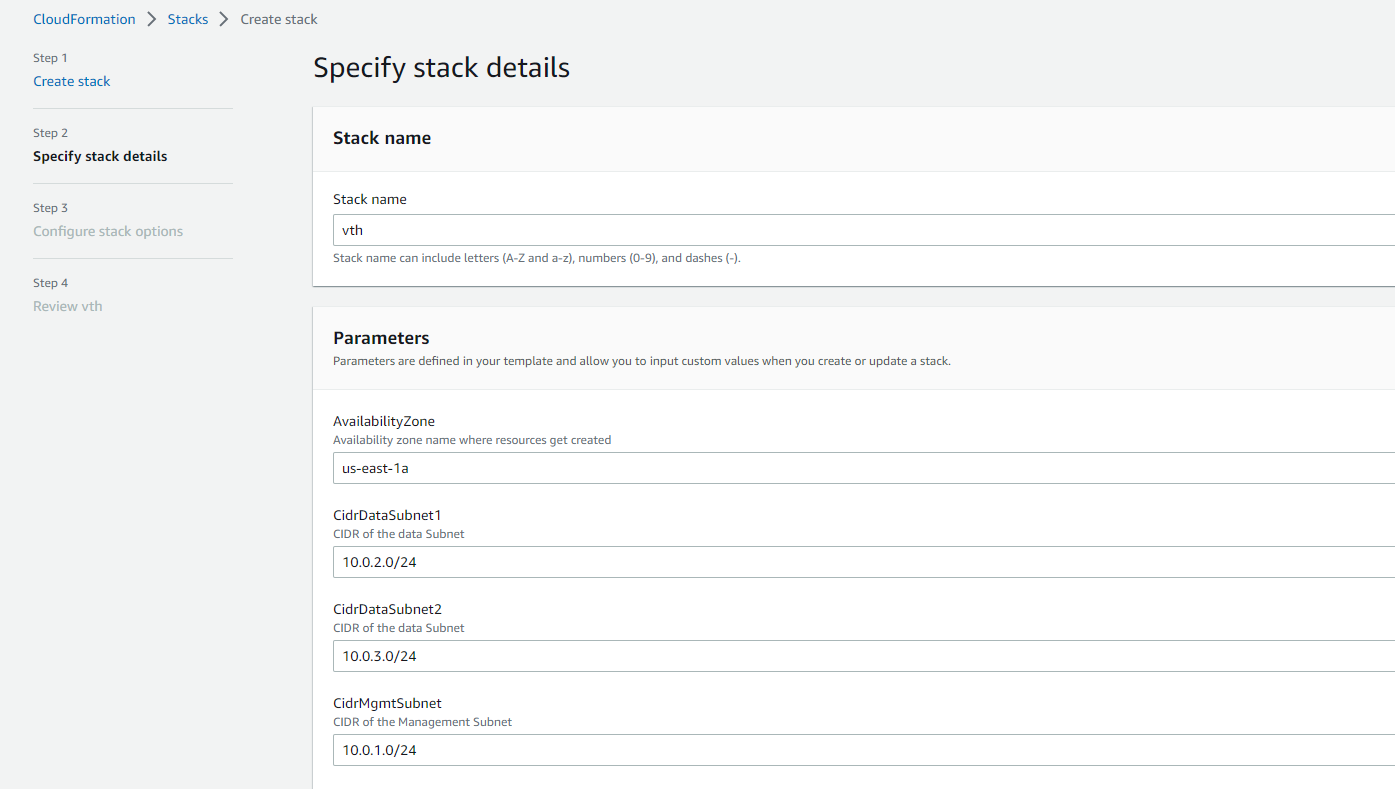
# 

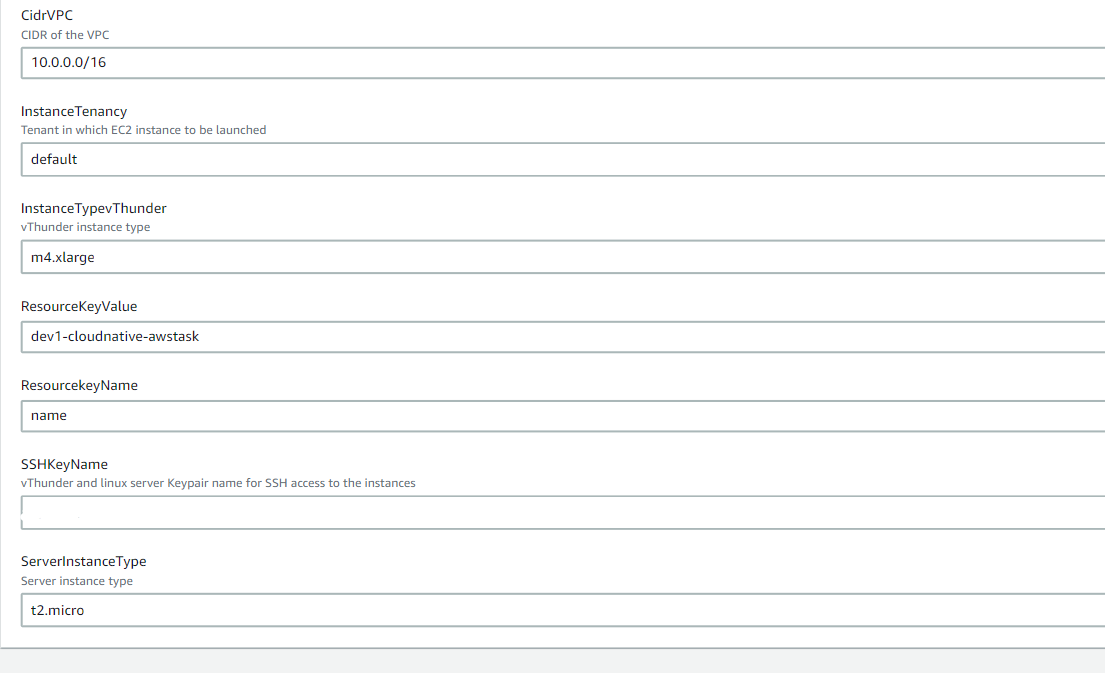
# Chapter 1 – Install vADC

All resources will get created in AWS.

## Install

1. Navigate to AWS Console -> CloudFormation -> Stacks -> Create Stack
2. Select prepare template.
3. Select “Upload a template file”.
4. Choose file CFT\_TMPL\_3NIC\_2VM\_HA\_1.json file
5. Go to next.
6. Provide stack name. For example: <vth>
7. Provide availability zone. For example: us-east-1a
8. Provide tagValue. For example: <a10-vthunder-adc>
9. Confirm default values.

-



1. Below listed sizes are verified for vThunder.

|  |  |  |  |
| --- | --- | --- | --- |
| **Instance** | **vCPU** | **Memory** | **Number of Network Interfaces** |
| c4.xlarge | 4 | 7680 | 4 |
| c4.4xlarge | 16 | 30720 | 8 |
| c4.8xlarge | 36 | 61440 | 8 |
| d2.xlarge | 4 | 31232 | 4 |
| d2.2xlarge | 8 | 62464 | 4 |
| d2.4xlarge | 16 | 124928 | 8 |
| d2.8xlarge | 36 | 249856 | 8 |
| m4.xlarge | 4 | 16384 | 4 |
| m4.2xlarge | 8 | 32768 | 4 |
| m4.4xlarge | 16 | 65536 | 8 |
| m4.10xlarge | 40 | 163840 | 8 |
| i2.xlarge | 4 | 31232 | 4 |
| i2.2xlarge | 8 | 62464 | 4 |
| i2.4xlarge | 16 | 124928 | 8 |
| i2.8xlarge | 32 | 249856 | 8 |
| c5d.large | 2 | 4096 | 3 |
| c5d.9xlarge | 36 | 73728 | 8 |
| c5d.2xlarge | 8 | 32768 | 4 |
| c5d.4xlarge | 16 | 73728 | 8 |
| c5.xlarge | 4 | 8192 | 4 |
| c5.2xlarge | 8 | 16384 | 4 |
| c5.4xlarge | 16 | 32768 | 8 |
| c5.9xlarge | 36 | 73728 | 8 |
| g3.4xlarge | 16 | 124928 | 8 |
| g3.8xlarge | 32 | 249856 | 8 |
| i3.large | 2 | 15616 | 3 |
| i3.xlarge | 4 | 31232 | 4 |
| i3.2xlarge | 8 | 62464 | 4 |
| i3.4xlarge | 16 | 124928 | 8 |
| i3.8xlarge | 32 | 249856 | 8 |
| m5d.large | 2 | 8192 | 3 |
| m5d.xlarge | 4 | 16384 | 4 |
| m5d.2xlarge | 8 | 32768 | 4 |
| m5d.4xlarge | 16 | 65536 | 8 |
| m5.large | 2 | 8192 | 3 |
| m5.xlarge | 4 | 16384 | 4 |
| m5.2xlarge | 8 | 32768 | 4 |
| m5.4xlarge | 16 | 65536 | 8 |
| r5d.large | 2 | 16384 | 3 |
| r5d.xlarge | 4 | 32768 | 4 |
| r5d.2xlarge | 8 | 65536 | 4 |
| r5d.4xlarge | 16 | 131072 | 8 |
| r5.large | 2 | 16384 | 3 |
| r5.xlarge | 4 | 32768 | 4 |
| r5.2xlarge | 8 | 65536 | 4 |
| r5.4xlarge | 16 | 131072 | 8 |
| r4.large | 2 | 15616 | 3 |
| r4.xlarge | 4 | 31232 | 4 |
| r4.2xlarge | 8 | 62464 | 4 |
| r4.4xlarge | 16 | 124928 | 8 |
| r4.8xlarge | 32 | 249856 | 8 |
| t3.medium | 2 | 4096 | 3 |
| t3.large | 2 | 8192 | 3 |
| t3.xlarge | 4 | 16384 | 4 |
| t3.2xlarge | 8 | 32768 | 4 |
| z1d.large | 2 | 16384 | 3 |
| z1d.xlarge | 4 | 32768 | 4 |
| z1d.2xlarge | 8 | 65536 | 4 |
| z1d.3xlarge | 12 | 98304 | 8 |
| z1d.6xlarge | 24 | 196608 | 8 |

11) Remain others as default.

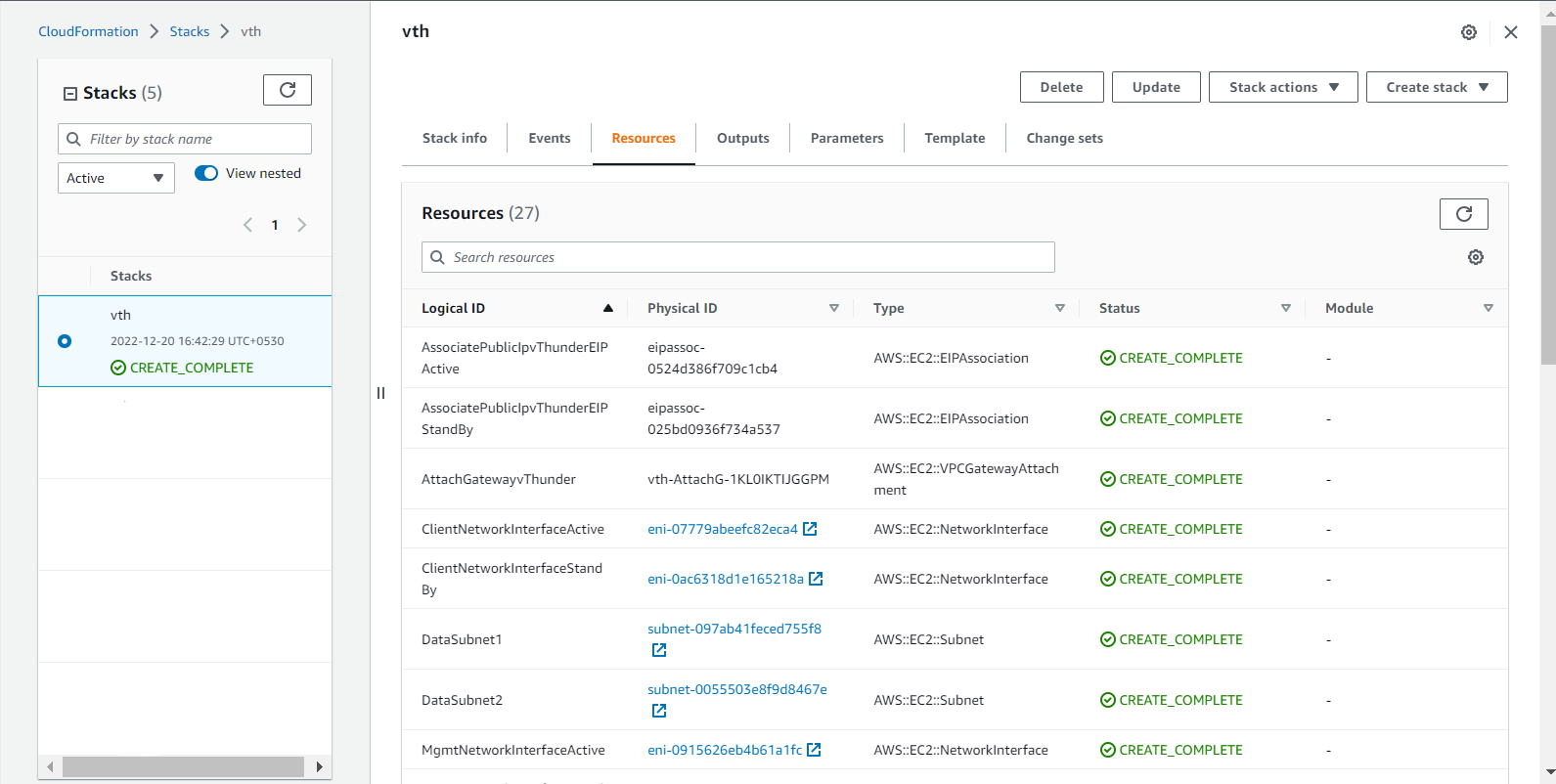
12) Go to next.

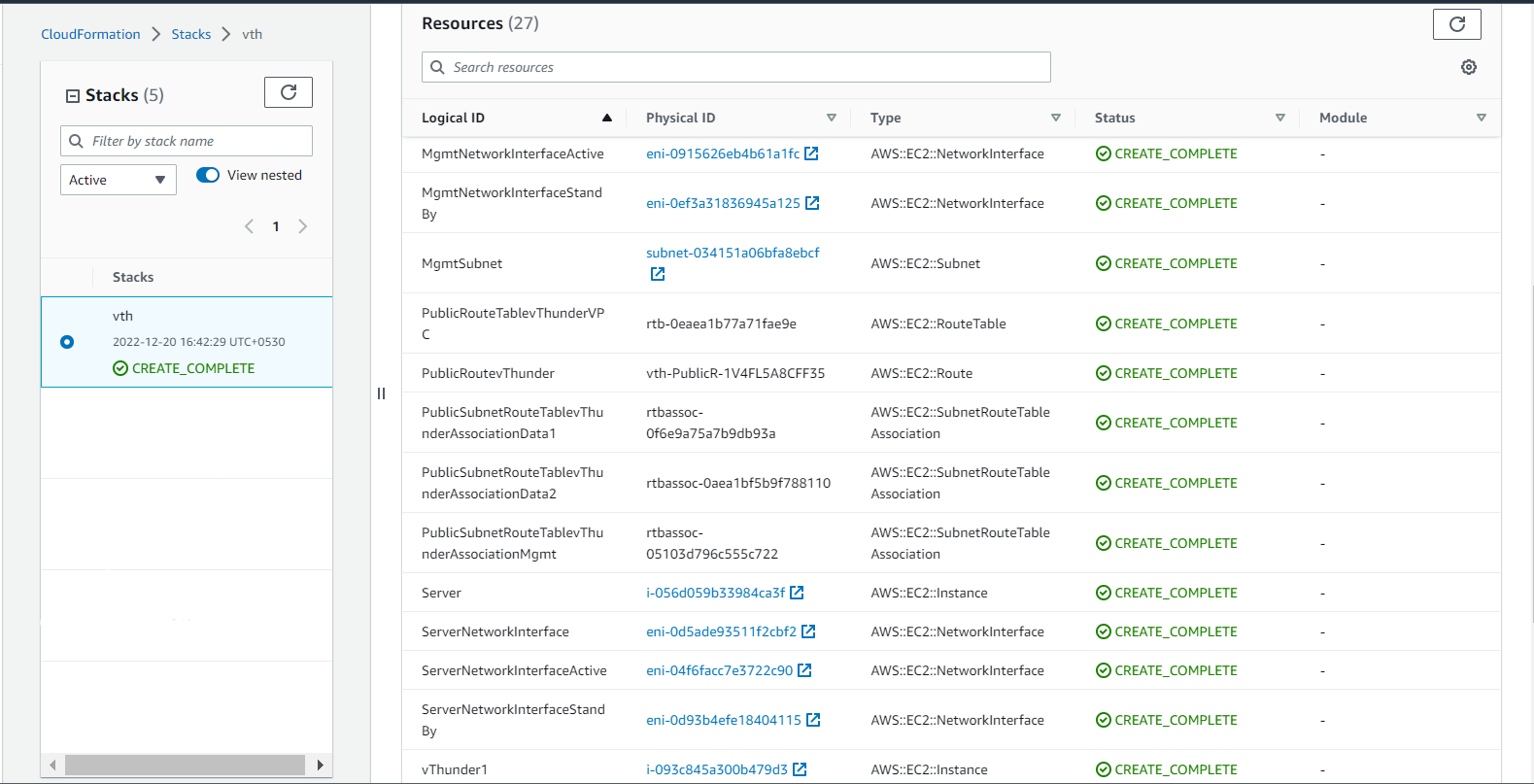
13) Review

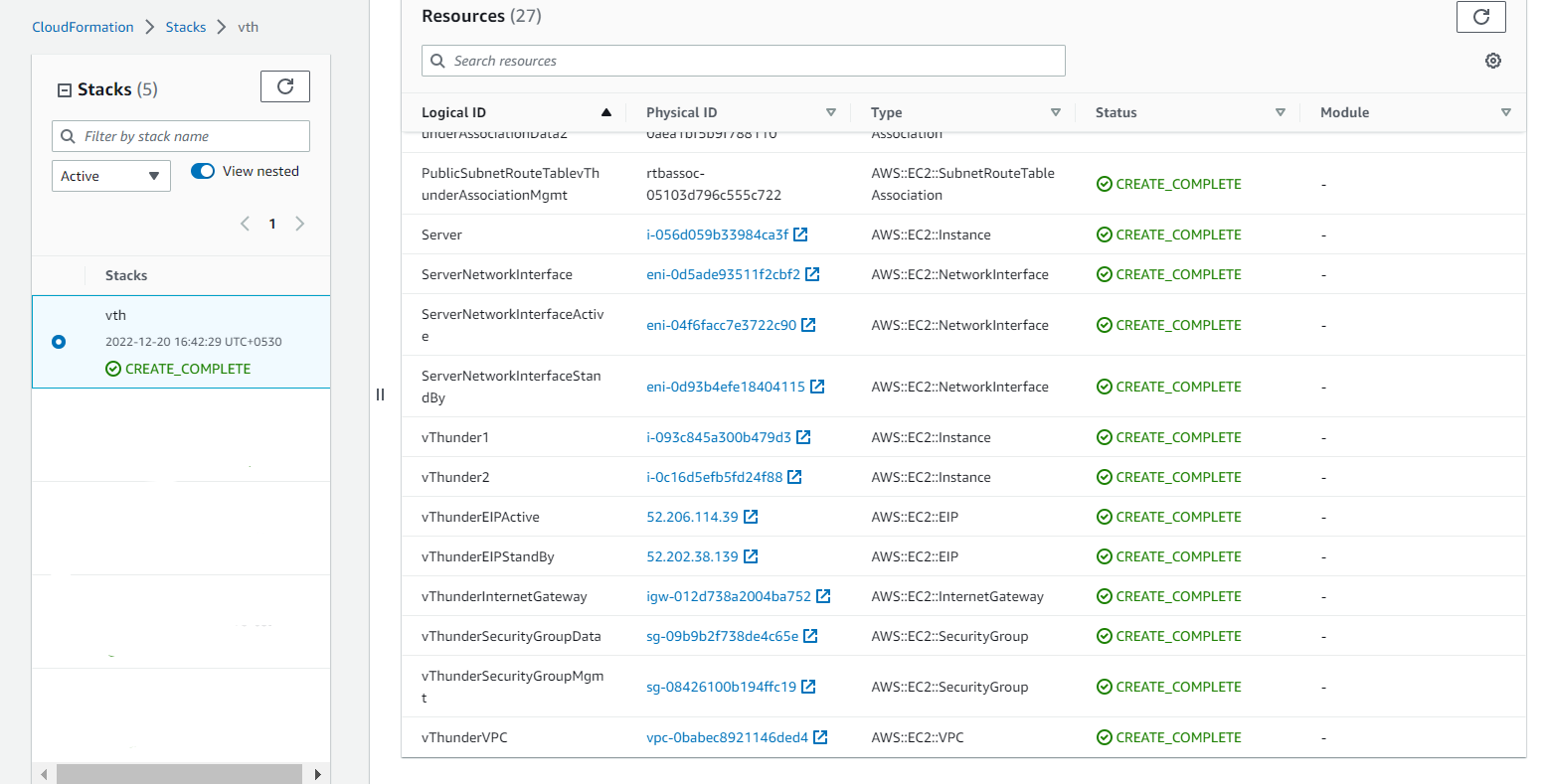
14) Submit

***Set Back and Relax, it will take maximum 10 mins. 😊***

Go to AWS Console -> *CloudFormation-> Stacks-> {stack name}*







15) Verify all above resources created.

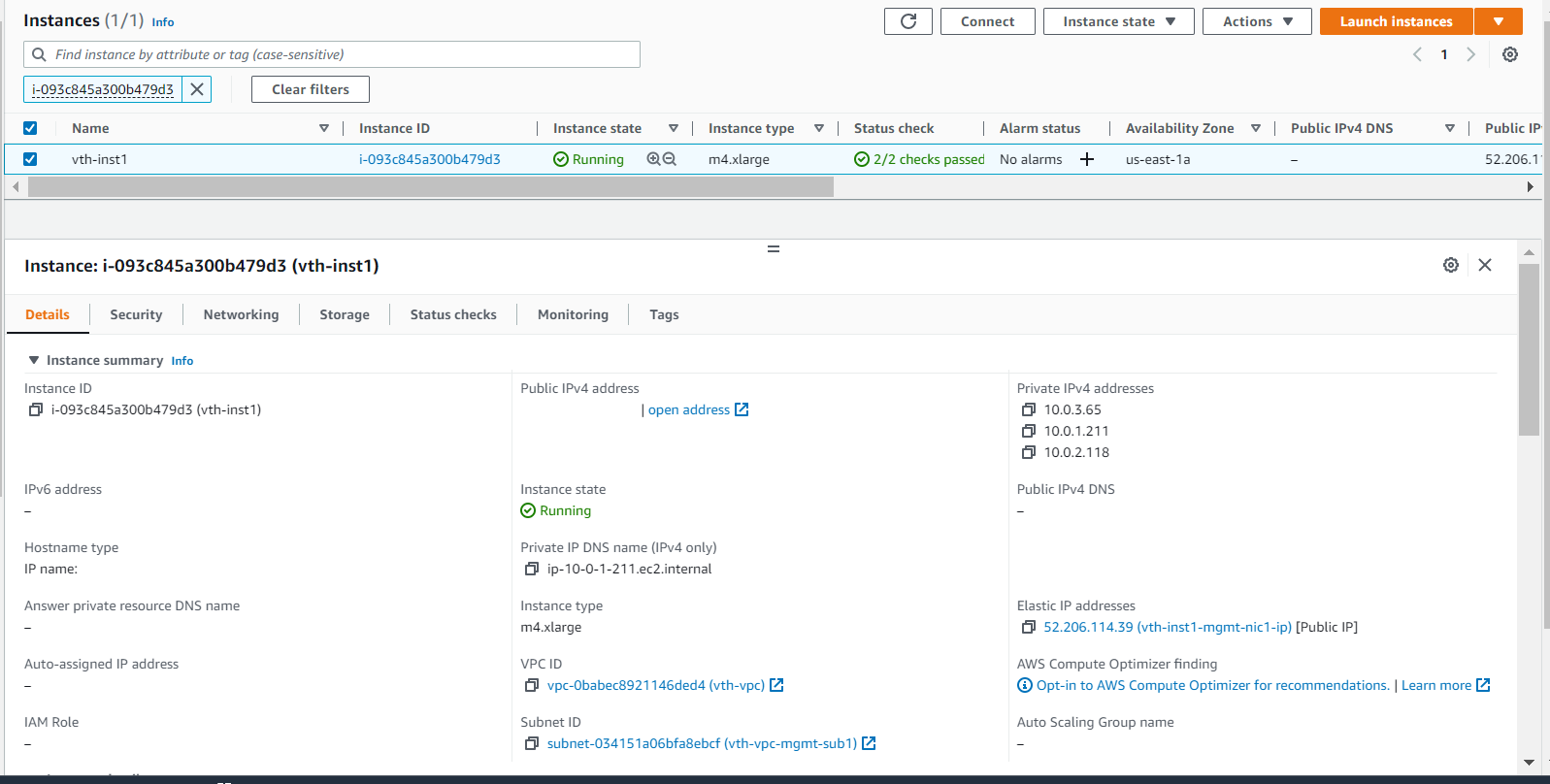
***Verify status check of vThunder instance we created.***

EC2-> Instances*-> <vth>-inst1*

Open any browser and type http://<vthunder\_public\_IP>

Enter username – admin

Enter Password – {ec2 instance id}



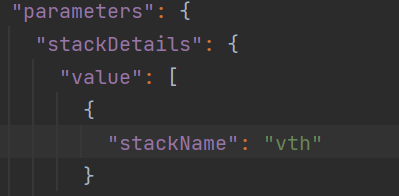


# Chapter 2 Configure vThunder

## configure

Here we will run python script and which will configure SLB and SSL to vThunder.

1. In parameter file *CFT\_TMPL\_3NIC\_2VM\_HA\_CONFIG\_SLB\_SSL\_HA\_PARAM.json* add stack name which got created using CFT template.
2. Default Name: vth



**SSL parameters**

1. Path- Should be the absolute path of the file
2. File- Name of the file



Note: Supported certification type *.pem*.

**SLB**

1. User can add or remove ports from below parameter.



1. **Service Group List**
2. Default service group list

Note: Service group name by default are **“sg+port\_number”**, If you want to change service group name then after changing name do respective changes in Virtual servers also.





1. **Services Details**

Default values of service group is **“sg+port\_number”.** You can modify service details in parameter file.





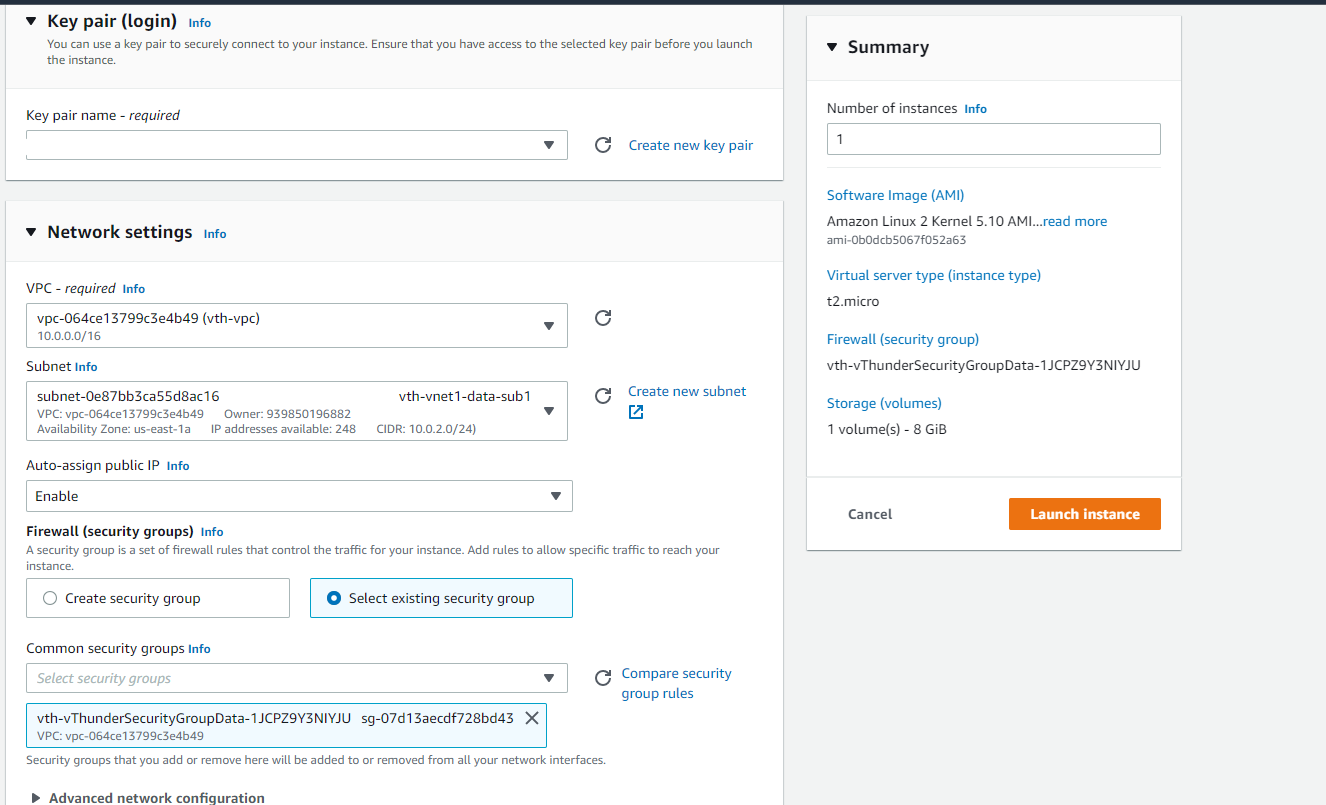
**Import AWS keys [Using FTP Server]**

Note: Below steps are for transferring AWS keys to vThunder instance. Here we need to create FTP server. You can use existing FTP server if you have [Skip step a].

1. Launch an ubuntu EC2 instance in the same subnet as that of the instance 1.

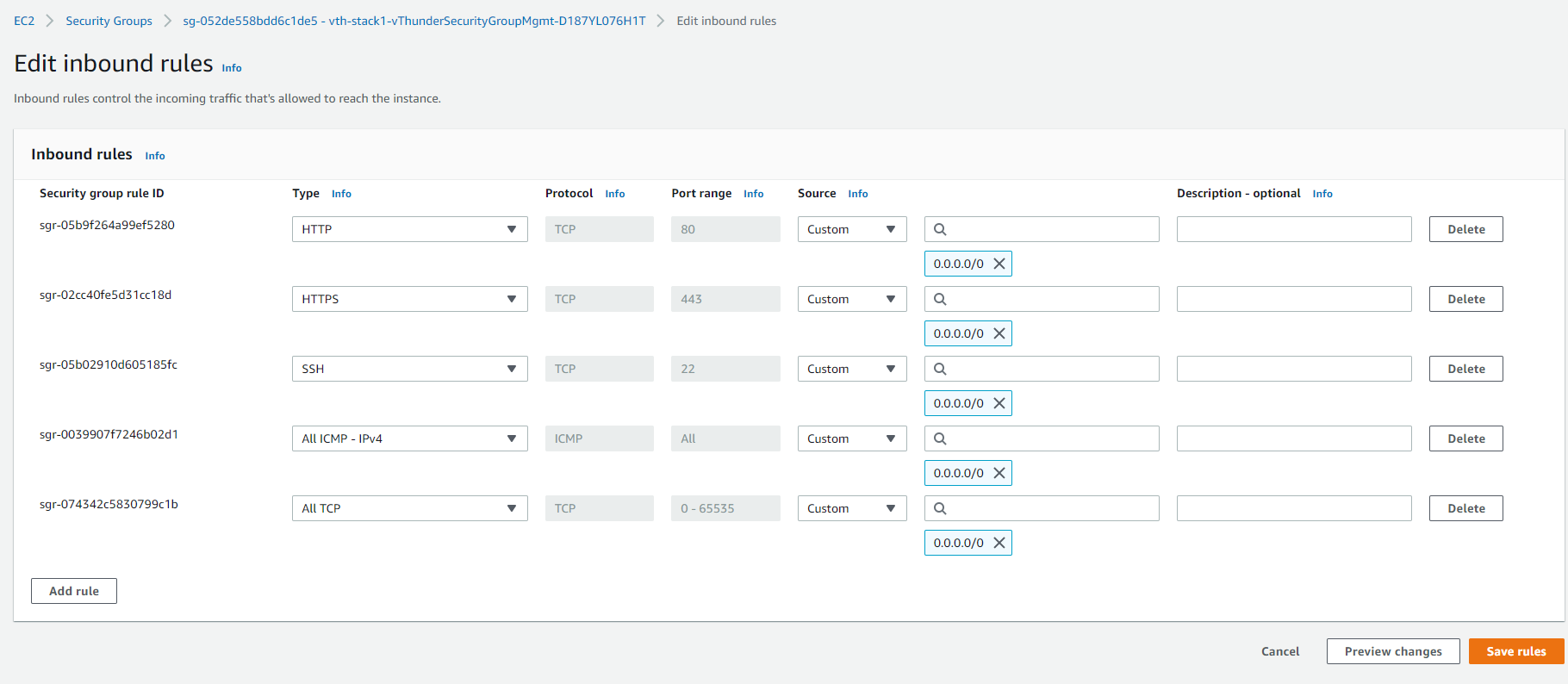
Graphical user interface, text, application, email

Description automatically generated



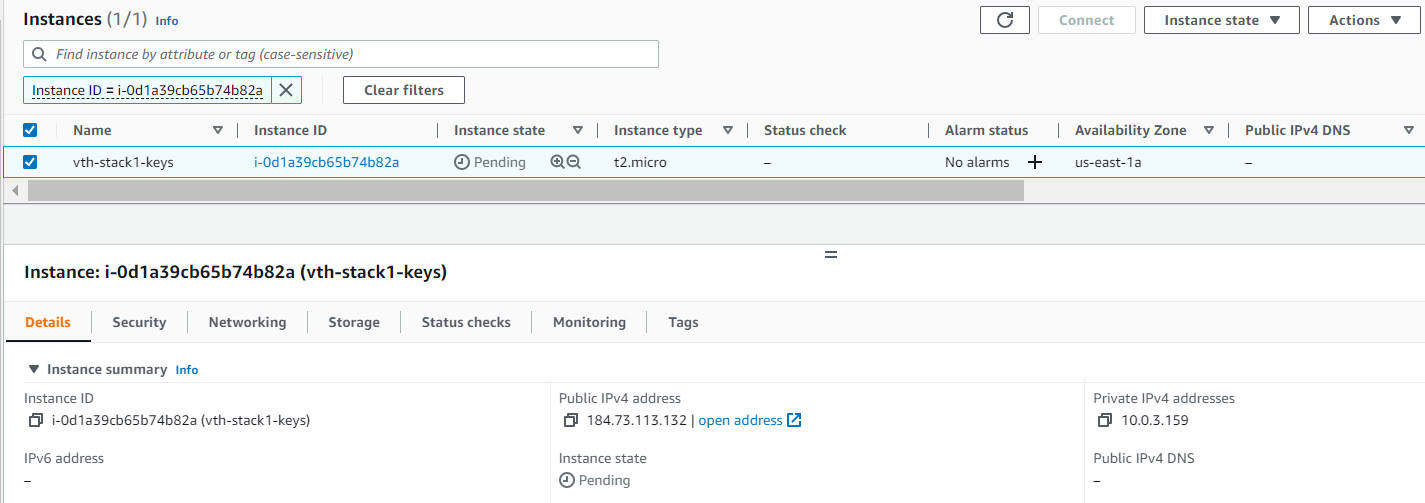
1. Add new rule in security group.

Path: EC2>> Security Groups>> {stack-Name} -sg-data>> Edit inbound rules

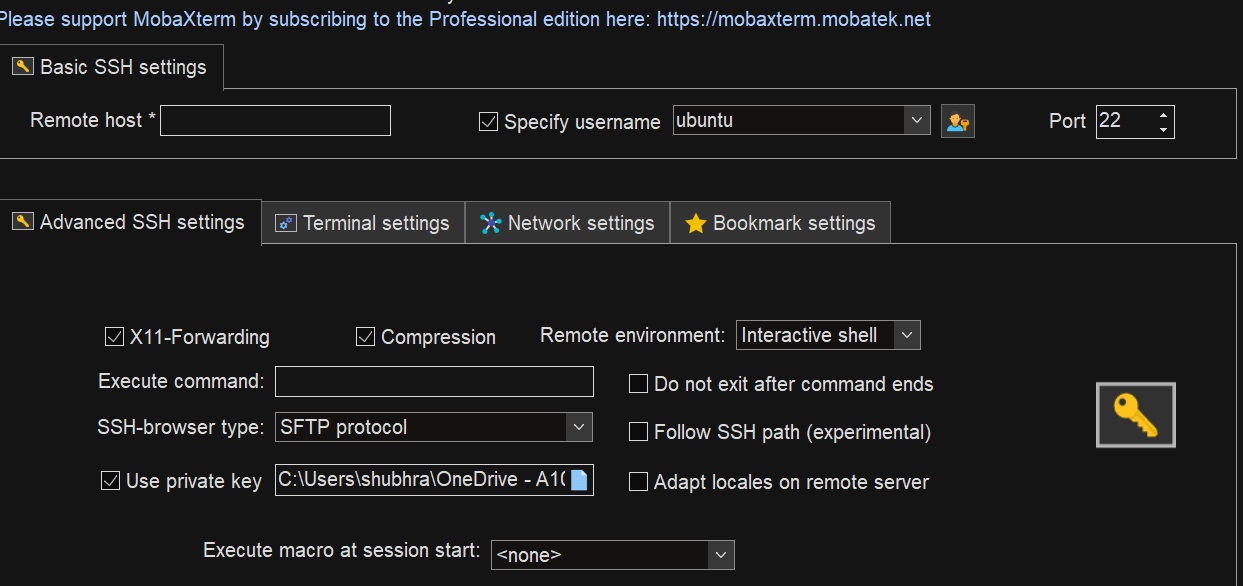


Note: All TCP rule is only to transfer AWS keys to vthunder instance.

1. Now copy the public ip of the instance.

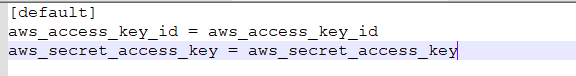


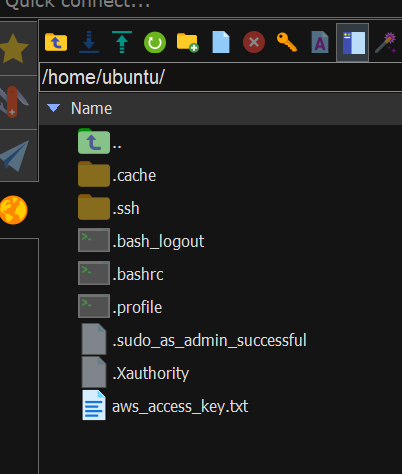
1. Paste this Ip to new session window of MobaxTerm and paste it in host.



1. Upload the access key in the instance.
2. Create aws\_access\_key.txt and add aws\_access\_key\_id as well as aws\_ secret\_access\_key.

Example:



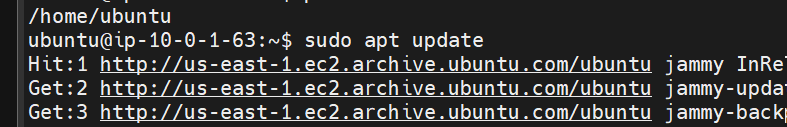


1. Run the command given below to provide the uploaded file permission with read and write permission.

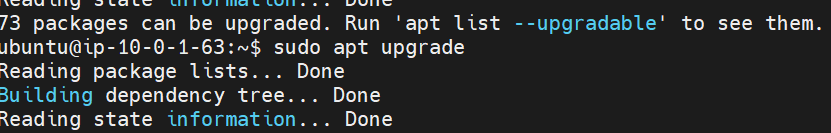
sudo chmod 777 aws\_access\_key.txt

1. Run the following commands.

sudo apt update

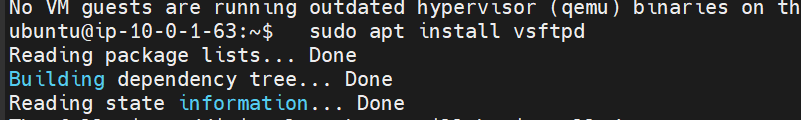


sudo apt upgrade



1. Run the following command to download ftp

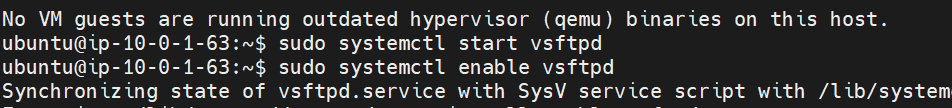
sudo apt-get install vsftpd



1. Run the following command to start the ftp.

sudo systemctl start vsftpd

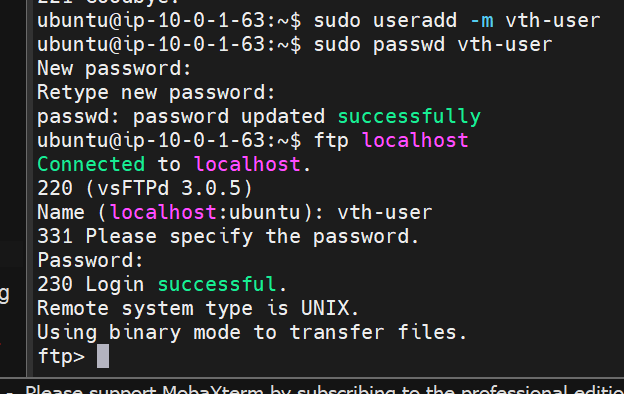
sudo systemctl enable vsftpd



1. Run the following commands to add the new user and set password.

sudo useradd -m vth-user

sudo passwd vth-user

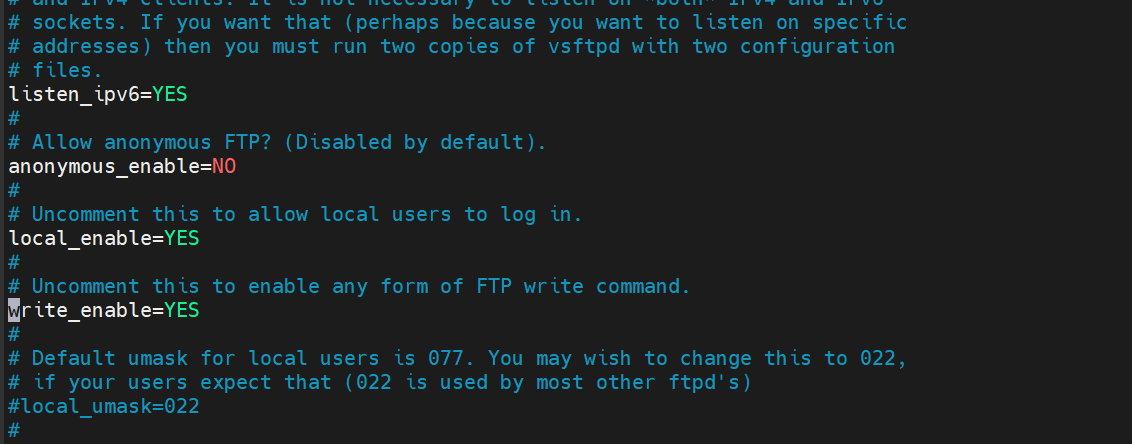


1. Go to the folder where you have uploaded AWS keys.

cd /etc/

1. Now give it write permission

sudo vi vsftpd.conf



1. Restart the ftp using commands

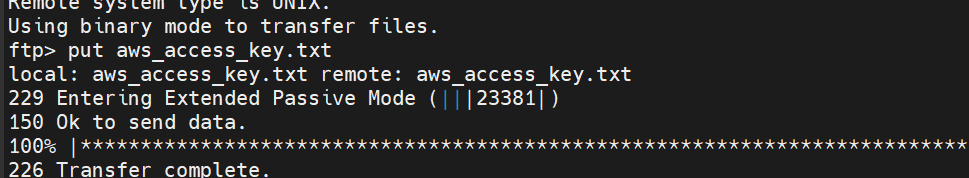
sudo systemctl restart vsftpd

cd /home/ubuntu

ftp localhost

1. Put the access keys in the instance.

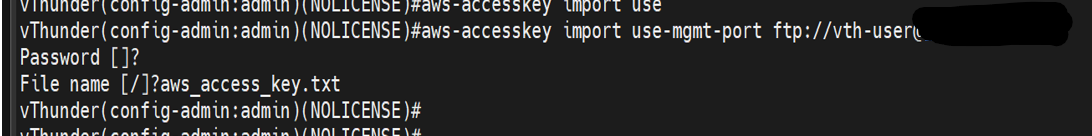
put aws\_access\_key.txt



1. Now add all TCP rule in inbound rules of the security group.
2. Now access the both the vthunder using its public ip and import the access keys using the given command to both of them.

admin admin

aws-accesskey import use-mgmt-port <ftp://vth-user@publicipof>theinstancewithkeys



You can check keys with following command.

aws-accesskey show

# Chapter 3- Install Client/Server VM

Client VM and Server VM is a temparatory vm which can be deleted later. This will help to test traffic. We can test traffic from client vm to server vm via vThunder.

## Install

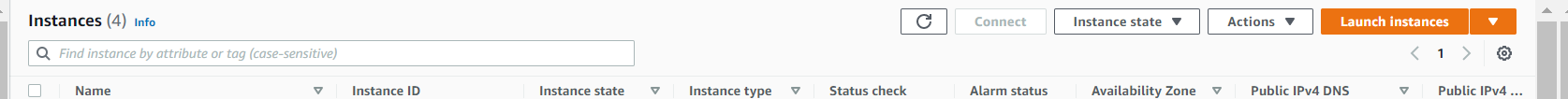
Apache Server VM:

Go to AWS Console -> *EC2 -> Instances - > Select <vth>-server -> connect -> EC2 instance connect -> Connect*

Run : sudo apt install apache2

Client VM:

1. Go to AWS Console -> *EC2 -> Instances -> Launch instances*



1. Provide instance name. For example: <vth-client>.
2. Select image as Ubuntu.
3. Select keypair as you created earlier.
4. Edit the network settings. Select VPC <vth-vpc>.
5. Select Data subnet 1. (e.g 10.0.2.0/24)
6. Edit Auto Assign Public IP to enable.
7. Edit Firewall (security groups) and select existing security group <vth-vThunderSecurityGroupData>.
8. Click on launch instance.
9. Wait for 5 min.

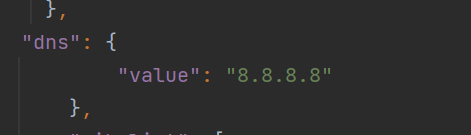
# Chapter 4-HA Setup

## Configuration

Parameter file

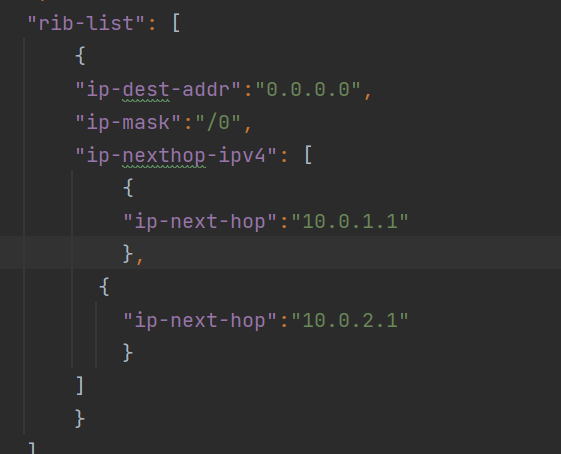
**DNS**

* 1. Default value is google dns address.

****

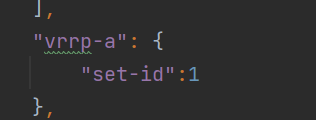
**Network Gateway IP**

1. Default value of network gateway ip address is 10.0.1.1 since this is first ip address of default management subnet configuration.



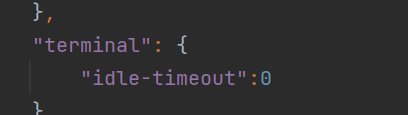
**Vrrp**

1. Default value of set id is 1



Terminal Idle Timeout

1. Default value for idle timeout is 0



Vrid

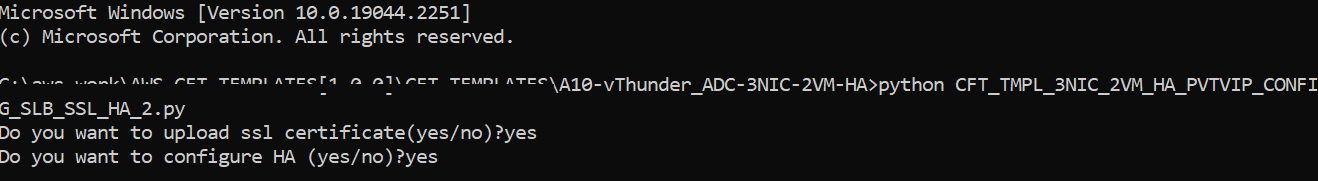
1. Default value of vrid is 0
2. Default priority for vThunder-1 is 100, and vThunder-2 will have 99 (100-1) priority.

## Install

Open your CMD in current working directory.

Run : Python ./CFT\_TMPL\_3NIC\_2VM\_HA\_CONFIG\_SLB\_SSL\_HA\_2.py

Provide below configuration params:



## 

## Chapter 5-Let us Verify

vThunder can be access by ssh to instance or GUI.

SSH to vThunder Instance:

Open MobaXterm and connect.

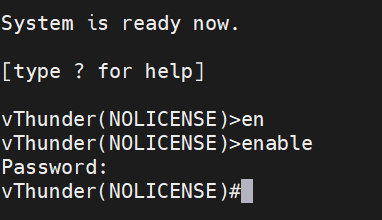
IP: Get from AWS Console -> Cloudformation template -> created stack name -> Resources-> vthunder->public ip.

Username [Default]: admin

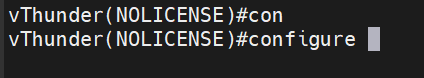
After login.

Execute Command -> *enable*

Password -> <just press enter>



Execute Command -> *configure*

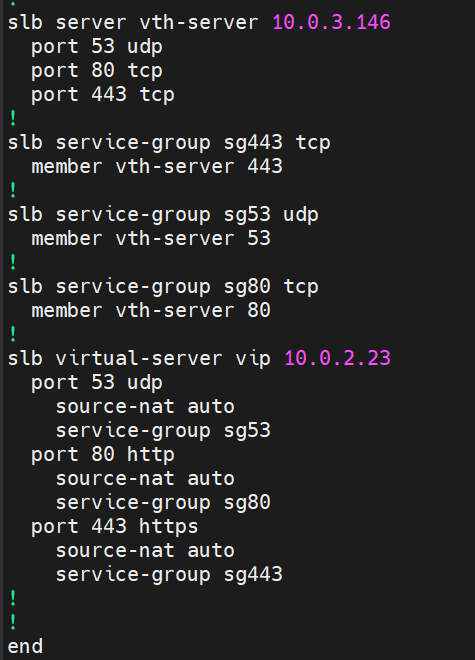


### 

### Slb verification

Execute Command -> *show running-config*

*vThunder 1(stack name-inst-1)*

**

### SSL verification

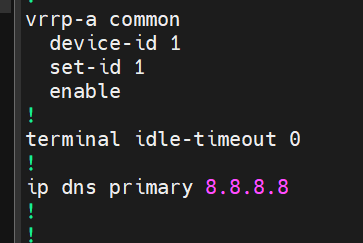
Execute Command -> *show pki cert*

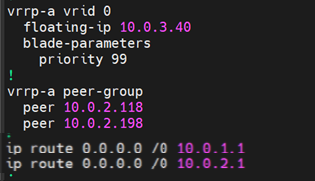


### HA verification

Execute Command -> *show running-config*

*vThunder 1(stack name-inst-1)*

**

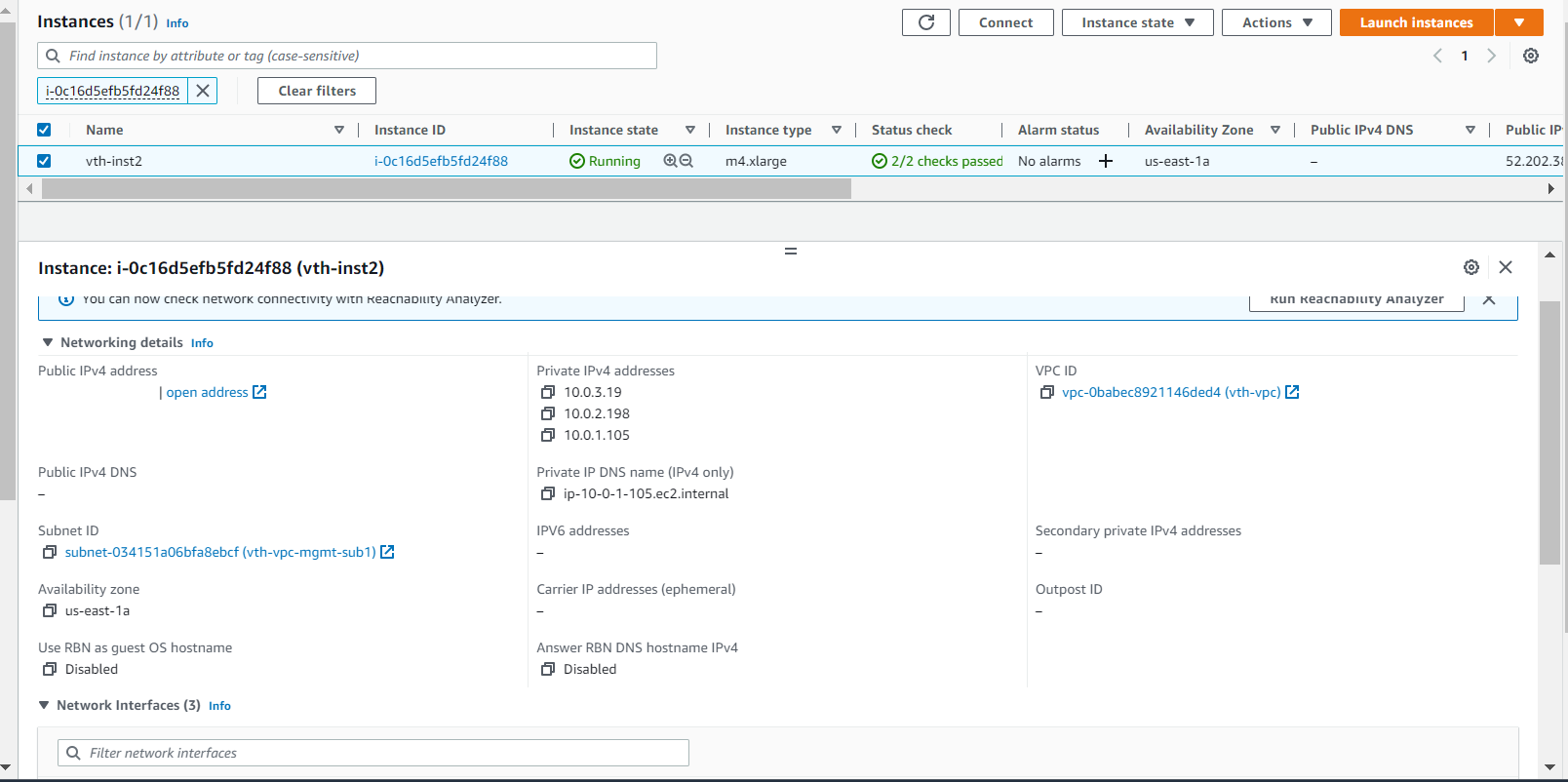
**

Path: *EC2>> Instances>> vth-inst1>> Networking>>* *vth-inst1*

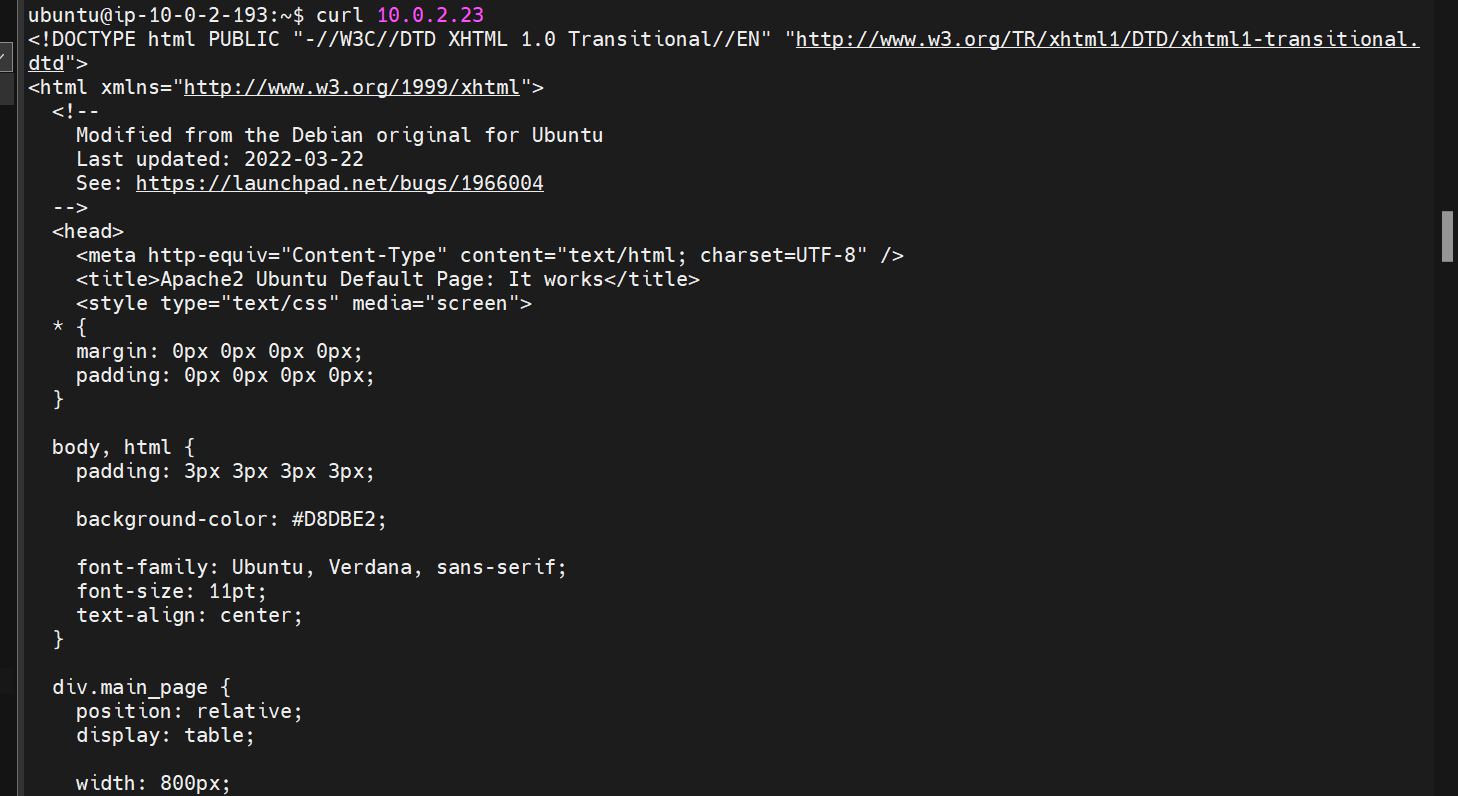
vth-inst1



vth-inst2



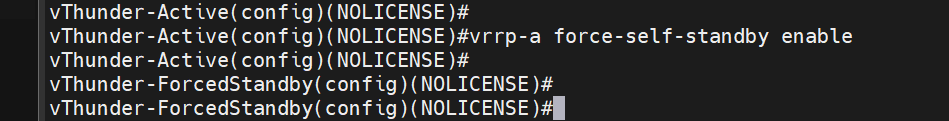
#### Traffic testing



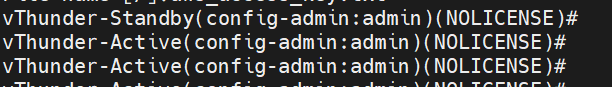
Run the following command to force stop the active vThunder and make standby to active.

*vrrp-a force-self-standby enable*

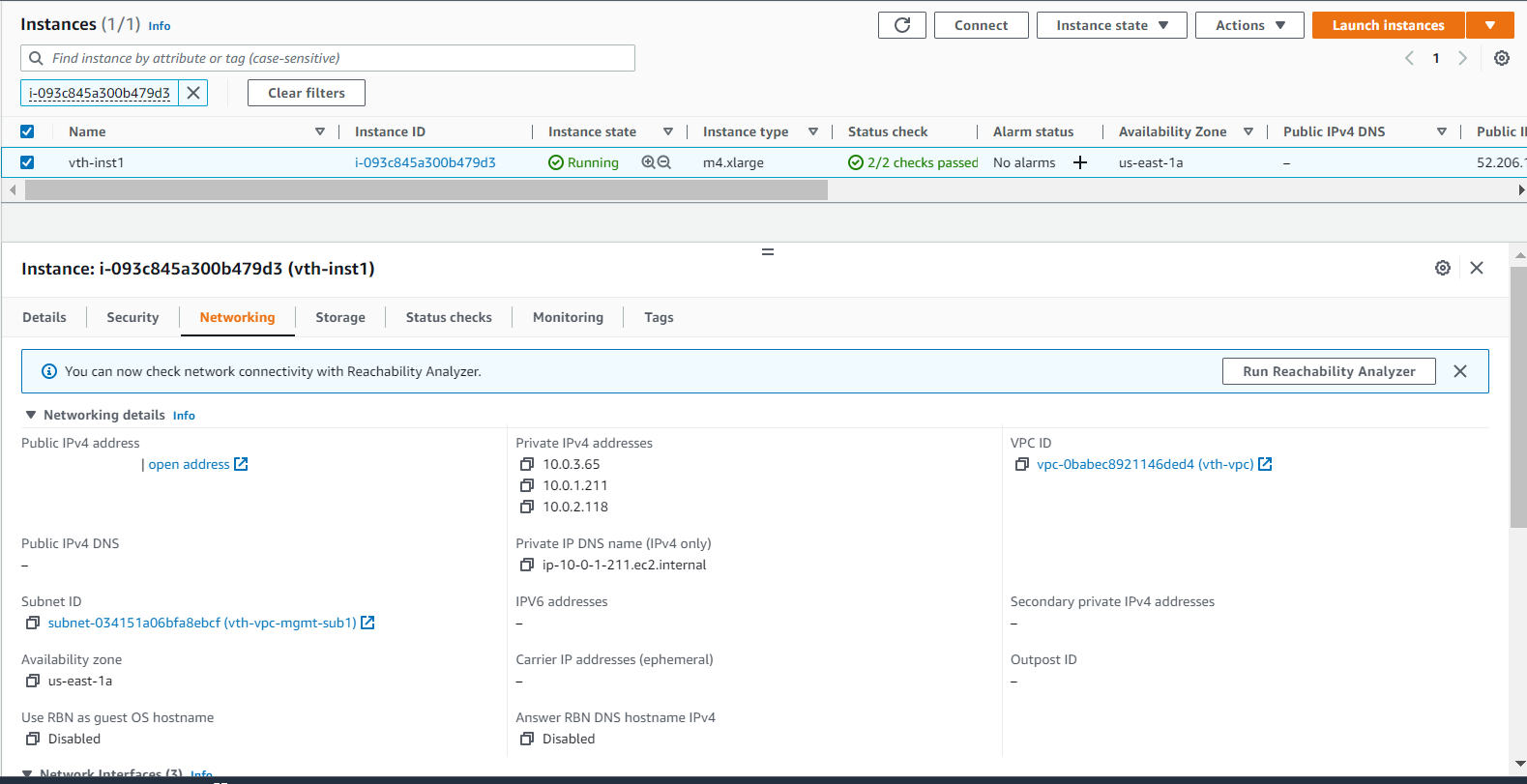
vth-inst1



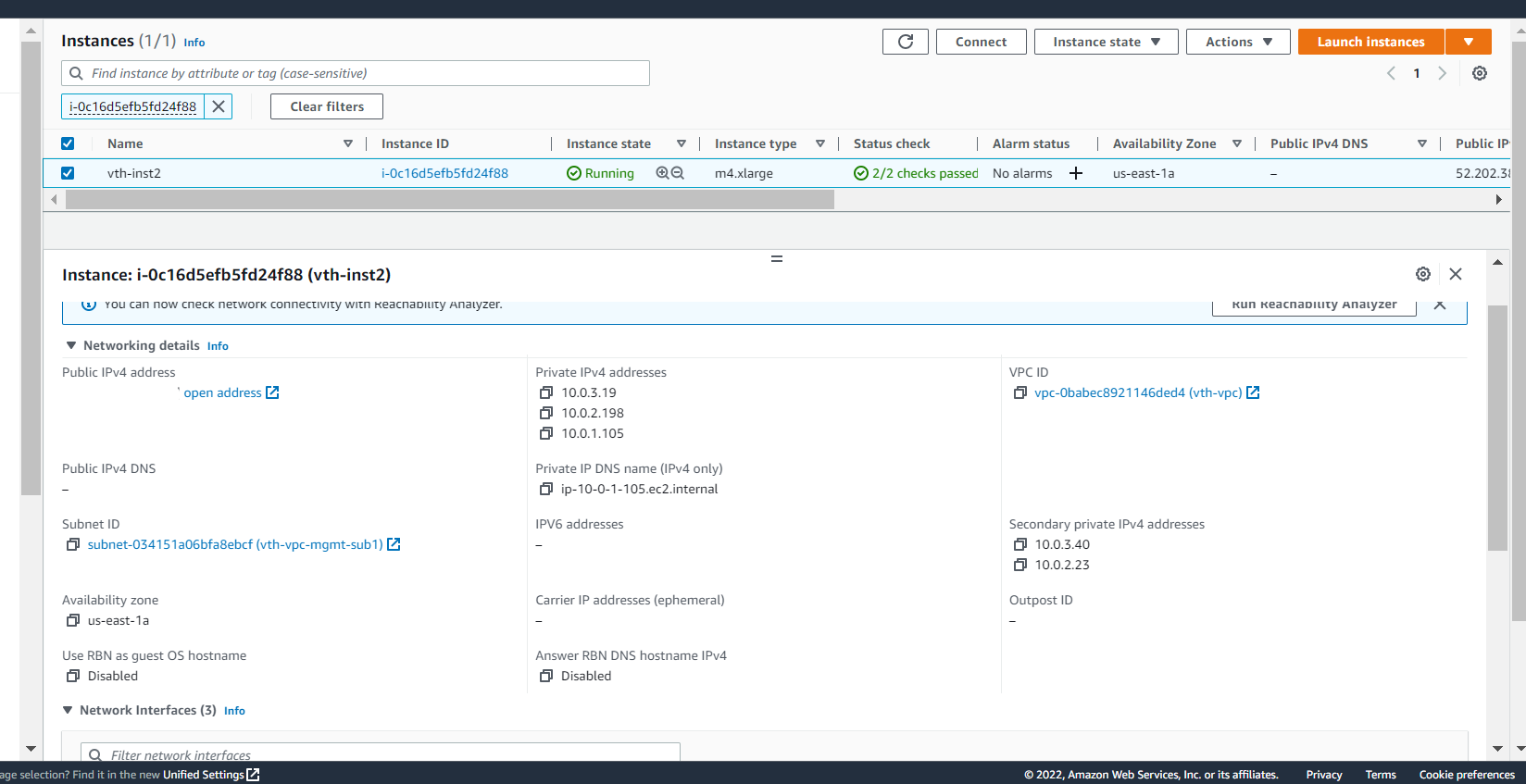
vth-inst1



vth-inst1



vth-inst2



## Traffic Flow from Client to Server via vThunder

Login to Client machine.

Run command : curl <vThunder Secondary Private IP>

