



Installing vThunder ADC using AWS CFT Templates

v1.1.0

March, 2023

© 2023 A10 Networks, Inc. All rights reserved.

Information in this document is subject to change without notice.

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.

Table of Contents

Introduction	8
AWS Terminology	11
Prerequisites	12
Image Repository	14
Configure Operating System	14
Windows	14
Linux, macOS, Unix	16
CloudFormation Templates	17
Deploy CFT A10-vThunder_ADC-2NIC-1VM	19
System Requirements	20
Supported Instance Types	21
Create vThunder Instance	23
Access vThunder using GUI or CLI	26
Access vThunder using GUI	26
Access vThunder using CLI	28
Configure Server and Client Machine	29
Configure a Server Machine	29
Configure a Client Machine	29
Configure vThunder as an SLB	30
Initial Setup	31
Deploy vThunder as an SLB	34
Verify Deployment	35
Verify Traffic Flow	37
Deploy CFT A10-vThunder_ADC-2NIC-1VM-GLM	39
System Requirements	40
Supported Instance Types	41
Create vThunder Instance	43
Access vThunder using GUI or CLI	46

Access vThunder using GUI	46
Access vThunder using CLI	48
Configure Server and Client Machine	49
Configure a Server Machine	49
Configure a Client Machine	49
Configure vThunder as an SLB	50
Initial Setup	51
Deploy vThunder as an SLB	54
Verify Deployment	56
Verify GLM	58
Verify License using GUI	58
Verify License using CLI	58
Verify Traffic Flow	60
Deploy CFT A10-vThunder_ADC-3NIC-2VM-HA	62
System Requirements	63
Supported Instance Types	64
Create vThunder Instances	67
Access vThunder using GUI or CLI	69
Access vThunder using GUI	69
Access vThunder using CLI	71
Configure Server and Client Machine	72
Configure a Server Machine	72
Configure a Client Machine	73
Configure FTP server	74
Configure vThunder as an SLB with HA	74
Initial Setup	75
Deploy vThunder as an SLB	79
Verify Access Keys on vThunder Instances	81
Verify Deployment	82
Verify Traffic Flow	89

Deploy CFT A10-vThunder_ADC-3NIC-2VM-HA-GLM-PVTVIP	91
System Requirements	92
Supported Instance Types	93
Create vThunder Instances	96
Access vThunder using GUI or CLI	98
Access vThunder using GUI	98
Access vThunder using CLI	100
Configure Server and Client Machine	101
Configure a Server Machine	101
Configure a Client Machine	102
Configure FTP server	103
Configure vThunder as an SLB with HA	103
Initial Setup	104
Deploy vThunder as an SLB	109
Verify Access Keys on vThunder Instances	111
Verify Deployment	112
Verify GLM	119
Verify License using GUI	119
Verify License using CLI	119
Verify Traffic Flow	121
Deploy CFT A10-vThunder_ADC-3NIC-2VM-HA-GLM-PUBVIP-BACKAUTO	123
System Requirements	124
Supported Instance Types	126
Create S3 Bucket	128
Create vThunder Instances	129
Access vThunder using GUI or CLI	133
Access vThunder using GUI	133
Access vThunder using CLI	135
Configure Server and Client Machine	136
Configure a Server Machine	136

Configure a Client Machine	136
Configure FTP server	137
Configure vThunder as an SLB with HA	138
Initial Setup	138
Deploy vThunder as an SLB	143
Verify Access Keys on vThunder Instances	146
Verify Deployment	146
Verify GLM	153
Verify License using GUI	153
Verify License using CLI	154
Verify Traffic Flow	155
Secondary Private IPv4 Address	155
Allocated Public IPv4 address	157
On-demand Password Change	157
Deploy CFT A10-vThunder_ADC-3NIC-6VM-2RG-GSLB	160
System Requirements	161
Supported Instance Types	165
Create vThunder Instances	168
vThunder instances for Region1	168
vThunder instances for Region2	171
Access vThunder using GUI or CLI	175
Access vThunder using GUI	175
Access vThunder using CLI	176
Configure vThunder as an SLB	178
Initial Setup	178
Deploy vThunder as an SLB	187
Verify Deployment	190
Verify Traffic Flow	208
DNS Lookup	209
WGET	212

Appendix 214

 Security Policy for AWS User 214

 Predefined 214

 Custom 214

 Default Password Policy 217

Introduction

vThunder is a fully operational, software-based Application Delivery Controller (ADC) solution that can run on Amazon Web Services (AWS) cloud. vThunder provides a robust, flexible, and easy-to-deploy application delivery and server load balancing service.

ACOS uses the CloudFormation Templates (CFT) to quickly deploy the vThunder instance on the AWS cloud. [Table 1](#) lists the available AWS CFT for deploying vThunder ADC on AWS cloud:

Table 1 : Available AWS CloudFormation Templates

Template	Description	Configuration
A10-vThunder_ ADC-2NIC-1VM	• Creates one vThunder instance with two Network Interface Cards (NICs). • Deploys a Certificate Authority SSL Certificate and Server Load Balancer (SLB).	• 2 NICs (1 Management + 1 Data) • BYOL (Bring Your Own License) • 1 VM (vThunder Virtual Instance) • SLB (vThunder Server Load Balancer) • SSL (Apply SSL Certificate)
A10-vThunder_ ADC-2NIC-1VM- GLM	• Creates one vThunder instance with two Network Interface Cards and A10 Global License Manager (GLM) integration. • Deploys a Certificate Authority SSL Certificate and Server Load Balancer.	• 2 NICs (1 Management + 1 Data) • BYOL (Bring Your Own License) • 1 VM (vThunder Virtual Instance) • SLB (vThunder Server Load Balancer) • SSL (Apply SSL Certificate) • GLM (Auto apply A10

Template	Description	Configuration
		license)
A10-vThunder_ ADC-3NIC-2VM- HA	- Creates two vThunder instances with High Availability (HA) setup, each vThunder contains three Network Interface Cards. - Deploys a Certificate Authority SSL Certificate and Server Load Balancer.	3 NICs (1 Management + 2 Data) - BYOL (Bring Your Own License) 2 VMs (vThunder Virtual Instances) - SLB (vThunder Server Load Balancer) - SSL (Apply SSL Certificate) - HA (High Availability with auto switchover with next available vThunder VM using VRRP)
A10-vThunder_ ADC-3NIC-2VM- HA-GLM-PVTVIP	- Creates two vThunder instances with High Availability setup and an A10 Global License Manager integration, each vThunder has three Network Interface Cards. - Deploys a Certificate Authority SSL Certificate, and a Server Load Balancer.	3 NICs (1 Management + 2 Data) - BYOL (Bring Your Own License) 2 VMs (vThunder Virtual Instances) - SLB (vThunder Server Load Balancer) - SSL (Apply SSL Certificate) - GLM (Auto apply A10 license) - HA (High Availability with auto switchover with available VM using VRRP) - VIP (Private Interface)

Template	Description	Configuration
A10-vThunder_ ADC-3NIC-2VM- HA-GLM- PUBVIP- BACKAUTO	- Creates two vThunder instances with High Availability (HA) setup and GLM integration, each vThunder contains three Network Interface Cards. - Deploys a Certificate Authority SSL Certificate, Server Load Balancer, and backend server autoscaling support.	3 NICs (1 Management + 2 Data) - BYOL (Bring Your Own License) 2 VMs (vThunder Virtual Instances) - SLB (vThunder Server Load Balancer) - SSL (Apply SSL Certificate) - GLM (Auto apply A10 license) - HA (High Availability with auto switchover for the available VM using VRRP) - VIP (Public Interface) - BACKAUTO (Lambda function to apply SLB config into vThunder for newly added/deleted web/app servers via autoscaling group)
A10-vThunder_ ADC-3NIC-6VM- 2RG-GSLB	- Creates two Global Server Load Balancer (GSLB) regions, one GSLB controller and two site devices in each of the two regions. - Creates two real servers (Ubuntu 16.04.0-LTS) in each region.	3 NICs (1 Management + 2 Data) - BYOL (Bring Your Own License) 6 VMs (Three vThunder Virtual Instances in each Region) 2 RGs (Regions) - GSLB (vThunder SLB in

Template	Description	Configuration
		multiple regions)

This documentation helps you to deploy the vThunder instance/s on the AWS cloud after completion of following tasks:

- Downloading the required template from GitHub on your local machine
- Configuring the vThunder installation parameters in the template
- Executing required commands in the AWS terminal window.

AWS Terminology

The following is a glossary of AWS terms commonly used in this document:

- **Access control list (ACL)** — A firewall or a security layer on the subnet level. For more information, see <https://docs.aws.amazon.com/AmazonS3/latest/userguide/acls.html>
- **CloudWatch** — A service that allows you to monitor various elements of your AWS account. For more information, see <https://docs.aws.amazon.com/cloudwatch/index.html>
- **Lambda** — A serverless computing that will replace Elastic Compute Cloud (EC2) instances, for most of the functionality of EC2. For more information, see <https://docs.aws.amazon.com/lambda/latest/dg/welcome.html>
- **Security group (SG)** — The firewall or security layer on the server or instance level. For more information, see <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-security-groups.html>.
- **Subnet** — A subsection of a network that generally includes all the computers in a specific location. For more information, see <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, see <https://docs.aws.amazon.com/vpc/latest/userguide/what-is-amazon-vpc.html>

Prerequisites

To deploy vThunder on AWS cloud using any of the available CFT, you must ensure the following prerequisites are met:

- AWS account with a valid subscription and access keys (Required)
 - Create the access keys (*access key ID* and *secret access key*) if you don't have them already. For more information, see <https://docs.aws.amazon.com/powershell/latest/userguide/pstools-appendix-sign-up.html>
 - Create an SSH key from **AWS Management Console** > **EC2 Formation** > **Key Pairs** > **Create key pair** with the following:
 - **Name:** <your key name>
 - **Key pair type:** RSA
 - **Private key file format:** .pemSave this SSH key for future use. For more information, see <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/create-key-pairs.html>
 - Download the following AWS tool to create and manage resources:
 - [AWS Management Console](#) — A web console to create and monitor AWS resources.
 - **AWS Command Line Interface (CLI)** — An interface that can be launched to start a CLI session. The interface can be launched using one of the following:
 - **Linux shells** — Use programs such as [bash](#), [zsh](#), and [tcsh](#) to run commands in Linux or macOS.
 - **Windows command line** — Use Windows command prompt or PowerShell to run commands in Windows.

- **Remotely** — Use Amazon Elastic Compute Cloud (Amazon EC2) instances through a remote terminal program such as MobaXterm, [PuTTY](#), [SSH](#), or with AWS Systems Manager to run commands in AWS. For more information, see http://docs.aws.amazon.com/cli/index.html?nc2=h_ql_doc_cli

- AWS User

A user with predefined and custom policies applied. For more information, see [Security Policy for AWS User](#).

- Configure the following parameters if you are deploying Backend server autoscaling CFT:

- Create AWS Secrets Manager secret from **AWS Management Console > Secrets Manager > Store a new secret** with the following:

- **Secret Type:** Other type of secret

- **Key/value pairs:** Provide the following values:

Table 2 : Key value pair

Key	Value
aws_access_key_id	<i>your aws access key id</i>
aws_secret_access_key	<i>your aws secret access key</i>

- **Encryption key:** aws/secretsmanager

- **Secret name:** <your AWS keys secret manager name>

For more information, see

https://docs.aws.amazon.com/secretsmanager/latest/userguide/create_secret.html

- Lambda Function Execution Role ARN

This is required for the Autoscaling template. For more information, see

<https://docs.aws.amazon.com/lambda/latest/dg/lambda-intro-execution-role.html#permissions-executionrole-console>

- Valid [SSL certificate](#) to apply on vThunder (Optional).
- Python [[3.8.5](#) and later] with [PIP](#).

- Text editor (Notepad++, Notepad or any other text editor application).
- [A10 GLM account](#) access with valid licenses.

This access is required for the templates using GLM. For more information, see [Global License Manager User Guide](#).

- CloudFormation Template (CFT)

Go to [GitHub](#) [Branch: release/v1.1.0] and download the required CFT folder to your local machine. For example, C:\Users\TestUser\Templates. The template folder contains the json parameter files and Python scripts for the deployment of the respective template.

- A10 vThunder default user credentials

Send a request to [A10 Networks Support](#) for A10 vThunder login default user credentials.

Image Repository

CFT templates are tested with the following A10 vThunder images:

- [64-bit Advanced Core OS \(ACOS\) version 5.2.1-P6, build 74](#)
- 64-bit Advanced Core OS (ACOS) version 5.2.1-P5, build 113

Configure Operating System

Before deploying CFT, perform the following steps depending on your operating system:

- [Windows](#)
- [Linux, macOS, Unix](#)

Windows

If you are deploying the CFT using Windows, perform the following steps:

1. Verify if the recommended Python version is installed correctly.

Launch the command prompt from Windows search, and enter the following command:

```
C:\Users\TestUser> python
```

If Python is installed, the version details are displayed. Ensure that the version is 3.8.5 or higher.

2. Verify if PIP is installed.

In the command prompt, enter the following command:

```
C:\Users\TestUser> pip
```

If PIP is installed, the pip command usage, commands and other general options are displayed.

3. Install all Python dependencies.

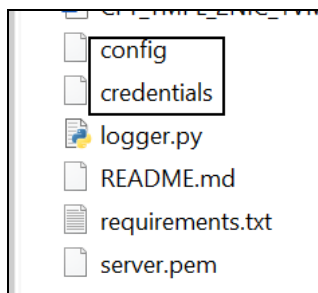
- a. From the command prompt, navigate to the downloaded CFT folder path and enter the following command:

```
C:\Users\TestUser\Templates> pip install -r requirements.txt
```

A .aws folder is automatically created under C:\Users\TestUser.

- b. From the downloaded CFT folder, locate and open the **credentials** file with a text editor.

Figure 1 : CFT folder



- c. Update the access key ID and secret access key as per your AWS account and then save the changes.

```
[default]
aws_access_key_id = your_aws_access_key_id
aws_secret_access_key = your_aws_secret_access_key
```

- d. Copy this file to the C:\Users\TestUser\.aws folder.
- e. From the downloaded CFT template folder, locate and open the **config** file with a text editor.
- f. Update region with the your working region and then save the changes.

```
[default]
region = you_working_region
output = json
```

- g. Copy this file to the C:\Users\TestUser\.aws folder.

Linux, macOS, Unix

If you are deploying the CFT using either Linux, macOS, or Unix, perform the following steps:

1. Verify if the recommended Python version is installed.

Open the Shell terminal, enter the following command:

```
testuser@localhost:~$ python
```

If Python is installed, the version details are displayed.

2. Verify if PIP is installed.

In the command prompt, enter the following command:

```
testuser@localhost:~$ pip
```

If PIP is installed, the pip command usage, available commands, and general options are displayed.

3. Install all Python dependencies.

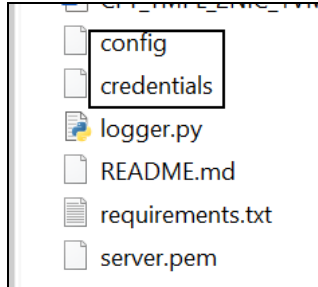
- a. From the command prompt, navigate to the downloaded CFT folder path and enter the following command:

```
testuser@localhost:~$ pip install -r requirements.txt
```

A ~/.aws folder is auto created under /home/username.

- b. From the downloaded CFT folder, locate and open the **credentials** file with a text editor.

Figure 2 : CFT folder



- c. Update the access key and secret access key as per your AWS account and then save the changes.

```
[default]
aws_access_key_id = your_aws_access_key_id
aws_secret_access_key = your_aws_secret_access_key
```

- d. Copy this file to the ~/.aws folder.
- e. From the downloaded CFT folder, locate and open **credentials** file with a text editor.
- f. Update region with logged-in-user region and then save the changes.

```
[default]
region = logged_in_user_region
output = json
```

- g. Copy this file to the ~/.aws folder.

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

CloudFormation Templates

To implement infrastructure as a code for your AWS solution, use CFT. The template is a json native file that defines the infrastructure and configuration for your project. The template uses declarative syntax to specify the resources that are to be deployed

and the properties for those resources without having to write the sequence of programming commands to create it.

The following templates are available:

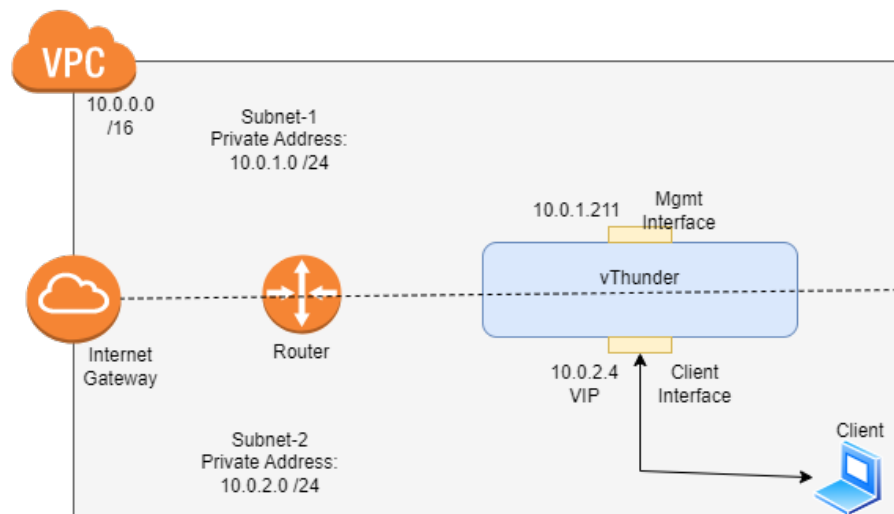
- [Deploy CFT A10-vThunder_ADC-2NIC-1VM](#)
- [Deploy CFT A10-vThunder_ADC-2NIC-1VM-GLM](#)
- [Deploy CFT A10-vThunder_ADC-3NIC-2VM-HA](#)
- [Deploy CFT A10-vThunder_ADC-3NIC-2VM-HA-GLM-PVTVIP](#)
- [Deploy CFT A10-vThunder_ADC-3NIC-2VM-HA-GLM-PUBVIP-BACKAUTO](#)
- [Deploy CFT A10-vThunder_ADC-3NIC-6VM-2RG-GSLB](#)

Deploy CFT A10-vThunder_ADC-2NIC-1VM

Using this template, you can deploy one vThunder instance containing one management interface and one data interface.

[Figure 3](#) illustrates the deployment topology for the 2NIC-1VM template.

Figure 3 : 2NIC-1VM Deployment Topology



The following topics are covered:

System Requirements	20
Supported Instance Types	21
Create vThunder Instance	23
Access vThunder using GUI or CLI	26
Configure Server and Client Machine	29
Configure vThunder as an SLB	30
Verify Deployment	35
Verify Traffic Flow	37

System Requirements

When the files from the CFT are downloaded and saved on your local machine, the resources and their default values are displayed as shown in [Table 3](#). You can configure the resources and modify these default values based on your deployment requirements.

Table 3 : System Requirements

Resource Name	Description	Default Value
Stack	A new stack is created with the specified name and location.	In this template, <code>vth</code> is used as the default stack name. Therefore, all the resources has <code>vth</code> as the prefix.
Network Interface Card [NIC]	Two types of interfaces are created for a vThunder instance: - One management interface - One data interface	<code>vth-inst1-mgmt-nic1</code> <code>vth-inst1-data-nic1</code>
Subnet	Two subnets are created with an address prefix each.	<code>vth-vpc-mgmt-subnet1</code> <code>vth-vpc-data-subnet1</code>
Virtual Private Network [VCN]	A virtual private network is assigned to the virtual machine instance.	<code>vth-vpc</code> Address prefix for virtual network: 10.0.0.0/16
Elastic Public IP	One Elastic Public IP address is created and attached to the management interface of the vThunder instance.	<code>vth-inst1-mgmt-nic1-ip</code>
Security Group	One security group is created for the management interface. One security group is created for the data interface.	Here, the tag names are: <code>vth-sg-mgmt</code> <code>vth-sg-data</code>

Resource Name	Description	Default Value
	Logical name: - vThunderSecurityGroupMgmt - vThunderSecurityGroupData	
vThunder Instance	A vThunder EC2 instance is created. Default type: m4.xlarge (40 Gb memory) Table 4 lists the supported instance types.	vth-inst1
Ubuntu Server Instance	One Ubuntu Server instance is created. Default size: t2.micro	vth-server

Supported Instance Types

[Table 4](#) provides detailed information about the supported instance types.

Table 4 : List of Supported Instance Types

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

Instance	vCPU	Memory	Number of Network Interfaces
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

Instance	vCPU	Memory	Number of Network Interfaces
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

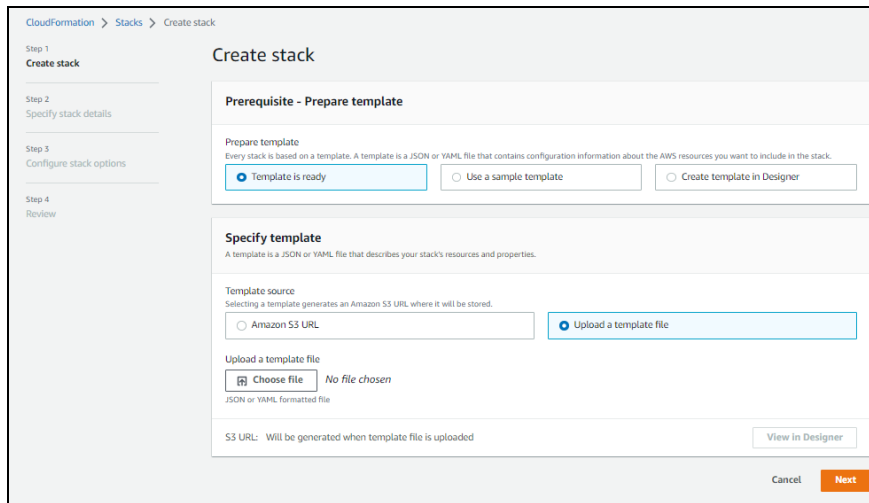
Create vThunder Instance

To create a vThunder instance, perform the following steps:

1. From AWS Management Console, navigate to **CloudFormation > Stacks > Create Stack > With new resources (standard)**.

The Create stack window is displayed.

Figure 4 : Create stack window



2. In the **Prerequisite - Prepare template** section, select **Template is ready**.

After the selecting this option, the Specify template section is displayed.

3. In the **Specify template** section, select **Upload a template file** and click **Choose file** to browse and upload the following template file from the downloaded CFT folder:

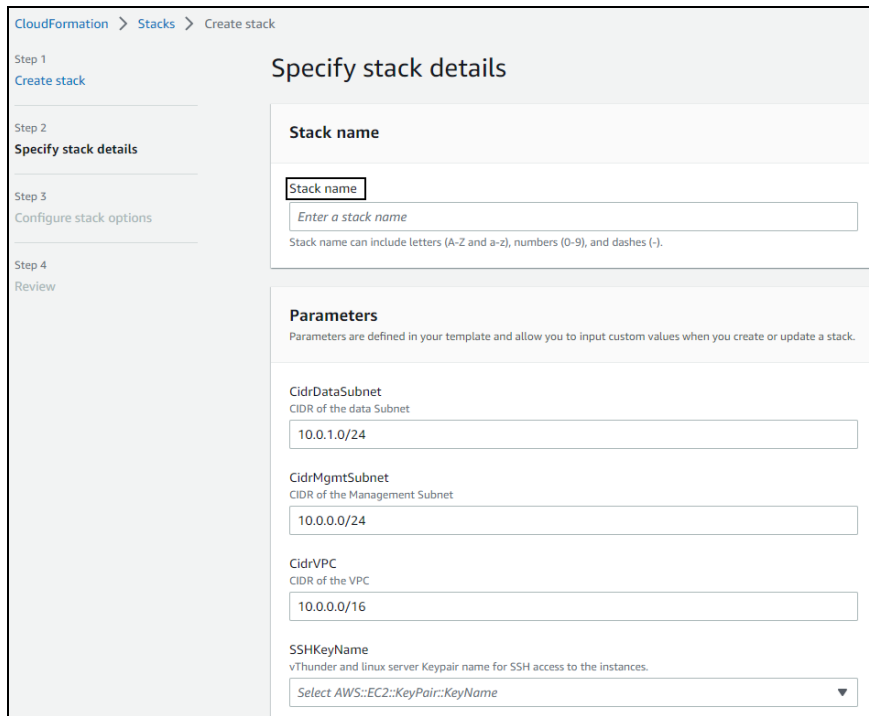
CFT_TMPL_2NIC_1VM_1.json

The selected template file name is displayed as the chosen file.

4. Click **Next**.

The **Specify stack details** window is displayed.

Figure 5 : Specify stack details window



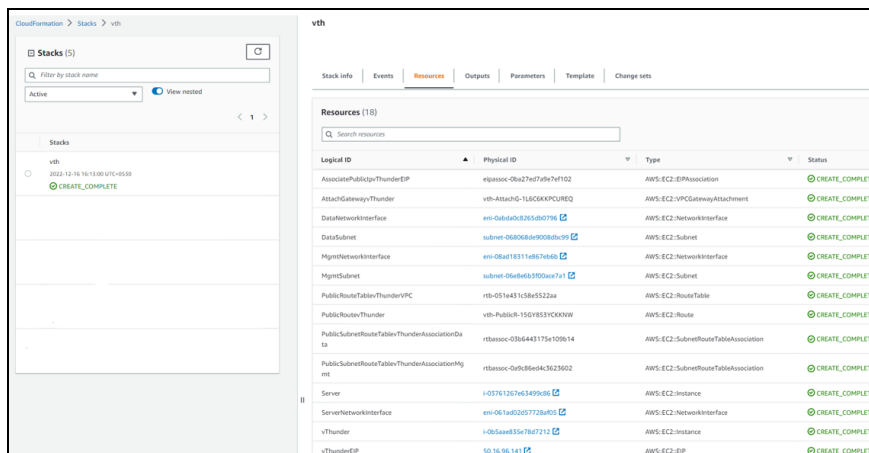
5. In the **Specify stack details** window, enter or select the following:
 - a. In the **Stack name** section, enter a **Stack name**.
Here, **vth** is provided as the default stack name.
 - b. In the **Parameters** section, enter or select the required values in the following fields:
 - **KeyPairName:** *<your SSH key>*
 - **TagValue:** **a10-vthunder-adc**
 - **Zone:** *<your availability zone>*
 - c. Verify the other fields and change the values appropriately. (Optional)
6. Click **Next**.
The **Configure stack options** window is displayed.
7. Verify the fields and change the values appropriately. (Optional)
8. Click **Next**.
The **Review** window is displayed.

9. Verify if all the stack configurations are correct and then click **Submit**.

NOTE: The system may take a few minutes to create the resources and display the stack status as **CREATE_COMPLETE**.

10. Verify if all the above resources are created in the **AWS Management Console > CloudFormation > Stacks > <stack_name> > Resources** tab.

Figure 6 : Resource listing in the resource group



Logical ID	Physical ID	Type	Status
AssociatePublicIpThunderEIP	eipassoc-0ba27e7fde7ef102	AWS::EC2::EIPAssociation	CREATE_COMPLETE
AttachGatewayThunder	vth-AttachG-1LEGGKPCUREQ	AWS::EC2::VPCGatewayAttachment	CREATE_COMPLETE
DataNetworkInterface	eni-04bda08265d0796	AWS::EC2::NetworkInterface	CREATE_COMPLETE
DataSubnet	subnet-06008d8d900d8c09	AWS::EC2::Subnet	CREATE_COMPLETE
MgmtNetworkInterface	eni-0ba18311d857d6b6	AWS::EC2::NetworkInterface	CREATE_COMPLETE
MgmtSubnet	subnet-06d6b03f00ac7a1	AWS::EC2::Subnet	CREATE_COMPLETE
PublicRouteTableThunderVPC	rtb-051e431c58d522aa	AWS::EC2::RouteTable	CREATE_COMPLETE
PublicRouteThunder	vth-PublicR-1SGY853YCKKNW	AWS::EC2::Route	CREATE_COMPLETE
PublicSubnetRouteTableThunderAssociationDe	rtbassoc-036d443175e109b14	AWS::EC2::SubnetRouteTableAssociation	CREATE_COMPLETE
PublicSubnetRouteTableThunderAssociationMg	rtbassoc-04f8d6e43623602	AWS::EC2::SubnetRouteTableAssociation	CREATE_COMPLETE
Server	i-0376126746349a86	AWS::EC2::Instance	CREATE_COMPLETE
ServerNetworkInterface	eni-061aa02d07728a05	AWS::EC2::NetworkInterface	CREATE_COMPLETE
vThunder	i-065aa8835e78d7212	AWS::EC2::Instance	CREATE_COMPLETE
vThunderEIP	elb-16-96-141	AWS::EC2::EIP	CREATE_COMPLETE

The vThunder instance is listed under **Resources** tab.

Access vThunder using GUI or CLI

vThunder instance can be accessed using any of the following ways:

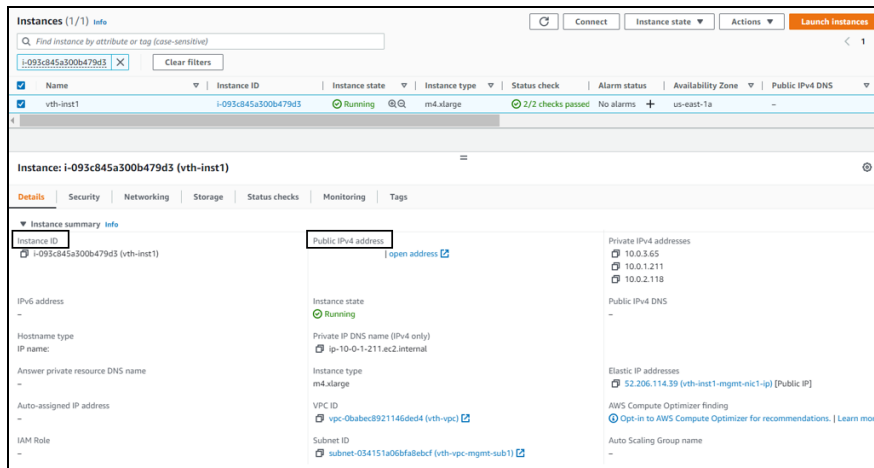
- [Access vThunder using GUI](#)
- [Access vThunder using CLI](#)

Access vThunder using GUI

To access vThunder instance using GUI, perform the following steps:

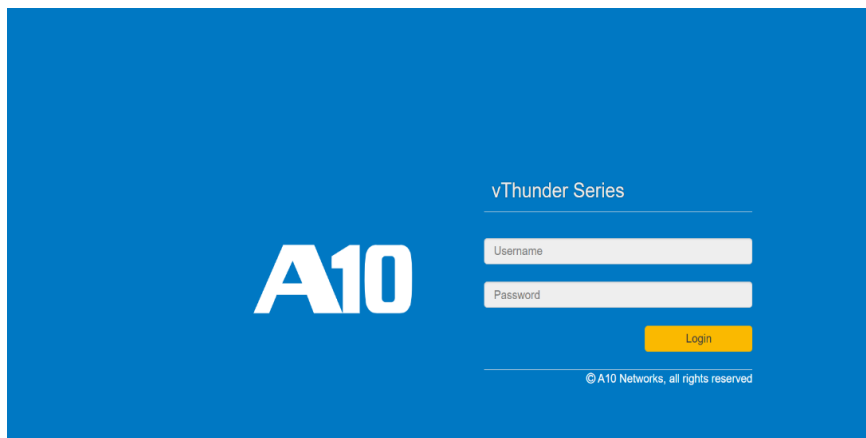
1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select your instance name.
Here, **vth-inst1** is the vThunder instance.

Figure 7 : vThunder instance



3. Copy the **Public IPv4 address** from the **Details** tab and replace the IP address in the below link:
http://<vThunder_public_IPv4_address>
4. Open the updated link in any browser.
 The vThunder login window is displayed.

Figure 8 : vThunder GUI



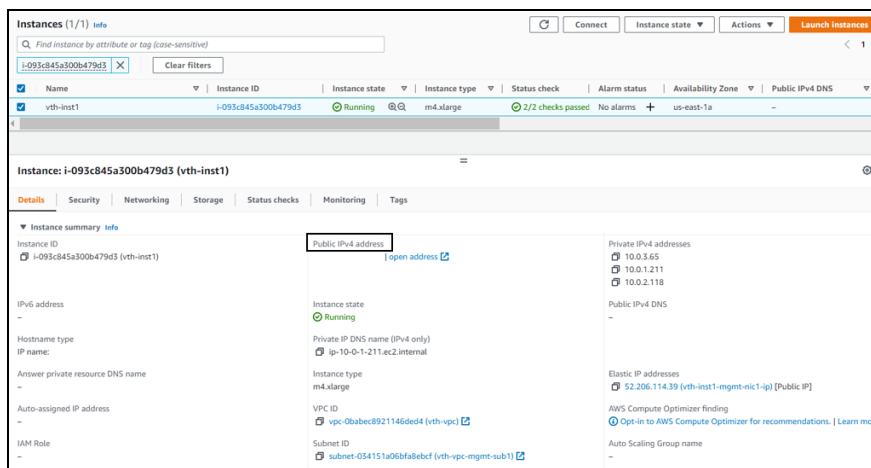
5. Enter the following credentials and click **Login**:
 - Username – **admin**
 - Password – **EC2 Instance ID**
 The home page is displayed if the entered credentials are correct.

Access vThunder using CLI

To access vThunder instance using CLI, perform the following steps:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select your instance name.
Here, **vth-inst1** is the vThunder instance.

Figure 9 : vThunder instance



3. Copy the **Public IPv4 address** from the **Details** tab.
4. Open any SSH client and provide the following to establish a connection:
 - Hostname: Public IPv4 address
 - Username: admin
 - Key: SSH Key
5. Connect to the session.
6. In the SSH client session, run the following commands:

```
vThunder (NOLICENSE) >enable <---Execute command--->
Password: <---just press Enter key--->
vThunder (NOLICENSE) #config <---Configuration mode--->
vThunder (config) (NOLICENSE) #
```

The vThunder instance is ready to use.

Configure Server and Client Machine

To test the traffic flow via vThunder, create and configure a server machine and a client machine:

- [Configure a Server Machine](#)
- [Configure a Client Machine](#)

Configure a Server Machine

To configure a server machine, perform the following steps:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select your server instance name.
Here, **vth-server** is the server instance name.
3. Click **Connect**.
A **Connect to instance** window with **EC2 Instance Connect** tab is displayed.
4. Click **Connect**.
A **Terminal** window is displayed.
5. Run the following commands in the Terminal window to update all the package information:

```
sudo apt update
```

6. Run the following command in the Terminal window to create an Apache Server virtual machine:

```
sudo apt install apache2
```

While the Apache server is getting installed, you get a prompt to continue further. Enter 'Y' to continue. After the installation is complete, a newline prompt is displayed.

Configure a Client Machine

To configure a client machine, perform the following steps:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Click **Launch Instances**.
A **Launch an instance** window is displayed.
3. In the **Name and tags** section, enter an instance name.
Here, enter `vth-client` as the client instance name.
4. In the **Application and OS Images** section, select **Ubuntu**.
5. In the **Instance type** section, select the required instance type.
6. In the **Key pair (login)** field, select your SSH key.
7. In the **Network settings** section, click **Edit** to edit the following:
 - VPC: *your VPC*
Here, `vth-vpc` is the VPC.
 - Subnet: Data subnet
Here, `10.0.1.0/24` is the data subnet value.
 - Auto-assign public IP: Enable
 - Firewall (security groups): Select existing security group
 - Common security groups: *your data security group*
Here, `vth-vThunderSecurityGroupData` is the security group.
8. Click **Launch instance**.

NOTE: The system may take a few minutes to launch the instance.

The client instance is displayed in the **Instances** list with the status as **Running**.

Configure vThunder as an SLB

The following topics are covered:

- [Initial Setup](#)
- [Deploy vThunder as an SLB](#)

Initial Setup

Before deploying vThunder on AWS cloud as an SLB, you need to configure the corresponding parameters in the CFT.

To configure the parameters, perform the following steps:

1. Navigate to the folder where you have downloaded the CFT, and open the CFT_TMPL_2NIC_1VM_CONFIG_SLB_SSL_PARAM.json with a text editor.

NOTE: Each parameter has a default value mentioned in the parameter file.

2. Configure the stack name created using CFT.

```
"stackDetails": {
  "value": [
    {
      "stackName": "vth"
    }
  ]
},
```

3. Configure SLB server ports.

```
"server-list": {
  "value": [
    {
      "port-list": [
        {
          "port-number": 53,
          "protocol": "udp"
        },
        {
          "port-number": 80,
          "protocol": "tcp"
        },
        {
          "port-number": 443,
          "protocol": "tcp"
        }
      ]
    }
  ]
}
```

```
    }  
  ]  
},
```

4. Configure Service Group List ports.

The Service group name by default is "sg+port_number". If you may want to change the service group name then after changing the name, change the names in the Virtual servers as well.

```
"serviceGroupList": {  
  "value": [  
    {  
      "name": "sg443",  
      "protocol": "tcp",  
      "member-list": [  
        {  
          "port": 443  
        }  
      ]  
    },  
    {  
      "name": "sg53",  
      "protocol": "udp",  
      "member-list": [  
        {  
          "port": 53  
        }  
      ]  
    },  
    {  
      "name": "sg80",  
      "protocol": "tcp",  
      "member-list": [  
        {  
          "port": 80  
        }  
      ]  
    }  
  ]  
}
```



```
]
},
```

5. Configure a Virtual Server.

The virtual server default name is “vip”. It is the Private IP address of Ethernet1.

```
"virtualServerList": {
  "virtual-server-name": "vip",
  "metadata": {
    "description": "virtual server is using ethernet 1 ip address"
  },
  "value": [
    {
      "port-number": 53,
      "protocol": "udp",
      "auto": 1,
      "service-group": "sg53"
    },
    {
      "port-number": 80,
      "protocol": "http",
      "auto": 1,
      "service-group": "sg80"
    },
    {
      "port-number": 443,
      "protocol": "https",
      "auto": 1,
      "service-group": "sg443"
    }
  ]
},
```

6. Configure SSL.

```
"sslConfig": {
  "requestTimeout": 40,
  "Path": "server.pem",
  "File": "server",
```

```
"CertificationType": "pem"
}
```

NOTE: By default, SSL configuration is disabled i.e., no SSL configuration is applied.

The server.pem file is included in the downloaded CFT folder. If necessary, you can edit this file or use a different certificate file. When you use a different certificate file, remember to change the path accordingly.

7. Verify if all the configurations in the CFT_TMPL_2NIC_1VM_CONFIG_SLB_SSL_PARAM.json file are correct and then save the changes.

Deploy vThunder as an SLB

To deploy vThunder on AWS cloud as an SLB, perform the following steps:

1. From your command prompt, navigate to the folder where you have downloaded the CFT.
2. Run the following command to create vThunder SLB instance:

```
python ./CFT_TMPL_2NIC_1VM_CONFIG_SLB_SSL_2.py
```

A message is prompted to upload the SSL certificate.

3. Enter **yes** if you want to upload the SSL certificate.

```
Do you want to upload ssl certificate(yes/no)?yes
```

A message is prompted that displays the criteria for a valid admin user password.

```
Primary conditions for password validation, user should provide the new password according to the given combination:
```

```
Minimum length of 9 characters
```

```
Minimum lowercase character should be 1
```

```
Minimum uppercase character should be 1
```

```
Minimum number should be 1
```

```
Minimum special character should be 1
```

```
Should not include repeated characters
Should not include more than 3 keyboard consecutive characters.
```

A message is prompted to enter a new password.

4. Provide the new password for the admin user and then confirm the new password.

The new password should follow the valid admin user password criteria mentioned in the above step. For more information, see [Default Password Policy](#).

```
Enter vThunder's new password:
Confirm new password:
```

If both the password and confirm password field values match and complies to the valid admin user password criteria, a message 'Password changed successfully' is displayed along with the other configurations.

```
Password changed successfully.
Password change configurations saved on partition: shared
-----
-----
configured ethernet ip
Configured server vth-server
Configured virtual servers
SSL Configured.
Configurations are saved on partition: shared
```

Verify Deployment

To verify vThunder SLB deployment using CFT, perform the following steps:

1. Access the vThunder instance using CLI.
2. Run the following command to verify the running configuration:

```
vThunder(config) (NOLICENSE) #show running-config
```

If the deployment is successful, the following SLB configuration is displayed:

```
!Current configuration: 0 bytes
!Configuration last updated at 17:36:35 GMT Wed Nov 30 2022
```

```
!Configuration last saved at 17:35:40 GMT Wed Nov 30 2022
!64-bit Advanced Core OS (ACOS) version 5.2.1, build 153 (Dec-11-
2020,14:34)
!
!
interface ethernet 1
    enable
    ip address dhcp
!
!
slb server vth-server 10.0.1.120
    port 53 udp
    port 80 tcp
    port 443 tcp
!
slb service-group sg443 tcp
    member vth-server 443
!
slb service-group sg53 udp
    member vth-server 53
!
slb service-group sg80 tcp
    member vth-server 80
!
slb virtual-server vip use-if-ip ethernet 1
    port 53 udp
        source-nat auto
        service-group sg53
    port 80 http
        source-nat auto
        service-group sg80
    port 443 https
        source-nat auto
        service-group sg443
!
!
end
```

3. Run the following command to verify the SSL configuration:

```
vThunder(config) (NOLICENSE) #show pki cert
```

If the deployment is successful, the following SSL configuration is displayed:

Name	Type	Expiration	Status
server certificate	Jan 28 12:00:00 2028 GMT	[Unexpired, Bound]	

NOTE:

If you want to access the vThunder instance using GUI, re-login the instance using the new admin user password and NOT the *EC2 Instance ID*.

Verify Traffic Flow

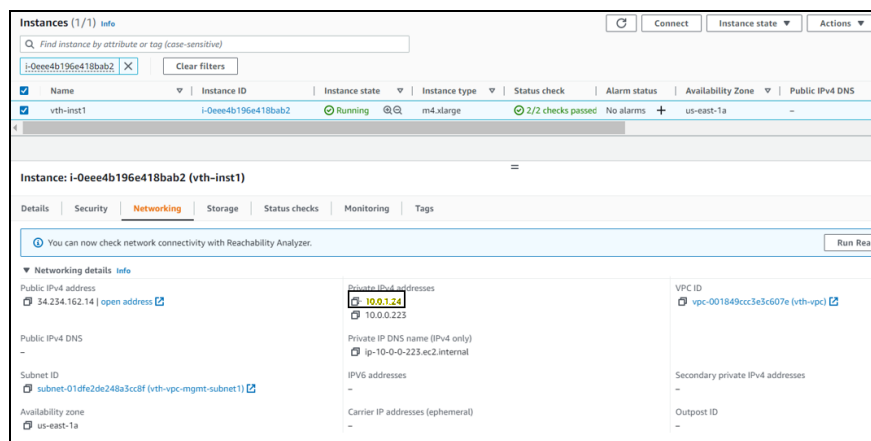
To verify the traffic flow from client machine to server machine via vThunder, perform the following:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select the vThunder instance name and then click the **Networking** tab.

Here, **vth-inst1** is the vThunder instance name.

3. Copy the IP address of the data subnet under the Private IPv4 address. Here, 10.0.1.24 is the data subnet value.

Figure 10 : vThunder instance



4. Select your client instance from the **Instances** list.

Here, `vth-client` is the client instance name.

5. Click **Connect**.

A **Connect to instance** window with **EC2 Instance Connect** tab is displayed.

6. Click **Connect**.

A **Terminal** window is displayed.

7. Run the following command in the Terminal window to send the traffic from the client machine:

```
curl <vThunder_instance_data_private_IPv4_Address>
```

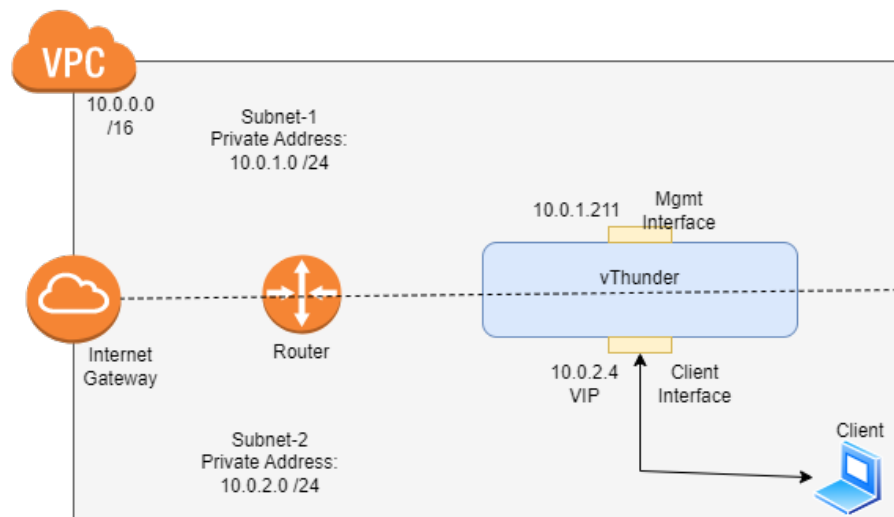
8. Verify if a response is received.

Deploy CFT A10-vThunder_ADC-2NIC-1VM-GLM

Using this template, you can deploy one vThunder instance containing one management interface and one data interface with GLM integration.

[Figure 11](#) illustrates the deployment topology for the 2NIC-1VM-GLM template.

Figure 11 : 2NIC-1VM-GLM Deployment Topology



The following topics are covered:

System Requirements	40
Supported Instance Types	41
Create vThunder Instance	43
Access vThunder using GUI or CLI	46
Configure Server and Client Machine	49
Configure vThunder as an SLB	50
Verify Deployment	56
Verify GLM	58
Verify Traffic Flow	60

System Requirements

When the files from the CFT are downloaded and saved on your local machine, the resources and their default values are displayed as shown in [Table 5](#). You can configure the resources and modify these default values based on your deployment requirements.

Table 5 : System Requirements

Resource Name	Description	Default Value
Stack	A new stack is created with the specified name and location.	In this template, <code>vth</code> is used as the default stack name. Therefore, all the resources has <code>vth</code> as the prefix.
Network Interface Card [NIC]	Two types of interfaces are created for a vThunder instance: - One management interface - One data interface	<code>vth-inst1-mgmt-nic1</code> <code>vth-inst1-data-nic1</code>
Subnet	Two subnets are created with an address prefix each.	<code>vth-vpc-mgmt-subnet1</code> <code>vth-vpc-data-subnet1</code>
Virtual Private Network [VCN]	A virtual private network is assigned to the virtual machine instance.	<code>vth-vpc</code> Address prefix for virtual network: 10.0.0.0/16
Elastic Public IP	One Elastic Public IP address is created and attached to the management interface of the vThunder instance.	<code>vth-inst1-mgmt-nic1-ip</code>
Security Group	One security group is created for the management interface. One security group is created for the data interface.	Here, the tag names are: <code>vth-sg-mgmt</code> <code>vth-sg-data</code>

Resource Name	Description	Default Value
	Logical name: - vThunderSecurityGroupMgmt - vThunderSecurityGroupData	
vThunder Instance	A vThunder EC2 instance is created. Default type: m4.xlarge (40 Gb memory) Table 6 lists the supported instance types.	vth-inst1
Ubuntu Server Instance	One Ubuntu Server instance is created. Default size: t2.micro	vth-server

Supported Instance Types

[Table 6](#) provides detailed information about the supported instance types.

Table 6 : List of Supported Instance Types

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

Instance	vCPU	Memory	Number of Network Interfaces
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

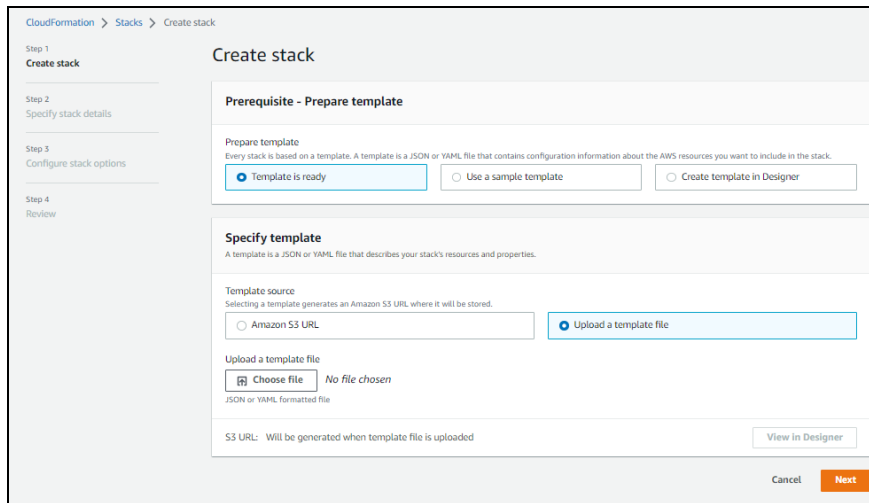
Instance	vCPU	Memory	Number of Network Interfaces
r5d.xlarge	4	32768	4
r5d.2xlarge	8	65536	4
r5d.4xlarge	16	131072	8
r5.large	2	16384	3
r5.xlarge	4	32768	4
r5.2xlarge	8	65536	4
r5.4xlarge	16	131072	8
r4.large	2	15616	3
r4.xlarge	4	31232	4
r4.2xlarge	8	62464	4
r4.4xlarge	16	124928	8
r4.8xlarge	32	249856	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

Create vThunder Instance

To create a vThunder instance, perform the following steps:

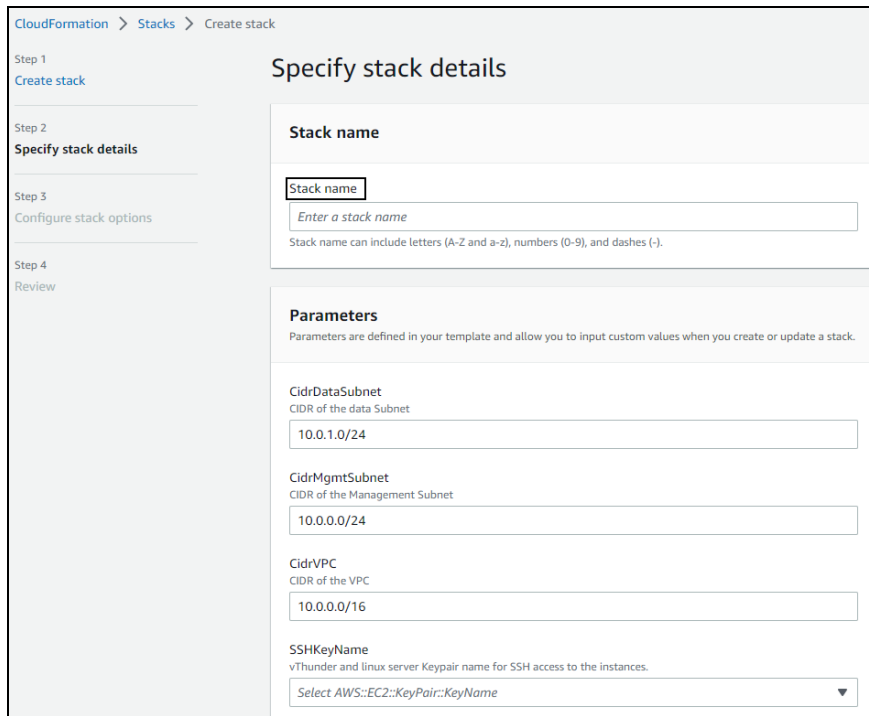
1. From **AWS Management Console**, navigate to **CloudFormation > Stacks > Create Stack > With new resources (standard)**.
The **Create stack** window is displayed.

Figure 12 : Create stack window



2. In the **Prerequisite - Prepare template** section, select **Template is ready**. After the selecting this option, the **Specify template** section is displayed.
3. In the **Specify template** section, select **Upload a template file** and click **Choose file** to browse and upload the following template file from the downloaded CFT folder:
CFT_TMPL_2NIC_1VM_GLM_1.json
The selected template file name is displayed as the chosen file.
4. Click **Next**.
The **Specify stack details** window is displayed.

Figure 13 : Specify stack details window



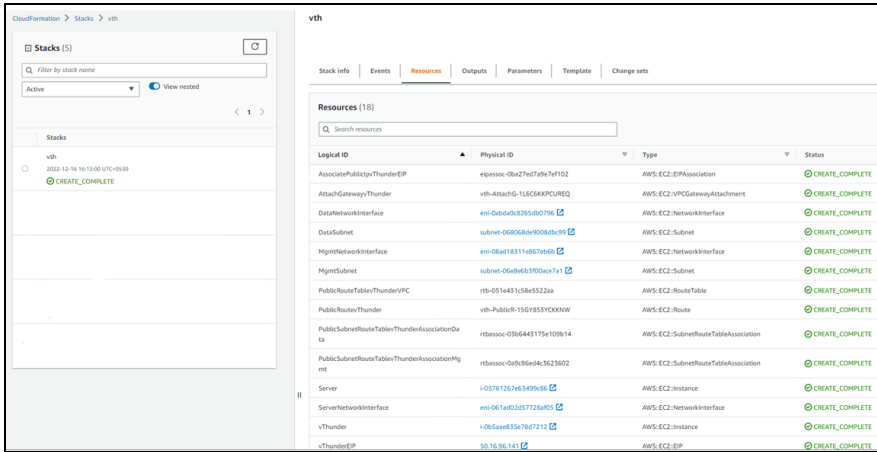
5. In the **Specify stack details** window, enter or select the following:
 - a. In the **Stack name** section, enter a **Stack name**.
Here, **vth** is provided as the default stack name.
 - b. In the **Parameters** section, enter or select the required values in the following fields:
 - **KeyPairName:** *<your SSH key>*
 - **TagValue:** **a10-vthunder-adc**
 - **Zone:** *<your availability zone>*
 - c. Verify the other fields and change the values appropriately. (Optional)
6. Click **Next**.
The **Configure stack options** window is displayed.
7. Verify the fields and change the values appropriately. (Optional)
8. Click **Next**.
The **Review** window is displayed.

9. Verify if all the stack configurations are correct and then click **Submit**.

NOTE: The system may take a few minutes to create the resources and display the stack status as **CREATE_COMPLETE**.

10. Verify if all the above resources are created in the **AWS Management Console > CloudFormation > Stacks > <stack_name> > Resources** tab.

Figure 14 : Resource listing in the resource group



Logical ID	Physical ID	Type	Status
AssociatePublicIpThunderEIP	eipassoc-0ba27e7d7e7d7d102	AWS::EC2::EIPAssociation	CREATE_COMPLETE
AttachGatewayThunder	vth-AttachG-1LEGGKPCUREQ	AWS::EC2::VPCEGatewayAttachment	CREATE_COMPLETE
DataNetworkInterface	eni-04bda0b0265d0796	AWS::EC2::NetworkInterface	CREATE_COMPLETE
DataSubnet	subnet-06008d9000d0c09	AWS::EC2::Subnet	CREATE_COMPLETE
MgmtNetworkInterface	eni-0ba018311d857d6d6	AWS::EC2::NetworkInterface	CREATE_COMPLETE
MgmtSubnet	subnet-06d6b03f00a07e1	AWS::EC2::Subnet	CREATE_COMPLETE
PublicRouteTableThunderVPC	rtb-051e431c58d522a4	AWS::EC2::RouteTable	CREATE_COMPLETE
PublicRouteThunder	vth-PublicR-15GV853YCKKNW	AWS::EC2::Route	CREATE_COMPLETE
PublicSubnetRouteTableThunderAssociationDe	rtbassoc-03b6443175e109b14	AWS::EC2::SubnetRouteTableAssociation	CREATE_COMPLETE
PublicSubnetRouteTableThunderAssociationMg	rtbassoc-04f8d6e43623602	AWS::EC2::SubnetRouteTableAssociation	CREATE_COMPLETE
Server	i-0376126764349a86	AWS::EC2::Instance	CREATE_COMPLETE
ServerNetworkInterface	eni-061aa0205772ba05	AWS::EC2::NetworkInterface	CREATE_COMPLETE
vThunder	i-065aa8b35e78d7212	AWS::EC2::Instance	CREATE_COMPLETE
vThunderEIP	elb-16-96-141	AWS::EC2::EIP	CREATE_COMPLETE

The vThunder instance is listed under **Resources** tab.

Access vThunder using GUI or CLI

vThunder instance can be accessed using any of the following ways:

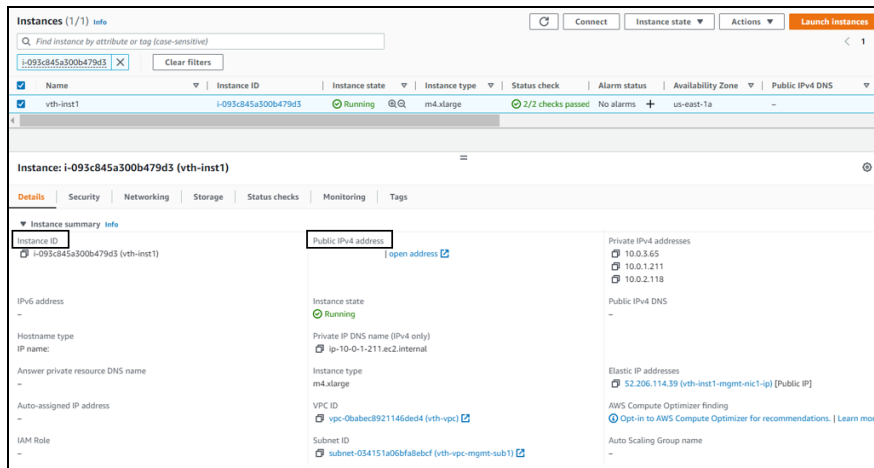
- [Access vThunder using GUI](#)
- [Access vThunder using CLI](#)

Access vThunder using GUI

To access vThunder instance using GUI, perform the following steps:

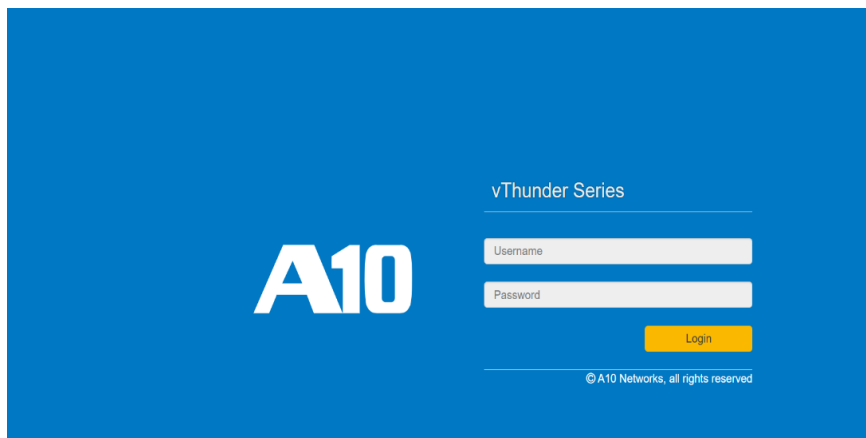
1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select your instance name.
Here, **vth-inst1** is the vThunder instance.

Figure 15 : vThunder instance



3. Copy the **Public IPv4 address** from the **Details** tab and replace the IP address in the below link:
http://<vThunder_public_IPv4_address>
4. Open the updated link in any browser.
 The vThunder login window is displayed.

Figure 16 : vThunder GUI



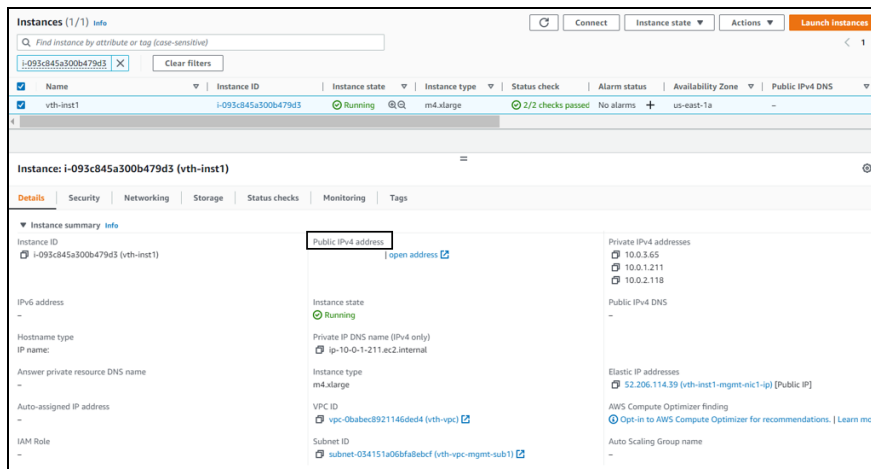
5. Enter the following credentials and click **Login**:
 - Username – **admin**
 - Password – **EC2 Instance ID**
 The home page is displayed if the entered credentials are correct.

Access vThunder using CLI

To access vThunder instance using CLI, perform the following steps:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select your instance name.
Here, **vth-inst1** is the vThunder instance.

Figure 17 : vThunder instance



3. Copy the **Public IPv4 address** from the **Details** tab.
4. Open any SSH client and provide the following to establish a connection:
 - Hostname: Public IPv4 address
 - Username: admin
 - Key: SSH Key
5. Connect to the session.
6. In the SSH client session, run the following commands:

```
vThunder (NOLICENSE) > enable <---Execute command--->
Password: <---just press Enter key--->
vThunder (NOLICENSE) # config <---Configuration mode--->
vThunder (config) (NOLICENSE) #
```

The vThunder instance is ready to use.

Configure Server and Client Machine

To test the traffic flow via vThunder, create and configure a server machine and a client machine:

- [Configure a Server Machine](#)
- [Configure a Client Machine](#)

Configure a Server Machine

To configure a server machine, perform the following steps:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select your server instance name.
Here, `vth-glm-server` is the server instance name.
3. Click **Connect**.
A **Connect to instance** window with **EC2 Instance Connect** tab is displayed.
4. Click **Connect**.
A **Terminal** window is displayed.
5. Run the following command in the Terminal window to update all the package information:

```
sudo apt update
```

6. Run the following command in the Terminal window to create an Apache Server virtual machine:

```
sudo apt install apache2
```

While the Apache server is getting installed, you get a prompt to continue further. Enter 'Y' to continue. After the installation is complete, a newline prompt is displayed.

Configure a Client Machine

To configure a client machine, perform the following steps:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Click **Launch Instances**.
A **Launch an instance** window is displayed.
3. In the **Name and tags** section, enter an instance name.
Here, enter `vth-glm-client` as the client instance name.
4. In the **Application and OS Images** section, select **Ubuntu**.
5. In the **Instance type** section, select the required instance type.
6. In the **Key pair (login)** field, select your SSH key.
7. In the **Network settings** section, click **Edit** to edit the following:
 - VPC: *your VPC*
Here, `vth-glm-vpc` is the VPC.
 - Subnet: Data subnet
Here, `10.0.1.0/24` is the data subnet value.
 - Auto-assign public IP: Enable
 - Firewall (security groups): Select existing security group
 - Common security groups: *your data security group*
Here, `vth-glm-vThunderSecurityGroupData` is the security group.
8. Click **Launch instance**.

NOTE: The system may take a few minutes to launch the instance.

The client instance is displayed in the **Instances** list with the status as **Running**.

Configure vThunder as an SLB

The following topics are covered:

- [Initial Setup](#)
- [Deploy vThunder as an SLB](#)

Initial Setup

Before deploying vThunder on AWS cloud as an SLB, you need to configure the corresponding parameters in the CFT.

To configure the parameters, perform the following steps:

1. Navigate to the folder where you have downloaded the CFT, and open the CFT_TMPL_2NIC_1VM_GLM_CONFIG_SLB_SSL_GLM_PARAM.json with a text editor.

NOTE: Each parameter has a default value mentioned in the parameter file.

2. Configure the stack name created using CFT.

```
"stackDetails": {
  "value": [
    {
      "stackName": "vth"
    }
  ]
},
```

3. Configure SLB server ports.

```
"server-list": {
  "value": [
    {
      "port-list": [
        {
          "port-number": 53,
          "protocol": "udp"
        },
        {
          "port-number": 80,
          "protocol": "tcp"
        },
        {
          "port-number": 443,
          "protocol": "tcp"
        }
      ]
    }
  ]
}
```

```

    }
  ]
},

```

4. Configure Service Group List ports.

The Service group name by default is “sg+port_number”. If you may want to change the service group name then after changing the name, change the names in the Virtual servers as well.

```

"serviceGroupList": {
  "value": [
    {
      "name": "sg443",
      "protocol": "tcp",
      "member-list": [
        {
          "port": 443
        }
      ]
    },
    {
      "name": "sg53",
      "protocol": "udp",
      "member-list": [
        {
          "port": 53
        }
      ]
    },
    {
      "name": "sg80",
      "protocol": "tcp",
      "member-list": [
        {
          "port": 80
        }
      ]
    }
  ]
}

```

```
]
},
```

5. Configure a Virtual Server.

The virtual server default name is “vip”. It is the Private IP address of Ethernet1.

```
"virtualServerList": {
  "virtual-server-name": "vip",
  "metadata": {
    "description": "virtual server is using ethernet 1 ip address"
  },
  "value": [
    {
      "port-number": 53,
      "protocol": "udp",
      "auto": 1,
      "service-group": "sg53"
    },
    {
      "port-number": 80,
      "protocol": "http",
      "auto": 1,
      "service-group": "sg80"
    },
    {
      "port-number": 443,
      "protocol": "https",
      "auto": 1,
      "service-group": "sg443"
    }
  ]
},
```

6. Configure SSL.

```
"sslConfig": {
  "requestTimeout": 40,
  "Path": "server.pem",
  "File": "server",
```

```
"CertificationType": "pem"
}
```

NOTE: By default, SSL configuration is disabled i.e., no SSL configuration is applied.

The server.pem file is included in the downloaded CFT folder. If necessary, you can edit this file or use a different certificate file. When you use a different certificate file, remember to change the path accordingly.

7. Configure your GLM account details.

To obtain the **Entitlement Token**, go to [GLM](#) > **Licenses** > select your license > **Overview** > **Info** tab.

```
"user_name": {
  "value": "xxxxxxxx@a10networks.com"
},
"user_password": {
  "value": "xxxxxxxx"
},
"entitlement_token": {
  "value": "xxxxxxxx"
}
}
```

8. Verify if all the configurations in the CFT_TMPL_2NIC_1VM_GLM_CONFIG_SLB_SSL_GLM_PARAM.json file are correct and then save the changes.

Deploy vThunder as an SLB

To deploy vThunder on AWS cloud as an SLB, perform the following steps:

1. From your command prompt, navigate to the folder where you have downloaded the CFT.
2. Run the following command to create vThunder SLB instance:

```
python ./CFT_TMPL_2NIC_1VM_GLM_CONFIG_SLB_SSL_GLM_2.py
```

A message is prompted to upload the SSL certificate.

3. Enter **yes** if you want to upload the SSL certificate and configure GLM.

```
Do you want to upload ssl certificate(yes/no)?yes
Do you want to configure GLM (yes/no)?yes
```

A message is prompted that displays the criteria for a valid admin user password.

```
Primary conditions for password validation, user should provide the new
password according to the given combination:
```

```
Minimum length of 9 characters
Minimum lowercase character should be 1
Minimum uppercase character should be 1
Minimum number should be 1
Minimum special character should be 1
Should not include repeated characters
Should not include more than 3 keyboard consecutive characters.
```

A message is prompted to enter a new password.

4. Provide the new password for the admin user and then confirm the new password.

The new password should follow the valid admin user password criteria mentioned in the above step. For more information, see [Default Password Policy](#).

```
Enter vThunder's new password:
Confirm new password:
```

If both the password and confirm password field values match and complies to the valid admin user password criteria, a message 'Password changed successfully' is displayed along with the other configurations.

```
Password changed successfully.
Password change configurations saved on partition: shared
-----
-----
configured ethernet ip
Configured server vth-server
Configured virtual servers
```

```
License activation completed.  
Configure glm configuration.  
Glm license request sent successfully.  
SSL Configured.  
Configurations are saved on partition: shared
```

Verify Deployment

To verify vThunder SLB deployment using CFT using CLI, perform the following steps:

1. Access the vThunder instance using CLI.
2. Run the following command to verify the running configuration:

```
vThunder(config)#show running-config
```

If the deployment is successful, the following SLB with GLM configuration is displayed:

```
!Current configuration: 0 bytes  
!Configuration last updated at 17:36:35 GMT Wed Nov 30 2022  
!Configuration last saved at 17:35:40 GMT Wed Nov 30 2022  
!64-bit Advanced Core OS (ACOS) version 5.2.1, build 153 (Dec-11-  
2020,14:34)  
!  
!  
glm use-mgmt-port  
glm enable-requests  
glm token A10f771cecbe  
!  
interface ethernet 1  
    enable  
    ip address dhcp  
!  
!  
slb server vth-server 10.0.1.120  
    port 53 udp  
    port 80 tcp  
    port 443 tcp
```



```

!
slb service-group sg443 tcp
    member vth-server 443
!
slb service-group sg53 udp
    member vth-server 53
!
slb service-group sg80 tcp
    member vth-server 80
!
slb virtual-server vip use-if-ip ethernet 1
    port 53 udp
        source-nat auto
        service-group sg53
    port 80 http
        source-nat auto
        service-group sg80
    port 443 https
        source-nat auto
        service-group sg443
!
!
end

```

3. Run the following command to verify the SSL configuration:

```
vThunder(config)#show pki cert
```

If the deployment is successful, the following SSL configuration is displayed:

Name	Type	Expiration	Status

server certificate	Jan 28 12:00:00 2028 GMT	[Unexpired, Bound]	

NOTE:

If you want to access the vThunder instance using GUI, re-login the instance using the new admin user password and NOT the *EC2 Instance ID*.

Verify GLM

The application of license can be verified using any of the following ways:

- [Verify License using GUI](#)
- [Verify License using CLI](#)

Verify License using GUI

To verify license using GUI, perform the following steps:

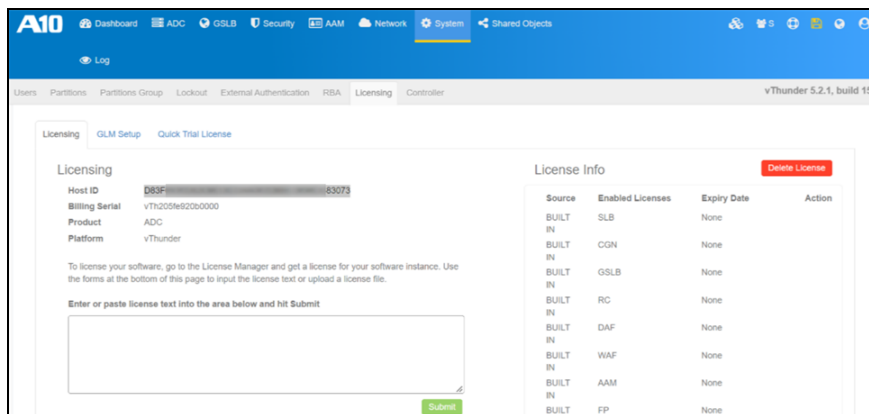
1. Log in to your vThunder instance using **Public IPv4 address**.

Here, `vth-inst1` is the vThunder instance.

2. Navigate to **Profile > Setting > Licensing**.
3. Click the **Licensing** tab.

If the license is successfully applied on vThunder, the Host ID is displayed.

Figure 18 : GLM > Licensing window



Verify License using CLI

To verify license using CLI, perform the following steps:

1. Verify if NOLICENSE is removed from the vThunder prompt.
2. Run the following command on vThunder:

```
vThunder(config)#show license
```

If the license is successfully applied on vThunder, the Host ID is displayed:

```
Host ID      : D83F****82EB633D****067DB84136****83073
```

3. Run the following command on vThunder instance:

```
vThunder(config)#show license-info
```

If the license is successfully applied on vThunder, the following GLM configuration is displayed:

```
Host ID      : D83F****82EB633D****067DB84136****83073
USB ID       : Not Available
Billing Serials: vTh205fe920b0000
Token        : Not Available
Product      : ADC
Platform     : vThunder
Burst        : Disabled
GLM Ping Interval In Hours : 24
```

```
-----
-----
Enabled Licenses   Expiry Date (UTC)           Notes
-----
-----
SLB                None
CGN                None
GSLB               None
RC                 None
DAF                None
WAF                None
AAM                None
FP                 None
WEBROOT            N/A                Requires an additional Webroot
license.
THREATSTOP         N/A                Requires an additional ThreatSTOP
license.
```

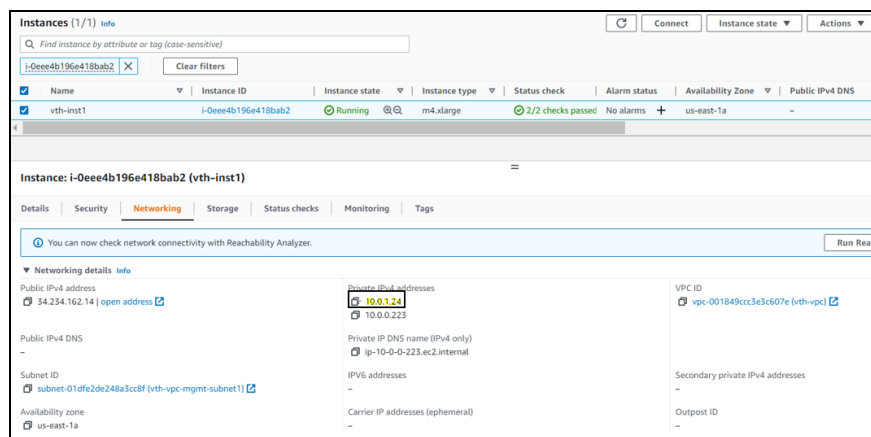
QOSMOS	N/A	Requires an additional QOSMOS license.
WEBROOT_TI	N/A	Requires an additional Webroot Threat Intel license.
IPSEC_VPN	N/A	Requires an additional IPsec VPN license.
500 Mbps Bandwidth 20-January-2023		

Verify Traffic Flow

To verify the traffic flow from client machine to server machine via vThunder, perform the following:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select the vThunder instance name and then click the **Networking** tab. Here, **vth-inst1** is the instance name.
3. Copy the IP address of the data subnet under the **Private IPv4 address**. Here, **10.0.1.24** is the data subnet value.

Figure 19 : vThunder instance



4. Select your client instance from the **Instances** list. Here, **vth-client** is the client instance name.
5. Click **Connect**.
A **Connect to instance** window with **EC2 Instance Connect** tab is displayed.
6. Click **Connect**.
A **Terminal** window is displayed.

7. Run the following command in the Terminal window to send the traffic from the client machine:

```
curl <vThunder_instance_data_private_IPv4_Address>
```

8. Verify if a response is received.

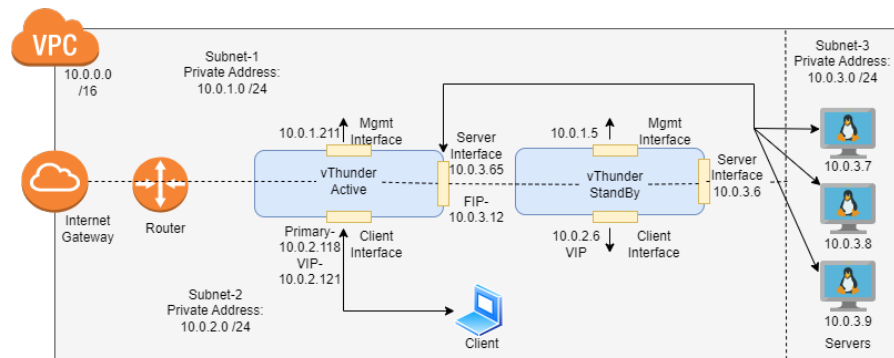
Deploy CFT A10-vThunder_ADC-3NIC-2VM-HA

Using this template, you can deploy two vThunder instances containing:

- One management interface and two data interfaces for each vThunder instance
- High Availability (HA) support

[Figure 20](#) illustrates the deployment topology for the 3NIC-2VM-HA template.

Figure 20 : 3NIC-2VM-HA Topology



The following topics are covered:

System Requirements	63
Supported Instance Types	64
Create vThunder Instances	67
Access vThunder using GUI or CLI	69
Configure Server and Client Machine	72
Configure FTP server	74
Configure vThunder as an SLB with HA	74
Verify Deployment	82
Verify Traffic Flow	89

System Requirements

When the files from the CFT are downloaded and saved on your local machine, the resources and their default values are displayed as shown in [Table 7](#). You can configure the resources and modify these default values based on your deployment requirements.

Table 7 : System Requirements

Resource Name	Description	Default Value
Stack	A new stack is created with the specified name and location.	In this template, <code>vth</code> is used as the default stack name. Therefore, all the resources has <code>vth</code> as the prefix.
Network Interface Card [NIC]	Two types of interfaces are created for each vThunder instance: - One management interface - Two data interfaces	<code>vth-inst1-mgmt-nic1</code> <code>vth-inst1-data-nic2</code> <code>vth-inst1-data-nic3</code> <code>vth-inst2-mgmt-nic1</code> <code>vth-inst2-data-nic2</code> <code>vth-inst2-data-nic3</code>
Subnet	Three subnets are created with an address prefix each.	<code>vth-vpc-mgmt-sub1</code> <code>vth-vpc-data-sub1</code> <code>vth-vpc-data-sub2</code>
Virtual Private Network [VCN]	A virtual private network is assigned to the virtual machine instance.	<code>vth-vpc</code> Address prefix for virtual network: 10.0.0.0/16
Elastic Public IP	Two Elastic Public IP addresses are created and attached to the	<code>vth-inst1-mgmt-nic1-ip</code> <code>vth-inst2-mgmt-nic1-ip</code>

Resource Name	Description	Default Value
	management interfaces of the vThunder instances.	
Security Group	Two security groups are created: - One security group for the management interface. - One security group for the data interface. Logical name: - vThunderSecurityGroupMgmt - vThunderSecurityGroupData	Here, the tag names are: vth-sg-mgmt vth-sg-data
vThunder Instance	Two vThunder EC2 instances are created: - One active - One standby Default type: m4.xlarge (40 Gb memory) Table 8 lists the supported instance types.	vth-inst1 vth-inst2
Server Instance	One Ubuntu Server instance is created. Default Size: t2.micro	vth-server
FTP Server	One FTP server is created.	vth-ftp-server

Supported Instance Types

[Table 8](#) provides detailed information about the supported instance types.

Table 8 : List of Supported Instance Types

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

Instance	vCPU	Memory	Number of Network Interfaces
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

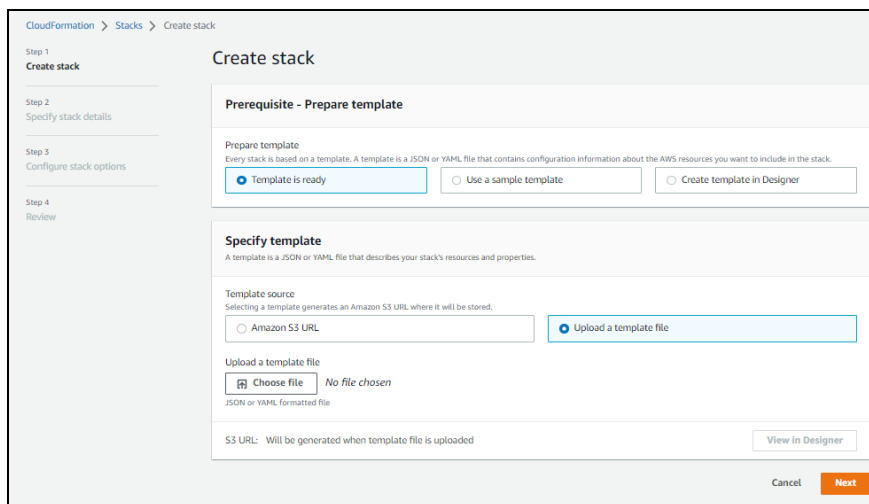
Instance	vCPU	Memory	Number of Network Interfaces
z1d.large	2	16384	3
z1d.xlarge	4	32768	4
z1d.2xlarge	8	65536	4
z1d.3xlarge	12	98304	8
z1d.6xlarge	24	196608	8

Create vThunder Instances

To create vThunder instances, perform the following steps:

1. From **AWS Management Console**, navigate to **CloudFormation > Stacks > Create Stack > With new resources (standard)**.
The **Create stack** window is displayed.

Figure 21 : Create stack window



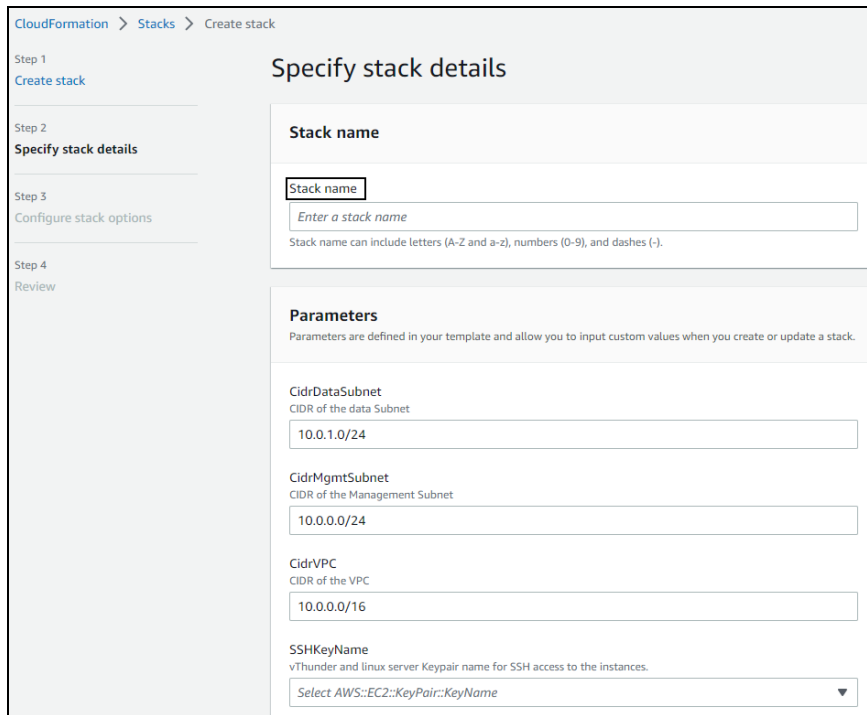
2. In the **Prerequisite - Prepare template** section, select **Template is ready**.
After the selecting this option, the **Specify template** section is displayed.
3. In the **Specify template** section, select **Upload a template file** and click **Choose file** to browse and upload the following template file from the downloaded CFT folder:
CFT_TMPL_3NIC_2VM_HA_1.json

The selected template file name is displayed as the chosen file.

4. Click **Next**.

The **Specify stack details** window is displayed.

Figure 22 : Specify stack details window



5. In the **Specify stack details** window, enter or select the following:

a. In the **Stack name** section, enter a **Stack name**.

Here, **vth** is provided as the default stack name.

b. In the **Parameters** section, enter or select the required values in the following fields:

- **KeyPairName:** *<your SSH key>*
- **TagValue:** **a10-vthunder-adc**
- **Zone:** *<your availability zone>*

c. Verify the other fields and change the values appropriately. (Optional)

6. Click **Next**.

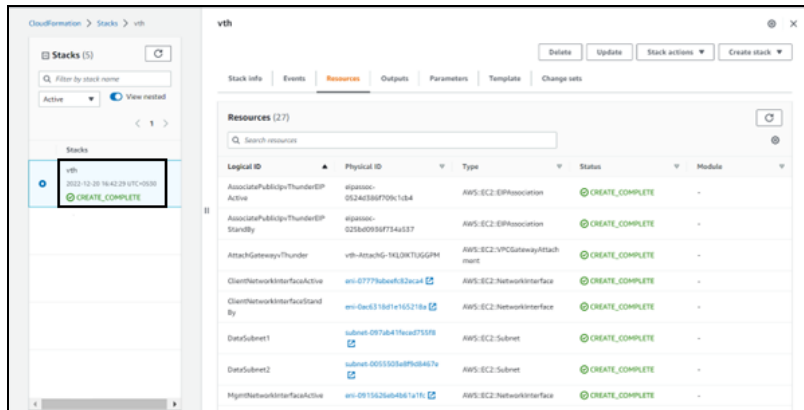
The **Configure stack options** window is displayed.

7. Verify the fields and change the values appropriately. (Optional)
8. Click **Next**.
The **Review** window is displayed.
9. Verify if all the stack configurations are correct and then click **Submit**.

NOTE: The system may take a few minutes to create the resources and display the stack status as **CREATE_COMPLETE**.

10. Verify if all the above resources are created in the **AWS Management Console** > **CloudFormation** > **Stacks** > <stack_name> > **Resources** tab.

Figure 23 : Resource listing in the resource group



The two vThunder instances are listed under **Resources** tab.

Access vThunder using GUI or CLI

vThunder instances can be accessed using any of the following ways:

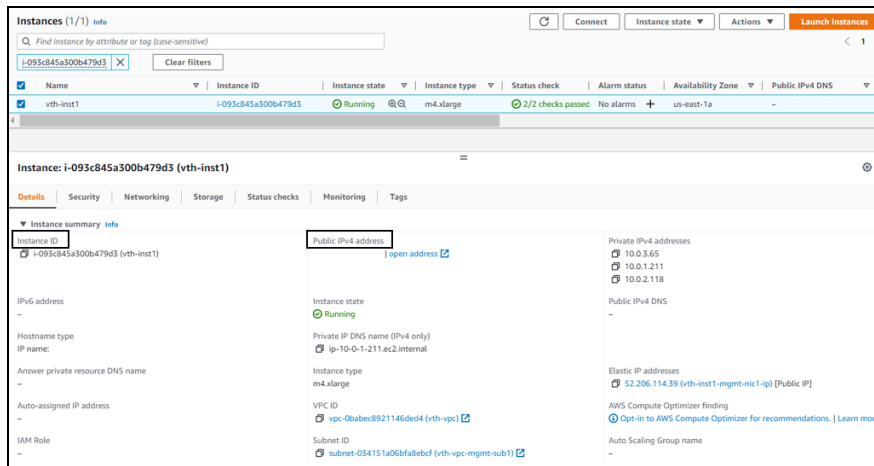
- [Access vThunder using GUI](#)
- [Access vThunder using CLI](#)

Access vThunder using GUI

To access vThunder instances using GUI, perform the following steps:

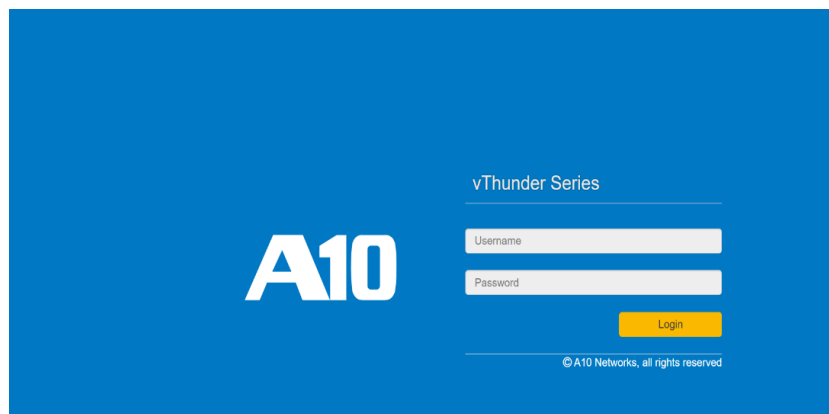
1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select your instance name.
Here, **vth-inst1**, **vth-inst2** are the vThunder instances.

Figure 24 : vThunder instance



3. For each vThunder instance, perform the following steps:
 - a. Copy the **Public IPv4 address** from the **Details** tab and replace the IP address in the below link:
http://<vThunder_public_IPv4_address>
 - b. Open the updated link in any browser.
The vThunder login window is displayed.

Figure 25 : vThunder GUI



c. Enter the following credentials and click **Login**:

- Username – **admin**
- Password – *EC2 Instance ID*

The home page is displayed if the entered credentials are correct.

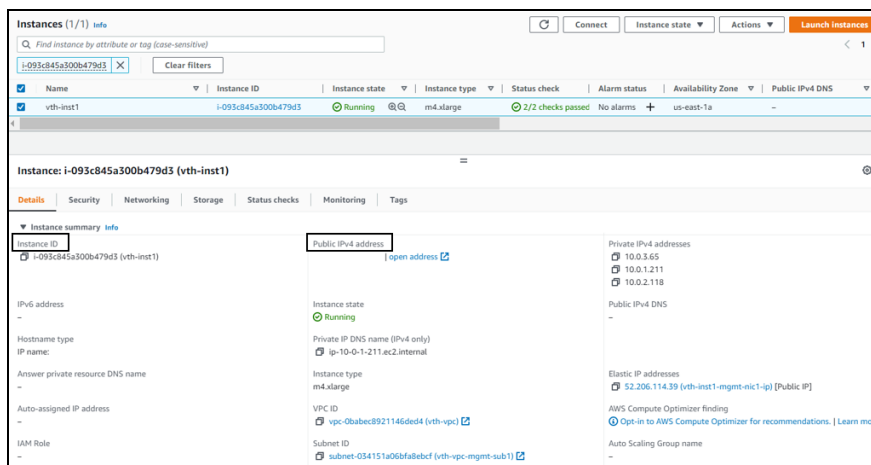
Access vThunder using CLI

To access vThunder instances using CLI, perform the following steps:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select your instance name.

Here, **vth-inst1**, **vth-inst2** are the vThunder instances.

Figure 26 : vThunder instance



3. For each vThunder instance, perform the following steps:
 - a. Copy the **Public IPv4 address** from the **Details** tab.
 - b. Open any SSH client and provide the following to establish a connection:
 - Hostname: Public IPv4 address
 - Username: admin
 - Key: SSH Key
 - c. Connect to the session.

d. In the SSH client session, run the following commands:

```
vThunder (NOLICENSE)>enable <---Execute command--->
Password: <---just press Enter key--->
vThunder (NOLICENSE) #config <---Configuration mode--->
vThunder (config) (NOLICENSE) #
```

The vThunder instances are ready to use.

Configure Server and Client Machine

To test the traffic flow via vThunder, create and configure a server machine and a client machine:

- [Configure a Server Machine](#)
- [Configure a Client Machine](#)

Configure a Server Machine

To configure a server machine, perform the following steps:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select your server instance name.
Here, **vth-server** is the server instance name.
3. Click **Connect**.
A **Connect to instance** window with **EC2 Instance Connect** tab is displayed.
4. Click **Connect**.
A **Terminal** window is displayed.
5. Run the following command in the Terminal window to update all the package information:

```
sudo apt update
```

6. Run the following command in the Terminal window to create an Apache Server virtual machine:

```
sudo apt install apache2
```


While the Apache server is getting installed, you get a prompt to continue further. Enter 'Y' to continue. After the installation is complete, a newline prompt is displayed.

Configure a Client Machine

To configure a client machine, perform the following steps:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Click **Launch Instances**.
A **Launch an instance** window is displayed.
3. In the **Name and tags** section, enter an instance name.
Here, enter `vth-client` as the client instance name.
4. In the **Application and OS Images** section, select **Ubuntu**.
5. In the **Instance type** section, select the required instance type.
6. In the **Key pair (login)** field, select your SSH key.
7. In the **Network settings** section, click **Edit** to edit the following:
 - VPC: *your VPC*
Here, `vth-vpc` is the VPC.
 - Subnet: Data subnet
Here, `10.0.2.0/24` is the data subnet value.
 - Auto-assign public IP: Enable
 - Firewall (security groups): Select existing security group
 - Common security groups: *your data security group*
Here, `vth-vThunderSecurityGroupData` is the security group.
8. Click **Launch instance**.

NOTE: The system may take a few minutes to launch the instance.

The client instance is displayed in the **Instances** list with the status as **Running**.

Configure FTP server

In a High Availability (HA) configuration, IP switching occurs between two vThunder instances. The IP switching is enabled when AWS keys are imported on the vThunder instances. To import the AWS Access Keys, configure an FTP server.

To configure an FTP server, perform the following steps:

1. From **AWS Management Console**, navigate to **EC2 > Instances > <FTP_server_name>**.
2. Click **Launch Instances**.
A **Launch an instance** window is displayed.
3. Run the following command in the FTP server instance to create an Apache Server virtual machine:

```
sudo apt install apache2
```

While the Apache server is getting installed, you get a prompt to continue further. Enter 'Y' to continue. After the installation is complete, a newline prompt is displayed.

4. Run the following command in the same instance to open the **aws_access_key.txt** file:

```
sudo vi /var/www/html/aws_access_key.txt
```

5. Press **Esc** and enter **i** to enable edit/insert mode.
6. Locate and open the **credentials** file from downloaded CFT folder.
7. Copy the updated access key ID and secret access key as per your AWS account and paste in the **aws_access_key.txt** file. For more information, see [Prerequisites](#).
8. After the changes, press **Esc** then type **:wq** to save the changes and exit.

Configure vThunder as an SLB with HA

The following topics are covered:

- [Initial Setup](#)
- [Deploy vThunder as an SLB](#)
- [Verify Access Keys on vThunder Instances](#)

Initial Setup

Before deploying vThunder on AWS cloud as an SLB with HA, you need to configure the corresponding parameters in the CFT.

To configure the parameters, perform the following steps:

1. Navigate to the folder where you have downloaded the CFT, and open the CFT_TMPL_3NIC_2VM_HA_CONFIG_SLB_SSL_HA_PARAM.json with a text editor.

NOTE: Each parameter has a default value mentioned in the parameter file.

2. Configure the stack name created using CFT.

```
"stackDetails": {
  "value": [
    {
      "stackName": "vth"
    }
  ]
},
```

3. Configure SLB server ports.

```
"server-list": {
  "value": [
    {
      "metadata": {
        "description": "SLB server host/fqdn-name."
      },
      "port-list": [
        {
          "port-number": 53,
          "protocol": "udp"
        },
        {
```

```
        "port-number": 80,  
        "protocol": "tcp"  
    },  
    {  
        "port-number": 443,  
        "protocol": "tcp"  
    }  
]  
}  
]  
},
```

4. Configure Service Group ports.

The Service group name by default is “sg+port_number”. If you may want to change the service group name then after changing the name, change the names in the Virtual servers as well.

```
"serviceGroupList": {  
    "value": [  
        {  
            "name": "sg443",  
            "protocol": "tcp",  
            "member-list": [  
                {  
                    "port": 443  
                }  
            ]  
        },  
        {  
            "name": "sg53",  
            "protocol": "udp",  
            "member-list": [  
                {  
                    "port": 53  
                }  
            ]  
        },  
        {  
            "name": "sg80",  
            "protocol": "tcp",  
            "member-list": [  
                {  
                    "port": 80  
                }  
            ]  
        }  
    ]  
}
```

```

        "name": "sg80",
        "protocol": "tcp",
        "member-list": [
            {
                "port": 80
            }
        ]
    }
]
},

```

5. Configure a Virtual Server and its ports.

The virtual server default name is “vip”. It is the Private IP address of Ethernet1.

```

"virtualServerList": {
    "virtual-server-name": "vip",
    "metadata": {
        "description": "virtual server is using ethernet 1 ip
address"
    },
    "value": [
        {
            "port-number": 53,
            "protocol": "udp",
            "auto": 1,
            "service-group": "sg53"
        },
        {
            "port-number": 80,
            "protocol": "http",
            "auto": 1,
            "service-group": "sg80"
        },
        {
            "port-number": 443,
            "protocol": "https",
            "auto": 1,
            "service-group": "sg443"
        }
    ]
}

```

```
    }
  ]
},
```

6. Configure DNS.

```
"dns": {
  "value": "8.8.8.8"
},
```

7. Configure a Network Gateway IP.

The default value of network gateway IP address is 10.0.1.1 as this is the first IP address of the data subnet 1 configuration.

```
"rib-list": [
  {
    "ip-dest-addr": "0.0.0.0",
    "ip-mask": "/0",
    "ip-nexthop-ipv4": [
      {
        "ip-next-hop": "10.0.1.1"
      },
      {
        "ip-next-hop": "10.0.2.1"
      }
    ]
  }
],
```

8. Set VRRP-A.

```
"vrrp-a": {
  "set-id": 1
},
```

9. Set a Terminal Idle Timeout.

```
"terminal": {
  "idle-timeout": 0
},
```

10. Set VRID.

For vThunder instance 1, the default priority is 100, and for vThunder instance 2, it is 99 (100-1).

```
"vrid-list": [  
  {  
    "vrid-val": 0,  
    "blade-parameters": {  
      "priority": 100  
    }  
  }  
],
```

11. Configure SSL.

```
"sslConfig": {  
  "requestTimeout": 40,  
  "Path": "server.pem",  
  "File": "server",  
  "CertificationType": "pem"  
}
```

NOTE: By default, SSL configuration is disabled i.e., no SSL configuration is applied.

The server.pem file is included in the downloaded CFT folder. If necessary, you can edit this file or use a different certificate file. When you use a different certificate file, remember to change the path accordingly.

12. Verify if all the configurations in the CFT_TMPL_3NIC_2VM_HA_CONFIG_SLB_SSL_HA_PARAM.json file are correct and then save the changes.

Deploy vThunder as an SLB

To deploy vThunder on AWS cloud as an SLB, perform the following steps:

1. From your command prompt, navigate to the folder where you have downloaded the CFT.
2. Run the following command to create vThunder SLB instance:

```
python ./CFT_TMPL_3NIC_2VM_HA_CONFIG_SLB_SSL_HA_2.py
```

A message is prompted to upload the SSL certificate and configure HA.

3. Enter **yes** if you want to upload the SSL certificate and configure HA.

```
Do you want to upload ssl certificate(yes/no)?yes
Do you want to configure HA(yes/no)?yes
```

A message is prompted that displays the criteria for a valid admin user password.

```
Primary conditions for password validation, user should provide the new
password according to the given combination:
```

```
Minimum length of 9 characters
Minimum lowercase character should be 1
Minimum uppercase character should be 1
Minimum number should be 1
Minimum special character should be 1
Should not include repeated characters
Should not include more than 3 keyboard consecutive characters.
```

A message is prompted to enter a new password.

4. Provide the new password for the admin user and then confirm the new password.

The new password should follow the valid admin user password criteria mentioned in the above step. For more information, see [Default Password Policy](#).

```
Enter vThunder's new password:
Confirm new password:
```

If both the password and confirm password field values match and complies to the valid admin user password criteria, a message 'Password changed successfully' is displayed along with the other configurations.

```
Password changed successfully.
-----
-----
Configuring vThunder with instance id i-05b2d645312ce8006
AWS access key uploaded to vThunder.
configured ethernet ip
```



```
configured ethernet ip
Configured server vth-server
Configure service group
Configured virtual servers
SSL Configured.
Configured primary dns
Configured IP Route
Configured Vrrp A common
Configured idle timeout
Configured vrrp rid
Configured peer group
Configurations are saved on partition: shared

-----

Configuring vThunder with instance id i-06770eb7e9d80e695
AWS access key uploaded to vThunder.
configured ethernet ip
configured ethernet ip
Configured server vth-server
Configure service group
Configured virtual servers
SSL Configured.
Configured primary dns
Configured IP Route
Configured Vrrp A common
Configured idle timeout
Configured vrrp rid
Configured peer group
Configurations are saved on partition: shared
```

Verify Access Keys on vThunder Instances

To verify if the access keys are uploaded on both vThunder instances, perform the following step:

1. Run the following commands on each vThunder instance to verify the access keys upload:

```
vThunder(config) (NOLICENSE) #admin admin
vThunder(config-admin:admin) (NOLICENSE) #aws-accesskey show
[default]
aws_access_key_id = your_aws_access_key_id
aws_secret_access_key = your_aws_secret_access_key
vThunder(config-admin:admin) (NOLICENSE) #exit
vThunder(config) (NOLICENSE) #
vThunder(config) (NOLICENSE) #
vThunder-Active(config) (NOLICENSE) #
```

The access keys are displayed.

NOTE: Once the AWS access keys are imported to vThunder, the FTP server gets deleted automatically.

Verify Deployment

To verify vThunder SLB deployment using CFT, perform the following steps:

1. Access the vThunder instances using CLI.
2. Run the following command on vThunder instance 1 using CLI:

```
vThunder-Active(config) (NOLICENSE) #show running-config
```

If the deployment is successful, the following SLB configuration is displayed:

```
!Current configuration: 349 bytes
!Configuration last updated at 10:56:58 GMT Fri Jan 6 2023
!Configuration last saved at 10:53:34 GMT Fri Jan 6 2023
!64-bit Advanced Core OS (ACOS) version 5.2.1, build 153 (Dec-11-
2020,16:36)
!
vrrp-a common
  device-id 1
  set-id 1
  enable
!
terminal idle-timeout 0
!
```

```
ip dns primary 8.8.8.8
!
!
interface ethernet 1
    enable
    ip address dhcp
!
interface ethernet 2
    enable
    ip address dhcp
!
vrrp-a vrid 0
    floating-ip 10.0.3.12
    blade-parameters
        priority 99
!
vrrp-a peer-group
    peer 10.0.2.117
    peer 10.0.2.173
!
ip route 0.0.0.0 /0 10.0.1.1
ip route 0.0.0.0 /0 10.0.2.1
!
slb server vth-server 10.0.3.23
    port 53 udp
    port 80 tcp
    port 443 tcp
!
slb service-group sg443 tcp
    member vth-server 443
!
slb service-group sg53 udp
    member vth-server 53
!
slb service-group sg80 tcp
    member vth-server 80
!
```

```

slb virtual-server vip 10.0.2.121
  port 53 udp
    source-nat auto
    service-group sg53
  port 80 http
    source-nat auto
    service-group sg80
  port 443 https
    source-nat auto
    service-group sg443
!
!
end
!Current config commit point for partition 0 is 0 & config mode is
classical-mode
vThunder-Active(config) (NOLICENSE) #

```

3. Run the following command on vThunder instance 1 using CLI:

```
vThunder-Active(config) (NOLICENSE) #show pki cert
```

If the deployment is successful, the following SSL configuration is displayed:

Name	Type	Expiration	Status

server certificate	Jan 28 12:00:00 2028 GMT	[Unexpired, Bound]	

At this point, the vThunder instance 2 has the following prompt:

```
vThunder-Standby(config) (NOLICENSE) #
```

Figure 27 : vThunder instance 1 - Active

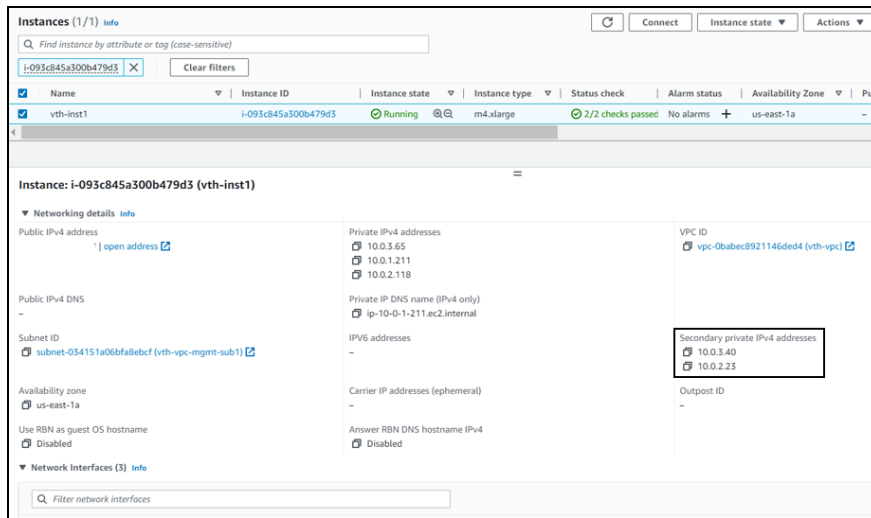
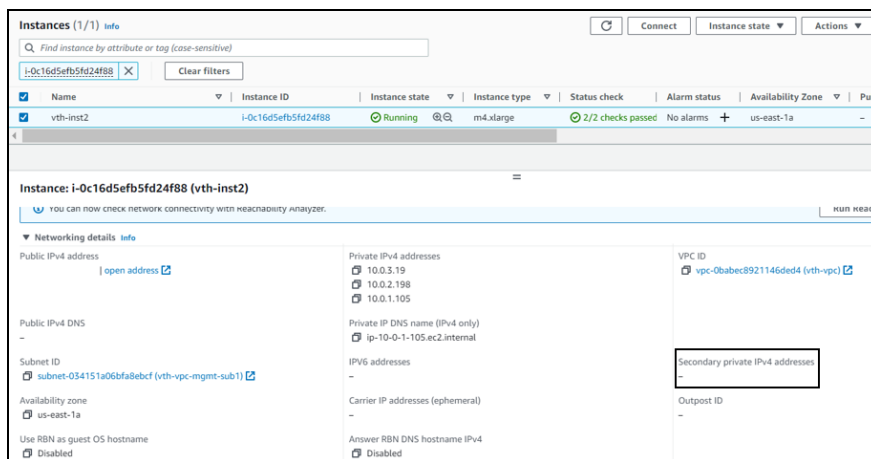


Figure 28 : vThunder instance 2 - Standby



4. Run the following command on vThunder instance 1 using CLI:

```
vThunder-Active(config) (NOLICENSE) #vrrp-a force-self-standby enable
vThunder-Active(config) (NOLICENSE) #
vThunder-ForcedStandby(config) (NOLICENSE) #
```

At this point, IP switching occurs and the vThunder instance 2 prompt becomes:

```
vThunder-Active(config) (NOLICENSE) #
```

5. Run the following command on vThunder instance 2 using CLI:

```
vThunder-Active(config) (NOLICENSE) #show running-config
```

If the deployment is successful, the following SLB configuration is displayed:

```
!Current configuration: 282 bytes
!Configuration last updated at 10:53:35 GMT Fri Jan 6 2023
!Configuration last saved at 10:53:37 GMT Fri Jan 6 2023
!64-bit Advanced Core OS (ACOS) version 5.2.1, build 153 (Dec-11-
2020,16:36)
!
vrrp-a common
    device-id 2
    set-id 1
    enable
!
terminal idle-timeout 0
!
ip dns primary 8.8.8.8
!
!
interface ethernet 1
    enable
    ip address dhcp
!
interface ethernet 2
    enable
    ip address dhcp
!
vrrp-a vrid 0
    floating-ip 10.0.3.12
    blade-parameters
        priority 98
!
vrrp-a peer-group
    peer 10.0.2.186
    peer 10.0.2.6
!
ip route 0.0.0.0 /0 10.0.1.1
ip route 0.0.0.0 /0 10.0.2.1
!
```

```
slb server vth-server 10.0.3.23
  port 53 udp
  port 80 tcp
  port 443 tcp
!
slb service-group sg443 tcp
  member vth-server 443
!
slb service-group sg53 udp
  member vth-server 53
!
slb service-group sg80 tcp
  member vth-server 80
!
slb virtual-server vip 10.0.2.23
  port 53 udp
    source-nat auto
    service-group sg53
  port 80 http
    source-nat auto
    service-group sg80
  port 443 https
    source-nat auto
    service-group sg443
!
sflow setting local-collection
!
sflow collector ip 127.0.0.1 6343
!
!
end
!Current config commit point for partition 0 is 0 & config mode is
classical-mode
vThunder-Active(config) (NOLICENSE) #
```

6. Run the following command on vThunder instance 2 using CLI:

```
vThunder-Active(config) (NOLICENSE) #show pki cert
```

If the deployment is successful, the following SSL configuration is displayed:

Name	Type	Expiration	Status
server certificate	Jan 28 12:00:00 2028 GMT	[Unexpired, Bound]	

Figure 29 : vThunder instance 1 - Standby

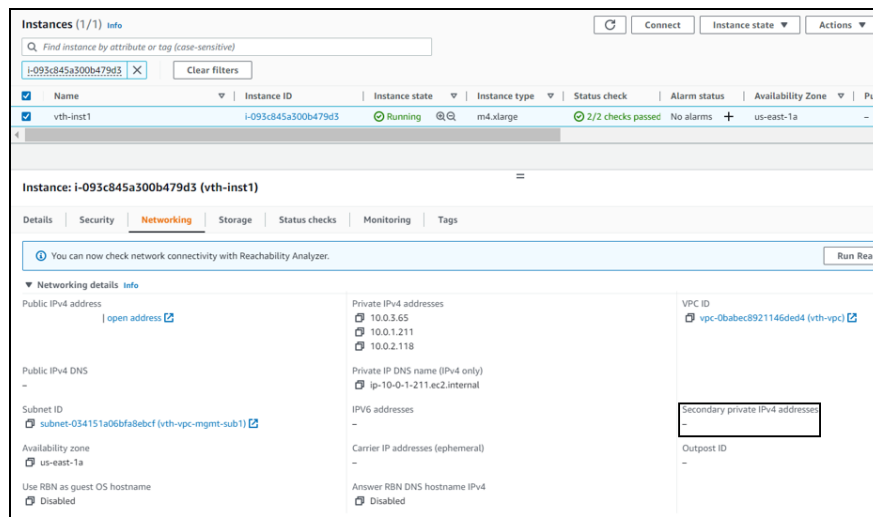
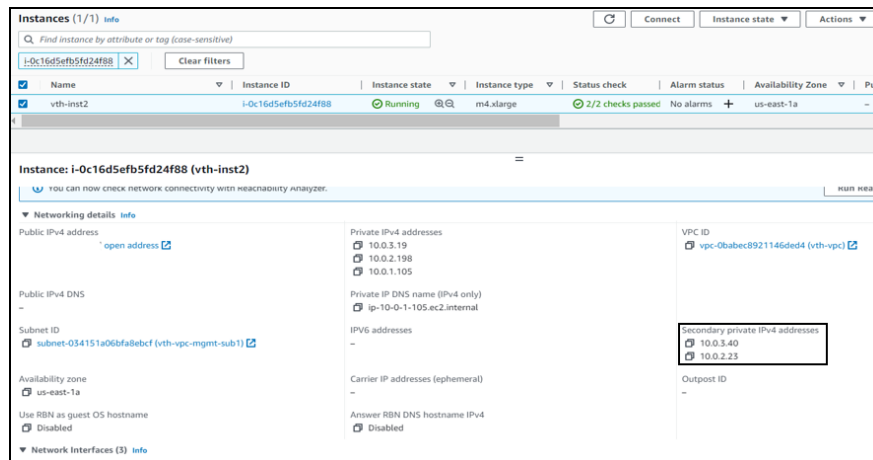


Figure 30 : vThunder instance 2 - Active



- If you want to make vThunder instance 1 active, run the following command on vThunder instance 2 using CLI:

```
vThunder-Active(config) (NOLICENSE) #vrrp-a force-self-standby disable
vThunder-Active(config) (NOLICENSE) #
vThunder-ForcedStandby(config) (NOLICENSE) #
```


At this point, IP switching occurs and the vThunder instance 1 prompt becomes:

```
vThunder-Active (config) (NOLICENSE) #
```

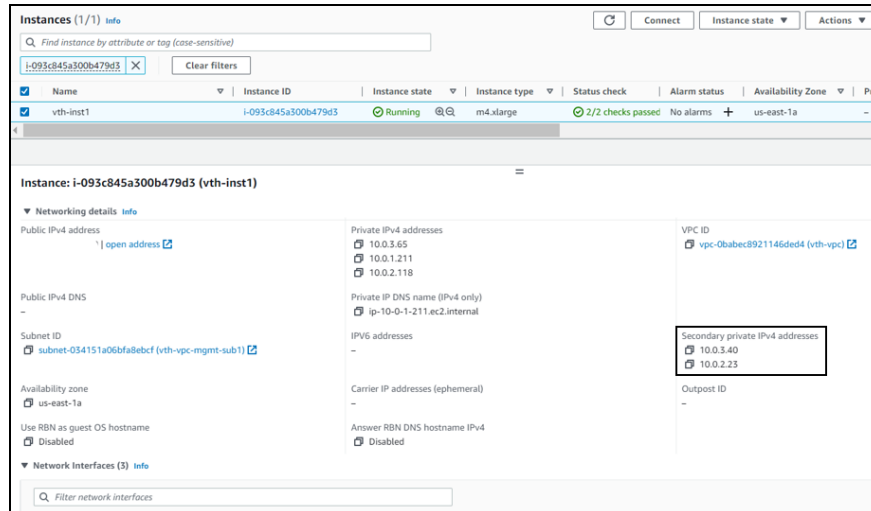
NOTE: If you want to access the vThunder instances using GUI, re-login the instances using the new admin user password and NOT the *EC2 Instance ID*.

Verify Traffic Flow

To verify the traffic flow from client machine to server machine via vThunder, perform the following:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select the active vThunder instance name and then click the **Networking** tab.
3. Copy the IP address of the vThunder Secondary IPv4 Address under the **Private IPv4 address**.

Figure 31 : vThunder instance 1



4. Select your client instance from the **Instances** list. Here, **vth-client** is the client instance name.
5. Click **Connect**. A **Connect to instance** window with **EC2 Instance Connect** tab is displayed.

6. Click **Connect**.

A **Terminal** window is displayed.

7. Run the following command in the Terminal window to send the traffic from the client machine:

```
curl <vThunder_instance_secondary_private_IPv4_Address>
```

Example

```
curl 10.0.2.23
```

8. Verify if a response is received.

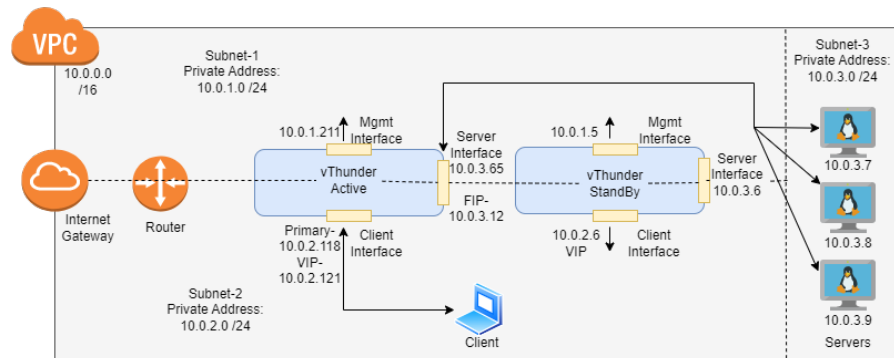
Deploy CFT A10-vThunder_ADC-3NIC-2VM-HA-GLM-PVTVIP

Using this template, you can deploy two vThunder instances containing:

- One management interface and two data interfaces for each vThunder instance
- High Availability (HA) support
- GLM integration

[Figure 32](#) illustrates the deployment topology for the 3NIC-2VM-HA-GLM-PVTVIP template.

Figure 32 : 3NIC-2VM-HA-GLM-PVTVIP Deployment Topology



The following topics are covered:

System Requirements	92
Supported Instance Types	93
Create vThunder Instances	96
Access vThunder using GUI or CLI	98
Configure Server and Client Machine	101
Configure FTP server	103
Configure vThunder as an SLB with HA	103
Verify Deployment	112
Verify GLM	119
Verify Traffic Flow	121

System Requirements

When the files from the CFT are downloaded and saved on your local machine, the resources and their default values are displayed as shown in [Table 9](#). You can configure the resources and modify these default values based on your deployment requirements.

Table 9 : System Requirements

Resource Name	Description	Default Value
Stack	A new stack is created with the specified name and location.	In this template, <code>vth</code> is used as the default stack name. Therefore, all the resources has <code>vth</code> as the prefix.
Network Interface Card [NIC]	Two types of interfaces are created for each vThunder instance: • One management interface • Two data interfaces	<code>vth-inst1-mgmt-nic1</code> <code>vth-inst1-data-nic2</code> <code>vth-inst1-data-nic3</code> <code>vth-inst2-mgmt-nic1</code> <code>vth-inst2-data-nic2</code> <code>vth-inst2-data-nic3</code>
Subnet	Three subnets are created with an address prefix each.	<code>vth-vpc-mgmt-sub1</code> <code>vth-vpc-data-sub1</code> <code>vth-vpc-data-sub2</code>
Virtual Private Network [VCN]	A virtual private network is assigned to the virtual machine instance.	<code>vth-vpc</code> Address prefix for virtual network: 10.0.0.0/16
Elastic Public IP	Two Elastic Public IP addresses are created and attached to the	<code>vth-inst1-mgmt-nic1-ip</code> <code>vth-inst2-mgmt-nic1-ip</code>

Resource Name	Description	Default Value
	management interfaces of the vThunder instances.	
Security Group	One security group is created for the management interface. One security group is created for the data interface. Logical name: - vThunderSecurityGroupMgmt - vThunderSecurityGroupData	Here, the tag names are: vth-sg-mgmt vth-sg-data
vThunder Instance	Two vThunder EC2 instances are created: - One active - One standby Default type: m4.xlarge (40 Gb memory) Table 10 lists the supported instance types.	vth-inst1 vth-inst2
Server Instance	One Ubuntu Server instance is created. Default Size: t2.micro	vth-server
FTP Server	One FTP server is created.	vth-ftp-server

Supported Instance Types

[Table 10](#) provides detailed information about the supported instance types.

Table 10 : List of Supported Instance Types

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

Instance	vCPU	Memory	Number of Network Interfaces
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

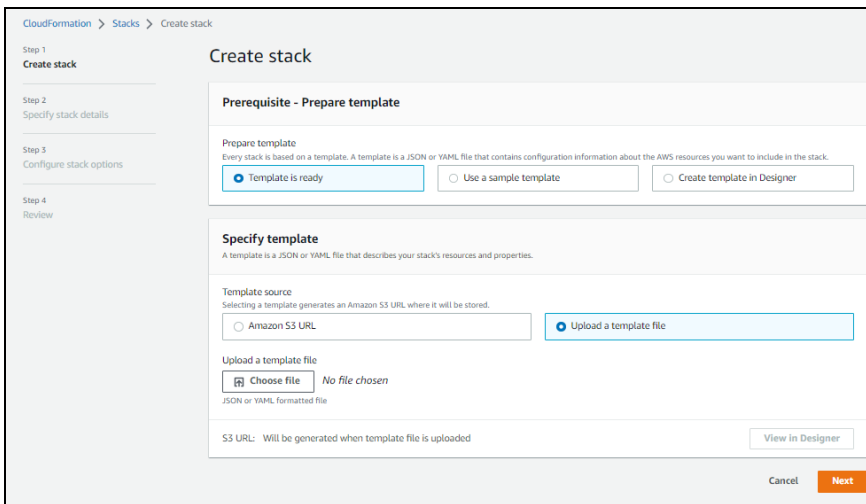
Instance	vCPU	Memory	Number of Network Interfaces
z1d.large	2	16384	3
z1d.xlarge	4	32768	4
z1d.2xlarge	8	65536	4
z1d.3xlarge	12	98304	8
z1d.6xlarge	24	196608	8

Create vThunder Instances

To create vThunder instances, perform the following steps:

1. From **AWS Management Console**, navigate to **CloudFormation > Stacks > Create Stack > With new resources (standard)**.
The **Create stack** window is displayed.

Figure 33 : Create stack window



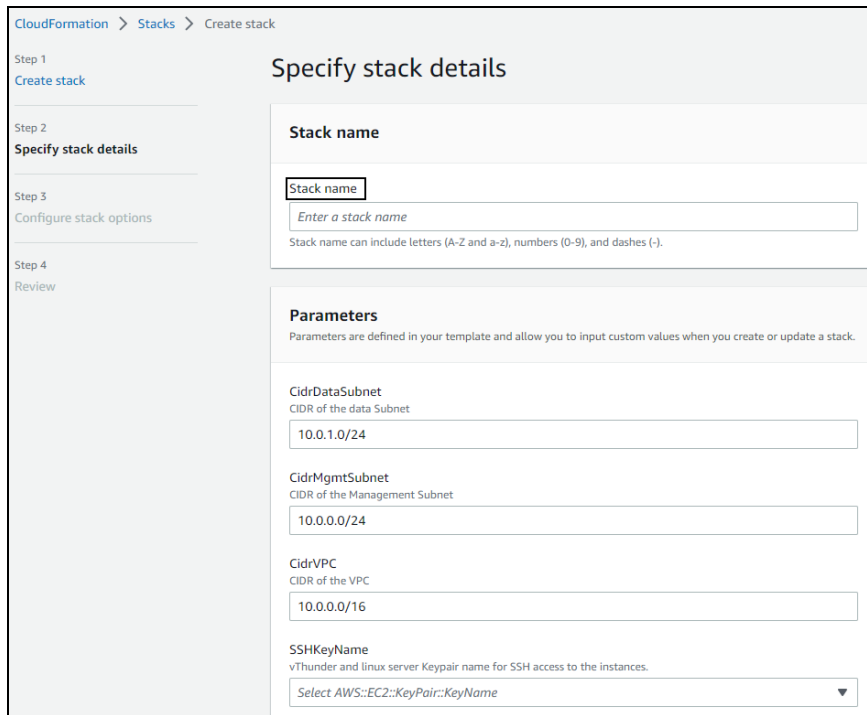
2. In the **Prerequisite - Prepare template** section, select **Template is ready**.
After the selecting this option, the **Specify template** section is displayed.
3. In the **Specify template** section, select **Upload a template file** and click **Choose file** to browse and upload the following template file from the downloaded CFT folder:
CFT_TMPL_3NIC_2VM_HA_GLM_PVTVIP_1.json

The selected template file name is displayed as the chosen file.

4. Click **Next**.

The **Specify stack details** window is displayed.

Figure 34 : Specify stack details window



5. In the **Specify stack details** window, enter or select the required values in the following fields:

a. In the **Stack name** section, enter a **Stack name**.

Here, **vth** is provided as the default stack name.

b. In the **Parameters** section, enter or select the following:

- **KeyPairName:** <your SSH key>
- **TagValue:** a10-vthunder-adc
- **Zone:** <your availability zone>

c. Verify the other fields and change the values appropriately. (Optional)

6. Click **Next**.

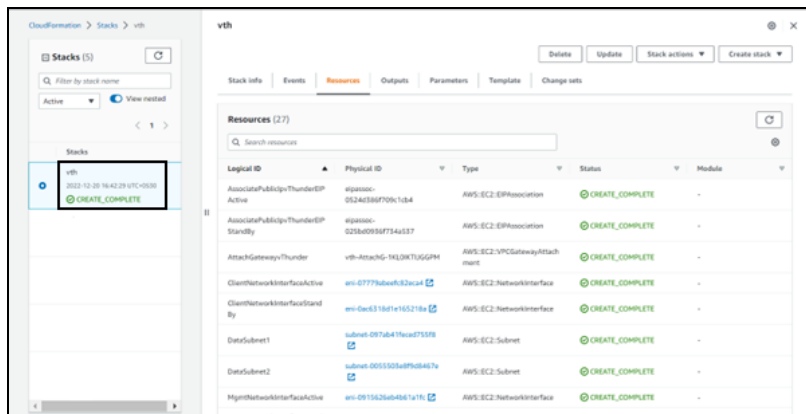
The **Configure stack options** window is displayed.

7. Verify the fields and change the values appropriately. (Optional)
8. Click **Next**.
The **Review** window is displayed.
9. Verify if all the stack configurations are correct and then click **Submit**.

NOTE: The system may take a few minutes to create the resources and display the stack status as **CREATE_COMPLETE**.

10. Verify if all the above resources are created in the **AWS Management Console** > **CloudFormation** > **Stacks** > <stack_name> > **Resources** tab.

Figure 35 : Resource listing in the resource group



The two vThunder instances are listed under **Resources** tab.

Access vThunder using GUI or CLI

vThunder instances can be accessed using any of the following ways:

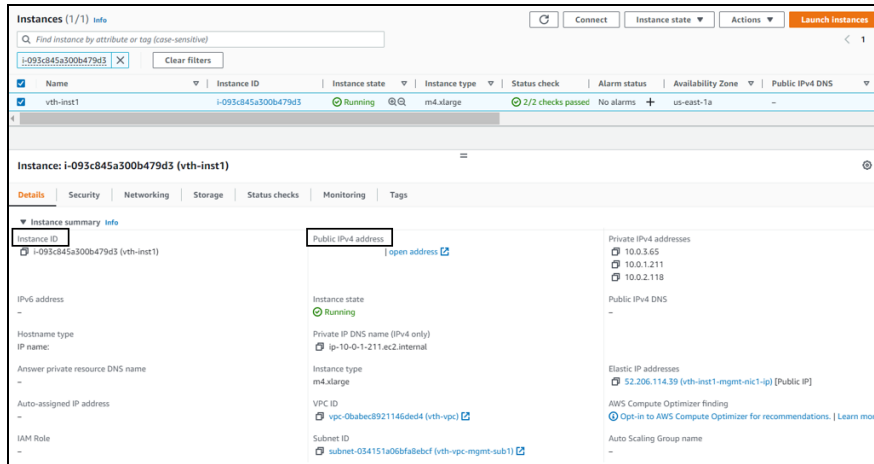
- [Access vThunder using GUI](#)
- [Access vThunder using CLI](#)

Access vThunder using GUI

To access vThunder instances using GUI, perform the following steps:

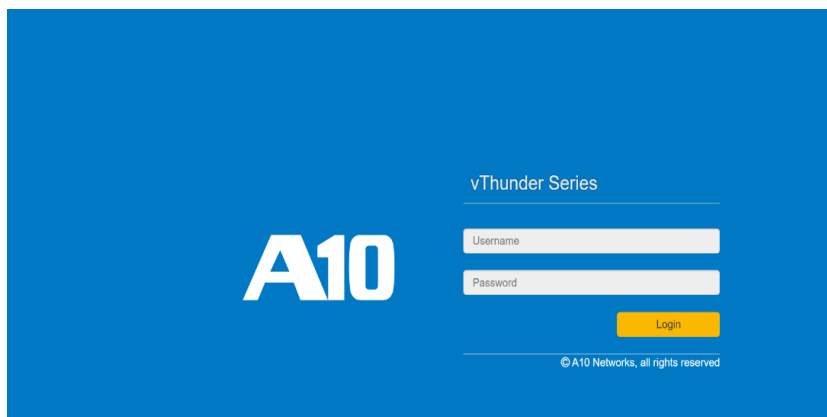
1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select your instance name.
Here, **vth-inst1**, **vth-inst2** are the vThunder instances.

Figure 36 : vThunder instance



3. For each vThunder instance, perform the following steps:
 - a. Copy the **Public IPv4 address** from the **Details** tab and replace the IP address in the below link:
http://<vThunder_public_IPv4_address>
 - b. Open the updated link in any browser.
The vThunder login window is displayed.

Figure 37 : vThunder GUI



c. Enter the following credentials and click **Login**:

- Username – **admin**
- Password – *EC2 Instance ID*

The home page is displayed if the entered credentials are correct.

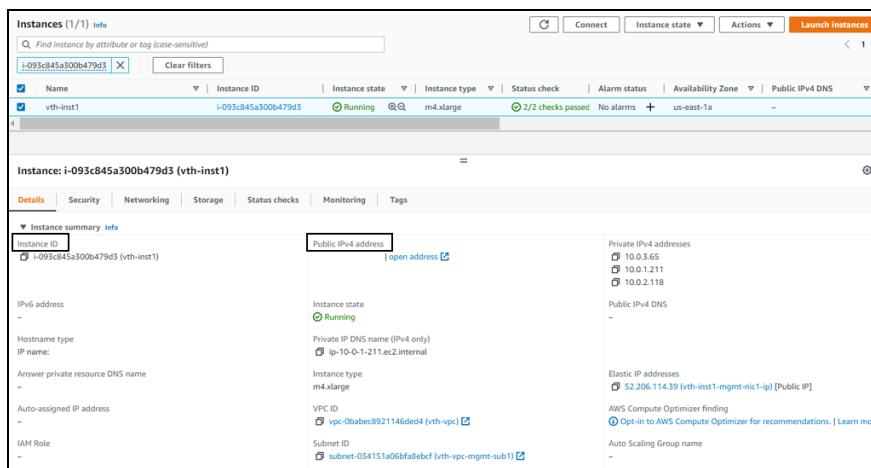
Access vThunder using CLI

To access vThunder instances using CLI, perform the following steps:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select your instance name.

Here, **vth-inst1**, **vth-inst2** are the vThunder instances.

Figure 38 : vThunder instance



3. For each vThunder instance, perform the following steps:
 - a. Copy the **Public IPv4 address** from the **Details** tab.
 - b. Open any SSH client and provide the following to establish a connection:
 - Hostname: Public IPv4 address
 - Username: admin
 - Key: SSH Key
 - c. Connect to the session.
 - d. In the SSH client session, run the following commands:

```
vThunder (NOLICENSE)>enable <---Execute command--->
Password: <---just press Enter key--->
vThunder (NOLICENSE) #config <---Configuration mode--->
vThunder (config) (NOLICENSE) #
```

The vThunder instances are ready to use.

Configure Server and Client Machine

To test the traffic flow via vThunder, create and configure a server machine and a client machine:

- [Configure a Server Machine](#)
- [Configure a Client Machine](#)

Configure a Server Machine

To configure a server machine, perform the following steps:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select your server instance name.
Here, **vth-server** is the server instance name.
3. Click **Connect**.
A **Connect to instance** window with **EC2 Instance Connect** tab is displayed.
4. Click **Connect**.
A **Terminal** window is displayed.
5. Run the following commands in the Terminal window to update all the package information:

```
sudo apt update
```

6. Run the following command in the Terminal window to create an Apache Server virtual machine:

```
sudo apt install apache2
```

While the Apache server is getting installed, you get a prompt to continue further. Enter 'Y' to continue. After the installation is complete, a newline prompt is displayed.

Configure a Client Machine

To configure a client machine, perform the following steps:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Click **Launch Instances**.
A **Launch an instance** window is displayed.
3. In the **Name and tags** section, enter an instance name.
Here, enter `vth-client` as the client instance name.
4. In the **Application and OS Images** section, select **Ubuntu**.
5. In the **Instance type** section, select the required instance type.
6. In the **Key pair (login)** field, select your SSH key.
7. In the **Network settings** section, click **Edit** to edit the following:
 - VPC: *your VPC*
Here, `vth-vpc` is the VPC.
 - Subnet: Data subnet 1
Here, `10.0.2.0/24` is the data subnet value.
 - Auto-assign public IP: Enable
 - Firewall (security groups): Select existing security group
 - Common security groups: *your data security group*
Here, `vth-vThunderSecurityGroupData` is the security group.
8. Click **Launch instance**.

NOTE: The system may take a few minutes to launch the instance.

The client instance is displayed in the **Instances** list with the status as **Running**.

Configure FTP server

In a High Availability (HA) configuration, IP switching occurs between two vThunder instances. The IP switching is enabled when AWS keys are imported on the vThunder instances. To import the AWS Access Keys, configure an FTP server.

To configure an FTP server, perform the following steps:

1. From **AWS Management Console**, navigate to **EC2 > Instances > <FTP_server_name>**.
2. Click **Launch Instances**.
A **Launch an instance** window is displayed.
3. Run the following command in the FTP server instance to create an Apache Server virtual machine:

```
sudo apt install apache2
```

While the Apache server is getting installed, you get a prompt to continue further. Enter 'Y' to continue. After the installation is complete, a newline prompt is displayed.

4. Run the following command in the same instance to open the **aws_access_key.txt** file:

```
sudo vi /var/www/html/aws_access_key.txt
```

5. Press **Esc** and enter **i** to enable edit/insert mode.
6. Locate and open the **credentials** file from downloaded CFT folder.
7. Copy the updated access key ID and secret access key as per your AWS account and paste in the **aws_access_key.txt** file. For more information, see [Prerequisites](#).
8. After the changes, press **Esc** then type **:wq** to save the changes and exit.

Configure vThunder as an SLB with HA

The following topics are covered:

- [Initial Setup](#)
- [Deploy vThunder as an SLB](#)
- [Verify Access Keys on vThunder Instances](#)

Initial Setup

Before deploying vThunder on AWS cloud as an SLB with HA, you need to configure the corresponding parameters in the CFT.

To configure the parameters, perform the following steps:

1. Navigate to the folder where you have downloaded the CFT, and open the CFT_TMPL_3NIC_2VM_HA_GLM_PVTVIP_CONFIG_SLB_SSL_HA_GLM_PARAM.json with a text editor.

NOTE: Each parameter has a default value mentioned in the parameter file.

2. Configure the stack name created using CFT.

```
"stackDetails": {
  "value": [
    {
      "stackName": "vth"
    }
  ]
},
```

3. Configure SLB server ports.

```
"server-list": {
  "value": [
    {
      "metadata": {
        "description": "SLB server host/fqdn-name."
      },
      "port-list": [
        {
          "port-number": 53,
          "protocol": "udp"
        }
      ]
    }
  ]
},
```



```
    {
      "port-number": 80,
      "protocol": "tcp"
    },
    {
      "port-number": 443,
      "protocol": "tcp"
    }
  ]
}
```

4. Configure Service Group ports.

The Service group name by default is “sg+port_number”. If you may want to change the service group name then after changing the name, change the names in the Virtual servers as well.

```
"serviceGroupList": {
  "value": [
    {
      "name": "sg443",
      "protocol": "tcp",
      "member-list": [
        {
          "port": 443
        }
      ]
    },
    {
      "name": "sg53",
      "protocol": "udp",
      "member-list": [
        {
          "port": 53
        }
      ]
    }
  ]
},
```

```
{
  "name": "sg80",
  "protocol": "tcp",
  "member-list": [
    {
      "port": 80
    }
  ]
}
```

5. Configure a Virtual Server and its ports.

The virtual server default name is “vip”. It is the Private IP address of Ethernet1.

```
"virtualServerList": {
  "virtual-server-name": "vip",
  "metadata": {
    "description": "virtual server is using ethernet 1 ip
address"
  },
  "value": [
    {
      "port-number": 53,
      "protocol": "udp",
      "auto": 1,
      "service-group": "sg53"
    },
    {
      "port-number": 80,
      "protocol": "http",
      "auto": 1,
      "service-group": "sg80"
    },
    {
      "port-number": 443,
      "protocol": "https",
      "auto": 1,
```

```
        "service-group": "sg443"
      }
    ]
  },
```

6. Configure DNS.

```
  "dns": {
    "value": "8.8.8.8"
  },
```

7. Configure a Network Gateway IP.

The default value of network gateway IP address is 10.0.1.1 as this is the first IP address of the data subnet 1 configuration.

```
  "rib-list": [
    {
      "ip-dest-addr": "0.0.0.0",
      "ip-mask": "/0",
      "ip-next-hop-ipv4": [
        {
          "ip-next-hop": "10.0.1.1"
        },
        {
          "ip-next-hop": "10.0.2.1"
        }
      ]
    }
  ],
```

8. Set VRRP-A.

```
  "vrrp-a": {
    "set-id": 1
  },
```

9. Set a Terminal Idle Timeout.

```
  "terminal": {
    "idle-timeout": 0
  },
```

10. Set VRID.

For vThunder instance 1, the default priority is 100, and for vThunder instance 2, it is 99 (100-1).

```
    "vrid-list": [  
      {  
        "vrid-val": 0,  
        "blade-parameters": {  
          "priority": 100  
        }  
      }  
    ],
```

11. Configure SSL.

```
"sslConfig": {  
  "requestTimeout": 40,  
  "Path": "server.pem",  
  "File": "server",  
  "CertificationType": "pem"  
}
```

NOTE: By default, SSL configuration is disabled i.e., no SSL configuration is applied.

The server.pem file is included in the downloaded CFT folder. If necessary, you can edit this file or use a different certificate file. When you use a different certificate file, remember to change the path accordingly.

12. Configure your GLM account details.

To obtain the **Entitlement Token**, go to [GLM](#) > **Licenses** > select your license > **Overview** > **Info** tab.

```
"user_name": {  
  "value": "xxxxxxxx@a10networks.com"  
},  
"user_password": {  
  "value": "xxxxxxxx"
```

```

    },
    "entitlement_token": {
        "value": "xxxxxxxx"
    }
}

```

13. Verify if all the configurations in the CFT_TMPL_3NIC_2VM_HA_GLM_PVTVIP_CONFIG_SLB_SSL_HA_GLM_PARAM.json file are correct and then save the changes.

Deploy vThunder as an SLB

To deploy vThunder on AWS cloud as an SLB, perform the following steps:

1. From your command prompt, navigate to the folder where you have downloaded the CFT.
2. Run the following command to create vThunder SLB instance:

```
python ./CFT_TMPL_3NIC_2VM_HA_GLM_PVTVIP_CONFIG_SLB_SSL_HA_GLM_2.py
```

A message is prompted to upload the SSL certificate and configure GLM and HA.

3. Enter **yes** if you want to upload the SSL certificate and configure GLM and HA.

```

Do you want to upload ssl certificate(yes/no)?yes
Do you want to configure GLM(yes/no)?yes
Do you want to configure HA(yes/no)?yes

```

A message is prompted that displays the criteria for a valid admin user password.

Primary conditions for password validation, user should provide the new password according to the given combination:

```

Minimum length of 9 characters
Minimum lowercase character should be 1
Minimum uppercase character should be 1
Minimum number should be 1
Minimum special character should be 1
Should not include repeated characters
Should not include more than 3 keyboard consecutive characters.

```

A message is prompted to enter a new password.

4. Provide the new password for the admin user and then confirm the new password.

The new password should follow the valid admin user password criteria mentioned in the above step. For more information, see [Default Password Policy](#).

```
Enter vThunder's new password:
Confirm new password:
```

If both the password and confirm password field values match and complies to the valid admin user password criteria, a message 'Password changed successfully' is displayed along with the other configurations.

```
Password changed successfully.
-----
Configuring vThunder with instance id i-0d7677775bb2c6f94
AWS access key uploaded to vThunder.
configured ethernet ip
configured ethernet ip
Configured server vth-server
Configure service group
Configured virtual servers
SSL Configured.
License activation completed.
configure glm.
Glm license request sent successfully.
Configured primary dns
Configured IP Route
Configured Vrrp A common
Configured idle timeout
Configured vrrp rid
Configured peer group
Configurations are saved on partition: shared
-----
Configuring vThunder with instance id i-00263bc7892bdde2f
AWS access key uploaded to vThunder.
```

```
configured ethernet ip
configured ethernet ip
Configured server vth-server
Configure service group
Configured virtual servers
SSL Configured.
License activation completed.
configure glm.
Glm license request sent successfully.
Configured primary dns
Configured IP Route
Configured Vrrp A common
Configured idle timeout
Configured vrrp rid
Configured peer group
Configurations are saved on partition: shared
```

Verify Access Keys on vThunder Instances

To verify if the access keys are uploaded on both vThunder instances, perform the following step:

1. Run the following commands on each vThunder instance:

```
vThunder(config)#admin admin
vThunder(config-admin:admin)#aws-accesskey show
[default]
aws_access_key_id = your_aws_access_key_id
aws_secret_access_key = your_aws_secret_access_key
vThunder(config-admin:admin)#exit
vThunder(config)#
vThunder(config)#
vThunder-Active(config)#
```

The access keys are displayed.

NOTE: Once the AWS access keys are imported to vThunder, the FTP server gets deleted automatically.

Verify Deployment

To verify vThunder SLB deployment using CFT, perform the following steps:

1. Access the vThunder instances using CLI.
2. Run the following command on vThunder instance 1:

```
vThunder-Active(config)#show running-config
```

If the deployment is successful, the following SLB configuration is displayed:

```
!Current configuration: 349 bytes
!Configuration last updated at 10:56:58 GMT Fri Jan 6 2023
!Configuration last saved at 10:53:34 GMT Fri Jan 6 2023
!64-bit Advanced Core OS (ACOS) version 5.2.1, build 153 (Dec-11-
2020,16:36)
!
vrrp-a common
  device-id 1
  set-id 1
  enable
!
terminal idle-timeout 0
!
ip dns primary 8.8.8.8
!
!
glm use-mgmt-port
glm enable-requests
glm token vTh205fe920b
!
interface ethernet 1
  enable
  ip address dhcp
!
interface ethernet 2
  enable
  ip address dhcp
```



```
!  
vrrp-a vrid 0  
    floating-ip 10.0.3.12  
    blade-parameters  
        priority 99  
!  
vrrp-a peer-group  
    peer 10.0.2.117  
    peer 10.0.2.173  
!  
ip route 0.0.0.0 /0 10.0.1.1  
ip route 0.0.0.0 /0 10.0.2.1  
!  
slb server vth-server 10.0.3.23  
    port 53 udp  
    port 80 tcp  
    port 443 tcp  
!  
slb service-group sg443 tcp  
    member vth-server 443  
!  
slb service-group sg53 udp  
    member vth-server 53  
!  
slb service-group sg80 tcp  
    member vth-server 80  
!  
slb virtual-server vip 10.0.2.23  
    port 53 udp  
        source-nat auto  
        service-group sg53  
    port 80 http  
        source-nat auto  
        service-group sg80  
    port 443 https  
        source-nat auto  
        service-group sg443
```

```
!
!
end
!Current config commit point for partition 0 is 0 & config mode is
classical-mode
vThunder-Active(config)#
```

3. Run the following command on vThunder instance 1:

```
vThunder-Active(config)#show pki cert
```

If the deployment is successful, the following SSL configuration is displayed:

Name	Type	Expiration	Status
server certificate	Jan 28 12:00:00 2028 GMT	[Unexpired, Bound]	

At this point, the vThunder instance 2 has the following prompt:

```
vThunder-Standby(config)#
```

Figure 39 : vThunder instance 1 - Active

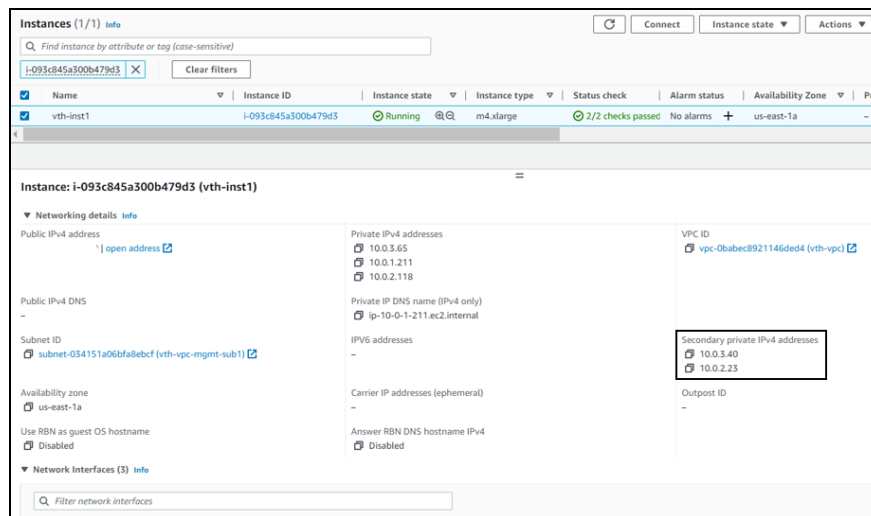
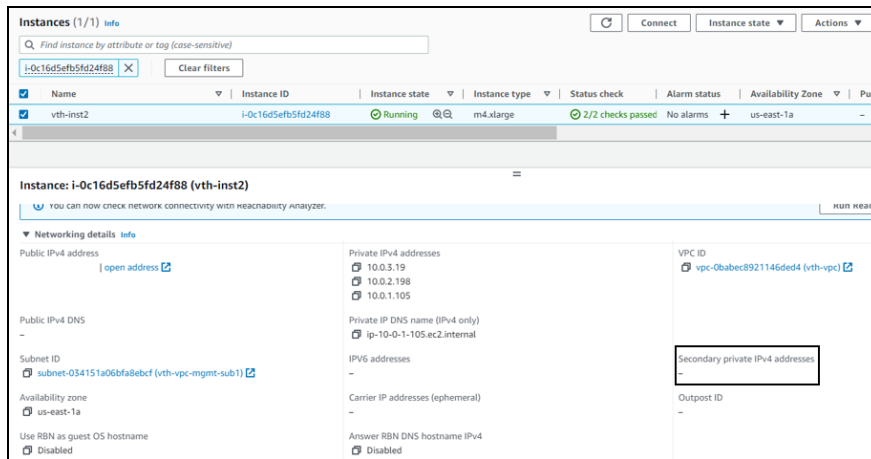


Figure 40 : vThunder instance 2 - Standby



4. Run the following command on vThunder instance 1:

```
vThunder-Active(config)#vrrp-a force-self-standby enable
vThunder-Active(config)#
vThunder-ForcedStandby(config)#
```

At this point, IP switching occurs and the vThunder instance 2 prompt becomes:

```
vThunder-Active(config)#
```

5. Run the following command on vThunder instance 2:

```
vThunder-Active(config)#show running-config
```

If the deployment is successful, the following SLB configuration is displayed:

```
!Current configuration: 282 bytes
!Configuration last updated at 10:53:35 GMT Fri Jan 6 2023
!Configuration last saved at 10:53:37 GMT Fri Jan 6 2023
!64-bit Advanced Core OS (ACOS) version 5.2.1, build 153 (Dec-11-
2020,16:36)
!
vrrp-a common
  device-id 2
  set-id 1
  enable
!
terminal idle-timeout 0
```

```
!  
ip dns primary 8.8.8.8  
!  
!  
glm use-mgmt-port  
glm enable-requests  
glm token vTh205fe920b  
!  
interface ethernet 1  
    enable  
    ip address dhcp  
!  
interface ethernet 2  
    enable  
    ip address dhcp  
!  
vrrp-a vrid 0  
    floating-ip 10.0.3.12  
    blade-parameters  
        priority 98  
!  
vrrp-a peer-group  
    peer 10.0.2.186  
    peer 10.0.2.6  
!  
ip route 0.0.0.0 /0 10.0.1.1  
ip route 0.0.0.0 /0 10.0.2.1  
!  
slb server vth-server 10.0.3.23  
    port 53 udp  
    port 80 tcp  
    port 443 tcp  
!  
slb service-group sg443 tcp  
    member vth-server 443  
!  
slb service-group sg53 udp
```

```

member vth-server 53
!
slb service-group sg80 tcp
    member vth-server 80
!
slb virtual-server vip 10.0.2.121
    port 53 udp
        source-nat auto
        service-group sg53
    port 80 http
        source-nat auto
        service-group sg80
    port 443 https
        source-nat auto
        service-group sg443
!
sflow setting local-collection
!
sflow collector ip 127.0.0.1 6343
!
!
end
!Current config commit point for partition 0 is 0 & config mode is
classical-mode
vThunder-Active(config)#

```

6. Run the following command on vThunder instance 2:

```
vThunder-Active(config)#show pki cert
```

If the deployment is successful, the following SSL configuration is displayed:

Name	Type	Expiration	Status

server certificate	Jan 28 12:00:00 2028 GMT	[Unexpired, Bound]	

Figure 41 : vThunder instance 1 - Standby

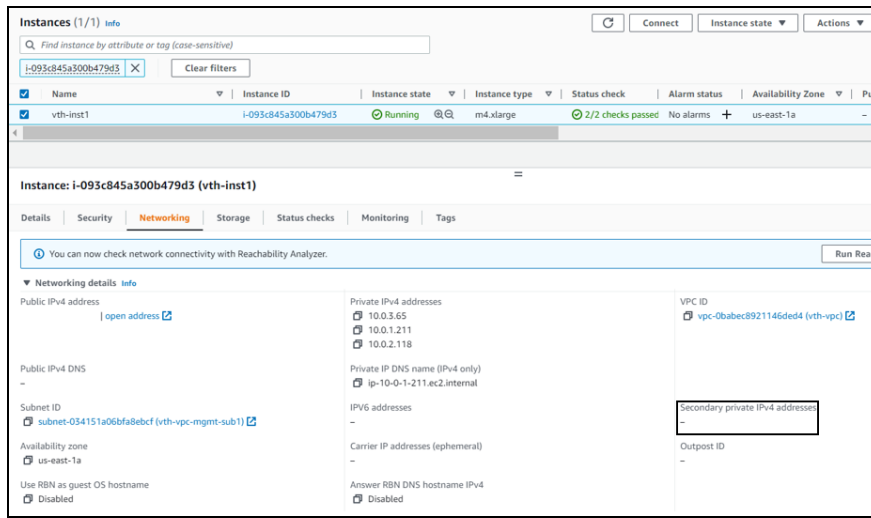
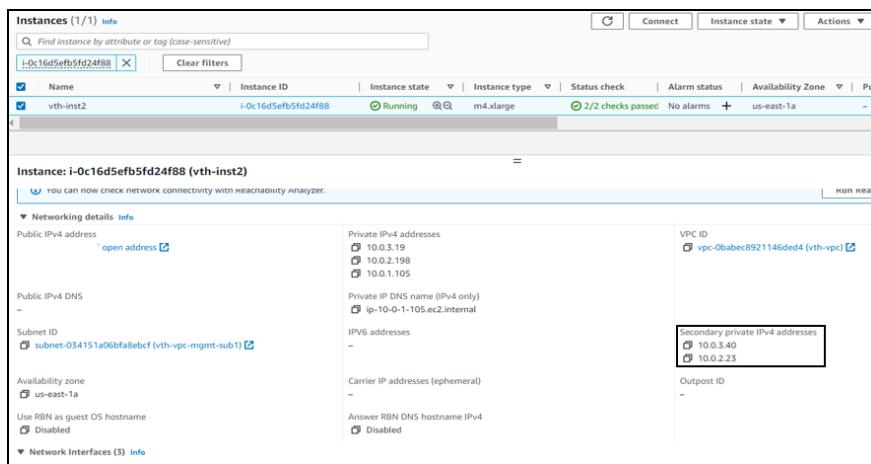


Figure 42 : vThunder instance 2 - Active



- If you want to make vThunder instance 1 active, run the following command on vThunder instance 2:

```
vThunder-Active(config)#vrrp-a force-self-standby disable
vThunder-Active(config)#
vThunder-ForcedStandby(config)#
```

At this point, IP switching occurs and the vThunder instance 1 prompt becomes:

```
vThunder-Active(config)#
```

NOTE: If you want to access the vThunder instances using GUI, re-login the instances using the new admin user password and NOT the *EC2 Instance ID*.

Verify GLM

The application of license can be verified using any of the following ways:

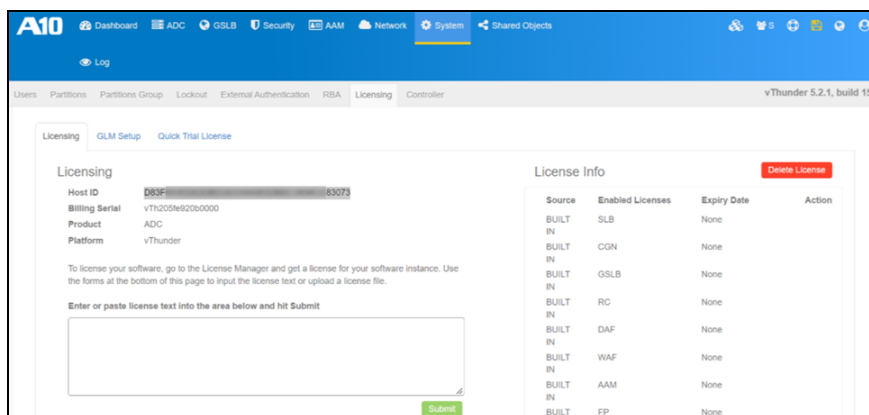
- [Verify License using GUI](#)
- [Verify License using CLI](#)

Verify License using GUI

To verify license using GUI, perform the following steps:

1. Log in to your vThunder instance using Public IPv4 address.
Here, `vth-inst1` is the vThunder instance.
2. Navigate to **Profile > Setting > Licensing**.
3. Click the **Licensing** tab.
If the license is successfully applied on vThunder, the Host ID is displayed.

Figure 43 : GLM > Licensing window



Verify License using CLI

To verify license using CLI, perform the following steps:

1. Verify if NOLICENSE is removed from the vThunder prompt.
2. Run the following command on vThunder:

```
vThunder(config)#show license
```

If the license is successfully applied on vThunder, the Host ID is displayed:

```
Host ID      : D83F****82EB633D****067DB84136****83073
```

3. Run the following command on vThunder instance:

```
vThunder(config)#show license-info
```

If the license is successfully applied on vThunder, the following GLM configuration is displayed:

```
Host ID      : D83F****82EB633D****067DB84136****83073
USB ID       : Not Available
Billing Serials: vTh205fe920b0000
Token        : Not Available
Product      : ADC
Platform     : vThunder
Burst        : Disabled
GLM Ping Interval In Hours : 24
```

```
-----
Enabled Licenses   Expiry Date (UTC)           Notes
-----
SLB                None
CGN                None
GSLB               None
RC                 None
DAF                None
WAF                None
AAM                None
FP                 None
WEBROOT            N/A                Requires an additional Webroot
license.
THREATSTOP         N/A                Requires an additional ThreatSTOP
license.
```

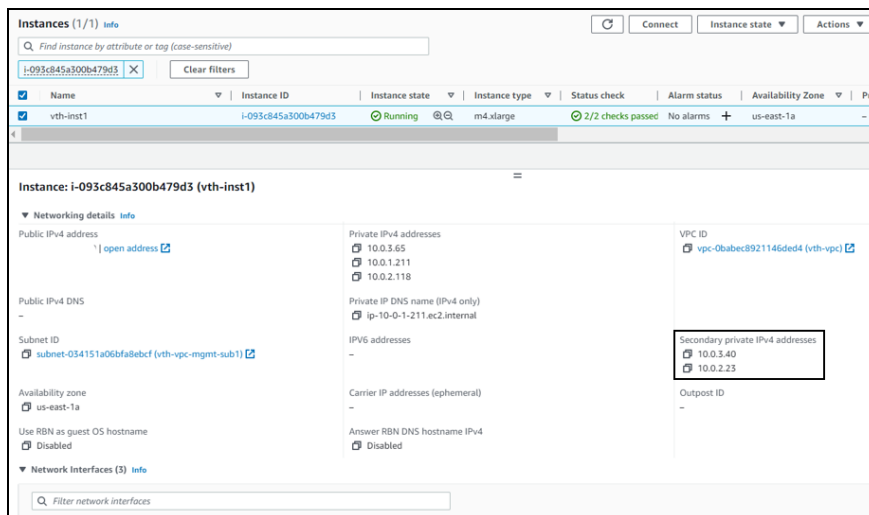

QOSMOS	N/A	Requires an additional QOSMOS license.
WEBROOT_TI	N/A	Requires an additional Webroot Threat Intel license.
IPSEC_VPN	N/A	Requires an additional IPsec VPN license.
500 Mbps Bandwidth 20-January-2023		

Verify Traffic Flow

To verify the traffic flow from client machine to server machine via vThunder, perform the following:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select the active vThunder instance name and then click the **Networking** tab.
3. Copy the IP address of the vThunder Secondary IPv4 Address under the **Private IPv4 address**.

Figure 44 : vThunder instance 1



4. Select your client instance from the **Instances** list.
Here, **vth-client** is the client instance name.
5. Click **Connect**.
A **Connect to instance** window with **EC2 Instance Connect** tab is displayed.

6. Click **Connect**.
A **Terminal** window is displayed.
7. Run the following command in the Terminal window to send the traffic from the client machine:

```
curl <vThunder_instance_secondary_private_IPv4_Address>
```

Example

```
curl 10.0.2.23
```

8. Verify if a response is received.

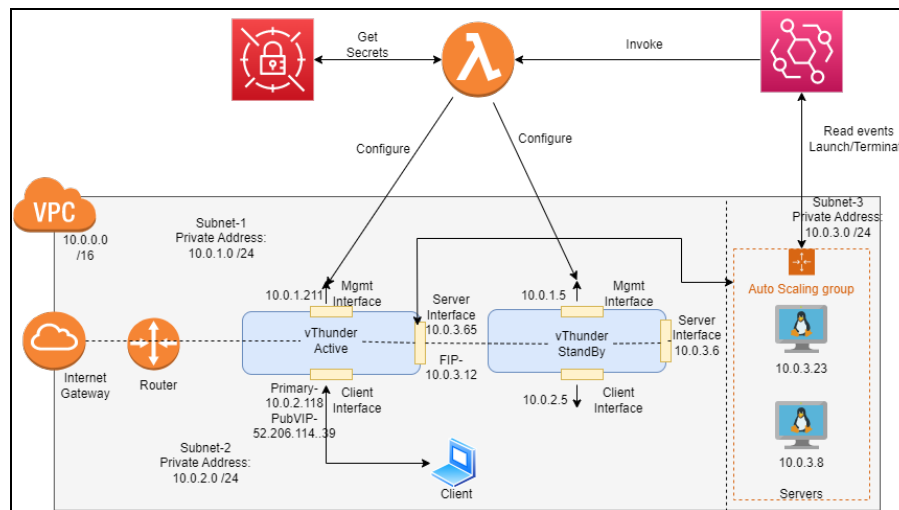
Deploy CFT A10-vThunder_ADC-3NIC-2VM-HA-GLM-PUBVIP-BACKAUTO

Using this template, you can deploy two vThunder instances containing:

- One management interface and two data interfaces for each vThunder instance
- High Availability (HA) support
- GLM integration
- Backend server autoscaling support.

[Figure 45](#) illustrates the deployment topology for the 3NIC-2VM-HA-GLM-PUBVIP-BACKAUTO template.

Figure 45 : 3NIC-2VM-HA-GLM-PUBVIP-BACKAUTO Deployment Topology



The following topics are covered:

System Requirements	124
Supported Instance Types	126
Create S3 Bucket	128
Create vThunder Instances	129
Access vThunder using GUI or CLI	133
Configure Server and Client Machine	136

Configure FTP server	137
Configure vThunder as an SLB with HA	138
Verify Deployment	146
Verify GLM	153
Verify Traffic Flow	155
On-demand Password Change	157

System Requirements

When the files from the CFT are downloaded and saved on your local machine, the resources and their default values are displayed as shown in [Table 11](#). You can configure the resources and modify these default values based on your deployment requirements.

Table 11 : System Requirements

Resource Name	Description	Default Value
Stack	A new stack is created with the specified name and location.	In this template, <code>vth</code> is used as the default stack name. Therefore, all the resources has <code>vth</code> as the prefix.
Network Interface Card [NIC]	Two types of interfaces are created for each vThunder instance: - One management interface - Two data interfaces	<code>vth-inst1-mgmt-nic1</code> <code>vth-inst1-data-nic2</code> <code>vth-inst1-data-nic3</code> <code>vth-inst2-mgmt-nic1</code> <code>vth-inst2-data-nic2</code> <code>vth-inst2-data-nic3</code>
Subnet	Three subnets are created with an address prefix each.	<code>vth-vpc-mgmt-sub1</code> <code>vth-vpc-data-sub1</code>

Resource Name	Description	Default Value
		<code>vth-vpc-data-sub2</code>
Virtual Private Network [VCN]	A virtual private network is assigned to the virtual machine instance.	<code>vth-vpc</code> Address prefix for virtual network: <code>10.0.0.0/16</code>
Elastic Public IP	Three Elastic Public IP addresses are created and attached to the management and data interfaces of the vThunder instances.	<code>vth-inst1-mgmt-nic1-ip</code> <code>vth-inst1-data-nic1-ip</code> <code>vth-inst2-mgmt-nic1-ip</code>
Security Group	Two security groups are created: - One security group for the management interface. - One security group for the data interface. Logical name: <code>vThunderSecurityGroupMgmt</code> <code>vThunderSecurityGroupData</code>	Here, the tag names are: <code>vth-sg-mgmt</code> <code>vth-sg-data</code>
vThunder Instance	Two vThunder EC2 instances are created: - One active - One standby Default type: <code>m4.xlarge</code> (40 Gb memory) Table 12 lists the supported instance	<code>vth-inst1</code> <code>vth-inst2</code>

Resource Name	Description	Default Value
	types.	
Lambda Function	A Lambda function is created, and the code is uploaded from S3 bucket.	vth-lambda-function
EventBus	An EventBus is created and event rules are added.	vth-eventbus
AutoScaling Group	An autoscaling group is created to add or delete instances as needed.	vth-auto-scale-group
FTP Server	One FTP server is created.	vth-ftp-server

Supported Instance Types

[Table 12](#) provides detailed information about the supported instance types.

Table 12 : List of Supported Instance Types

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

Instance	vCPU	Memory	Number of Network Interfaces
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

Instance	vCPU	Memory	Number of Network Interfaces
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

Create S3 Bucket

Amazon Web Services' (AWS) Simple Storage Service (S3) bucket is used to store the template resources. Therefore, you need to create the S3 bucket and then upload the SLB configuration file on AWS using this S3 bucket.

To create an S3 bucket, perform the following steps:

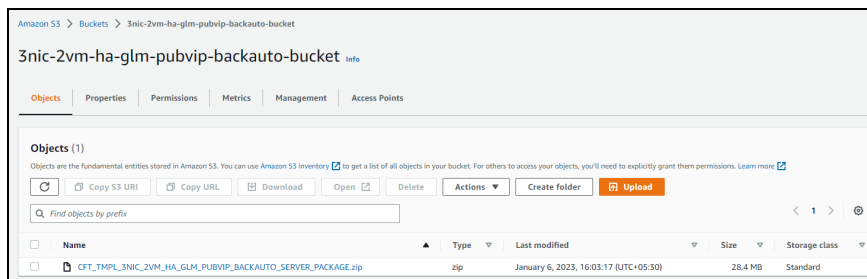
1. From your command prompt, navigate to the folder where you have downloaded the CFT.
2. Run the following command to create an S3 bucket and store the Lambda function python script:

```
python ./CFT_TMPL_3NIC_2VM_HA_GLM_PUBVIP_BACKAUTO_SERVER_PACKAGE_S3_1.py
```

A message 'File uploaded in S3 bucket successfully' is displayed.

3. Navigate to **AWS Management Console > Buckets > <bucket_name>** to verify if the 3nic-2vm-ha-glm-pubvip-backauto-bucket is created and CFT_TMPL_3NIC_2VM_HA_GLM_PUBVIP_BACKAUTO_SERVER_PACKAGE.zip is uploaded.

Figure 46 : S3 Bucket

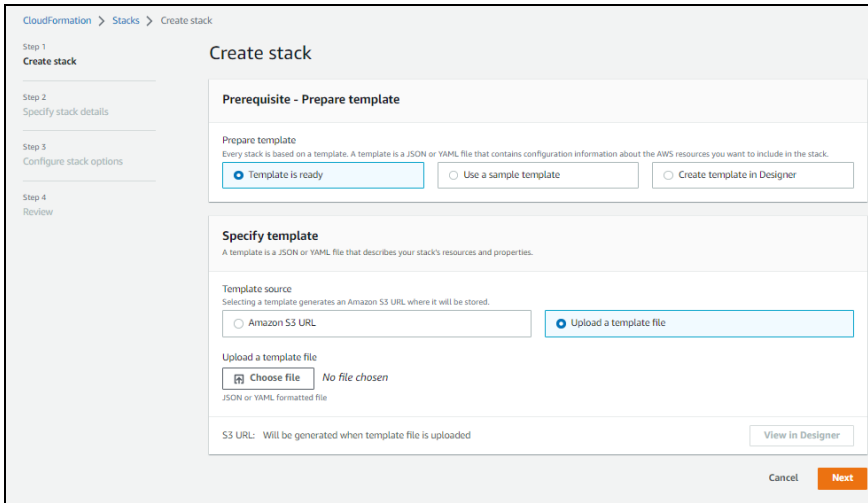


Create vThunder Instances

To create vThunder instances, perform the following steps:

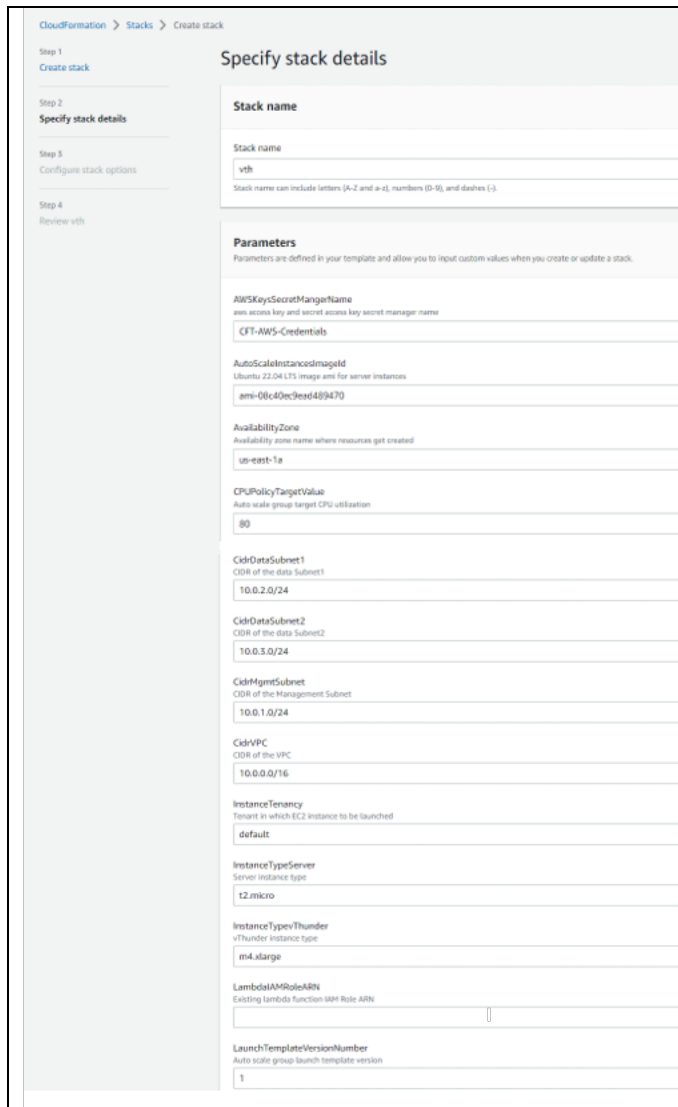
1. From **AWS Management Console**, navigate to **CloudFormation > Stacks > Create Stack > With new resources (standard)**. The **Create stack** window is displayed.

Figure 47 : Create stack window



2. In the **Prerequisite - Prepare template** section, select **Template is ready**. After the selecting this option, the **Specify template** section is displayed.
3. In the **Specify template** section, select **Upload a template file** and click **Choose file** to browse and upload the following template file from the downloaded CFT folder:
CFT_TMPL_3NIC_2VM_HA_GLM_PUBVIP_BACKAUTO_2.json
The selected template file name is displayed as the chosen file.
4. Click **Next**.
The **Specify stack details** window is displayed.

Figure 48 : Specify stack details window



CloudFormation > Stacks > Create stack

Step 1
Create stack

Step 2
Specify stack details

Step 3
Configure stack options

Step 4
Review vth

Specify stack details

Stack name

Stack name

vth

Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-).

Parameters

Parameters are defined in your template and allow you to input custom values when you create or update a stack.

AWSKeysSecretManagerName
aws access key and secret access key secret manager name

CFT-AWS-Credentials

AutoScaleInstanceImageId
Ubuntu 22.04 LTS image ami for server instances

ami-06c40e3ed489470

AvailabilityZone
Availability zone name where resources get created

us-east-1a

CPUPolicyTargetValue
Auto scale group target CPU utilization

80

CidrDataSubnet1
CIDR of the data Subnet1

10.0.2.0/24

CidrDataSubnet2
CIDR of the data Subnet2

10.0.3.0/24

CidrMgmtSubnet
CIDR of the Management Subnet

10.0.1.0/24

CidrVPC
CIDR of the VPC

10.0.0.0/16

InstanceTenancy
Tenant in which EC2 instance to be launched

default

InstanceTypeServer
Server instance type

t2.micro

InstanceTypevThunder
vThunder instance type

m4.xlarge

LambdaIAMRoleARN
Existing lambda function IAM Role ARN

LaunchTemplateVersionNumber
Auto scale group launch template version

1

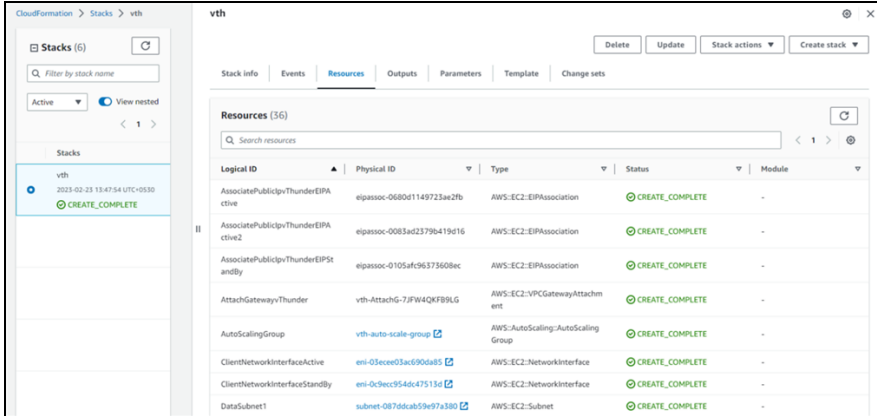
5. In the **Specify stack details** window, enter or select the following:
 - a. In the **Stack name** section, enter a **Stack name**.
Here, **vth** is provided as the default stack name.
 - b. In the **Parameters** section, enter or select the required values in the following fields:
 - **AWSKeysSecretManagerName:** <your AWS keys secret manager name>
 - **KeyPairName:** <your SSH key>

- **LambdaIAMRoleARN:** *<your Lambda function execution role ARN>*
 - **TagValue:** **a10-vthunder-adc**
 - **Zone:** *<your availability zone>*
- c. Verify the other fields and change the values appropriately. (Optional)
- Click **Next**.
The **Configure stack options** window is displayed.
 - Verify the fields and change the values appropriately. (Optional)
 - Click **Next**.
The **Review** window is displayed.
 - Verify if all the stack configurations are correct and then click **Submit**.

NOTE: The system may take a few minutes to create the resources and display the stack status as **CREATE_COMPLETE**.

- Verify if all the above resources are created in the **AWS Management Console > CloudFormation > Stacks > <stack_name> > Resources** tab.

Figure 49 : Resource listing in the resource group

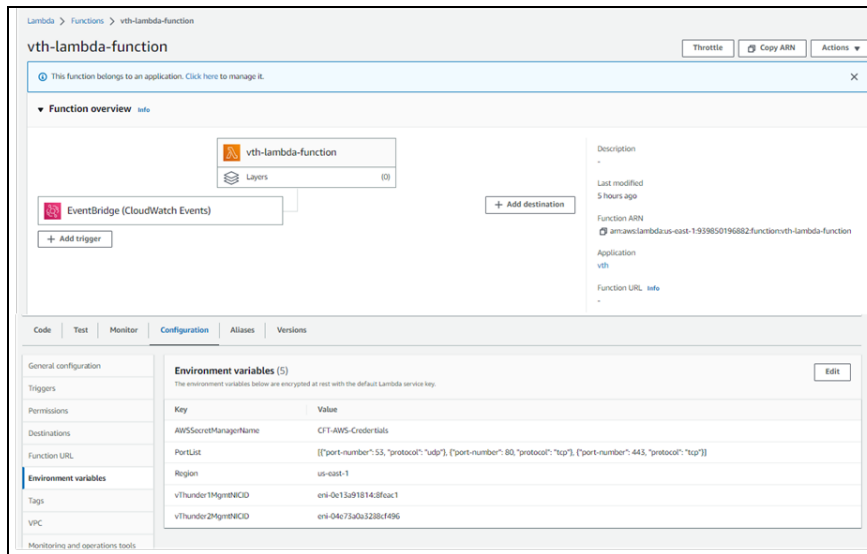


Logical ID	Physical ID	Type	Status	Module
AssociatePublicIpThunderEIPActive	eipassoc-0680d1149723ae2fb	AWS::EC2::EIPAssociation	CREATE_COMPLETE	-
AssociatePublicIpThunderEIPActive2	eipassoc-0083ed2379b419d16	AWS::EC2::EIPAssociation	CREATE_COMPLETE	-
AssociatePublicIpThunderEIPStandby	eipassoc-0105af96373608ec	AWS::EC2::EIPAssociation	CREATE_COMPLETE	-
AttachGatewayvThunder	vth-AttachG-7JFW4QF89LG	AWS::EC2::VPCEGatewayAttachment	CREATE_COMPLETE	-
AutoScalingGroup	vth-auto-scale-group	AWS::AutoScaling::AutoScalingGroup	CREATE_COMPLETE	-
ClientNetworkInterfaceActive	eni-03eece03ae6906ae5	AWS::EC2::NetworkInterface	CREATE_COMPLETE	-
ClientNetworkInterfaceStandby	eni-0c9ec954dc47513d	AWS::EC2::NetworkInterface	CREATE_COMPLETE	-
DataSubnet1	subnet-0876dab59d97a380	AWS::EC2::Subnet	CREATE_COMPLETE	-

The two vThunder instances are listed under **Resources** tab.

- Verify if the environment variables of the Lambda function are created in **AWS Management Console > Lambda > Functions > <function_name> > Configuration > Environment variables** tab.

Figure 50 : Lambda Function

**NOTE:**

If you delete a stack that contains a Lambda function, the Lambda function will not be deleted automatically along with the stack. You need to manually delete the Lambda function separately.

Access vThunder using GUI or CLI

vThunder instances can be accessed using any of the following ways:

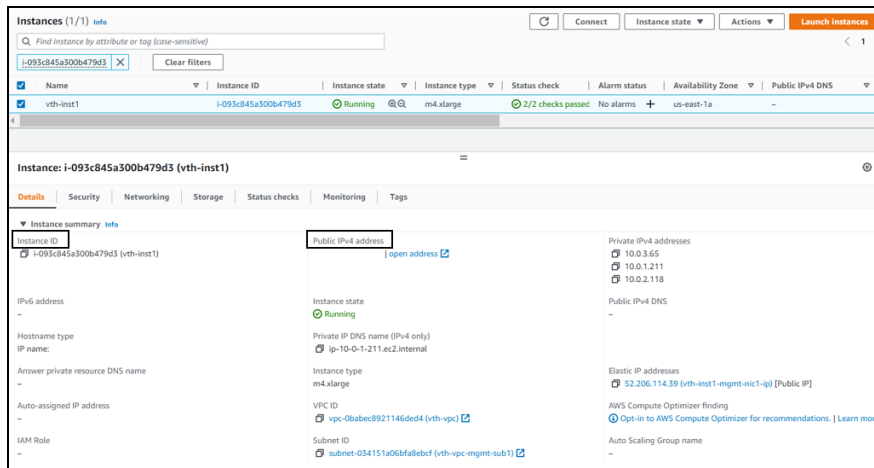
- [Access vThunder using GUI](#)
- [Access vThunder using CLI](#)

Access vThunder using GUI

To access vThunder instances using GUI, perform the following steps:

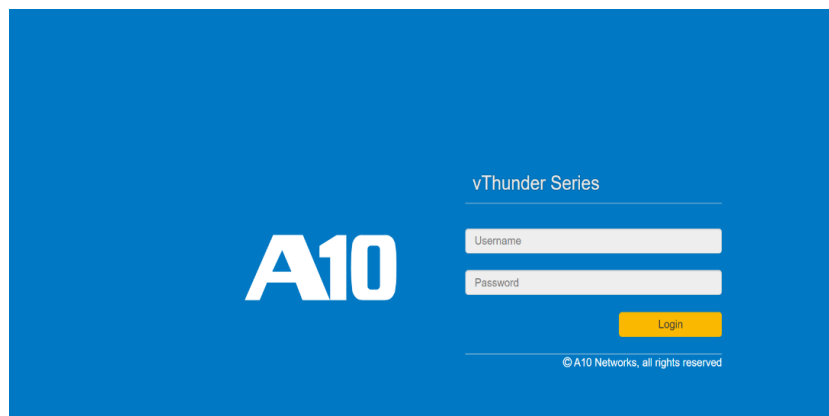
1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select your instance name.
Here, **vth-inst1**, **vth-inst2** are the vThunder instances.

Figure 51 : vThunder instance



3. For each vThunder instance, perform the following steps:
 - a. Copy the **Public IPv4 address** from the **Details** tab and replace the IP address in the below link:
`http://<vThunder_public_IPv4_address>`
 - b. Open the updated link in any browser.
 The vThunder login window is displayed.

Figure 52 : vThunder GUI



- c. Enter the following credentials and click **Login**:
 - Username – **admin**
 - Password – **EC2 Instance ID**

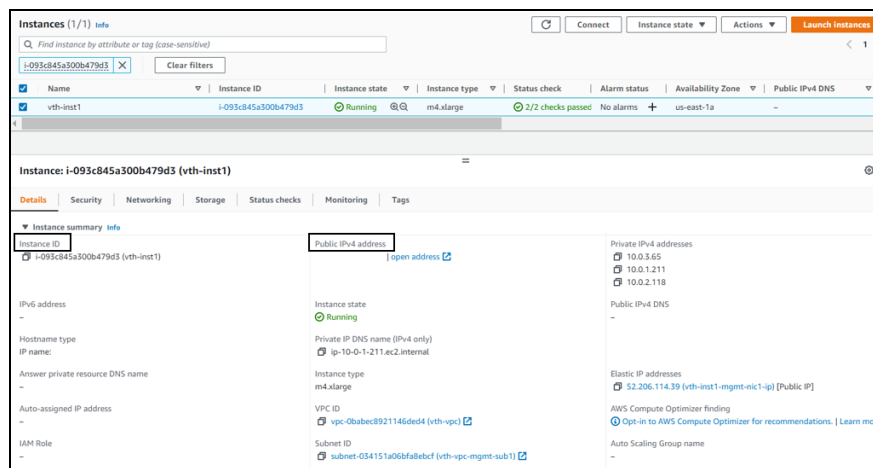
The home page is displayed if the entered credentials are correct.

Access vThunder using CLI

To access vThunder instances using CLI, perform the following steps:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select your instance name.
Here, **vth-inst1**, **vth-inst2** are the vThunder instances.

Figure 53 : vThunder instance



3. For each vThunder instance, perform the following steps:
 - a. Copy the **Public IPv4 address** from the **Details** tab.
 - b. Open any SSH client and provide the following to establish a connection:
 - Hostname: Public IPv4 address
 - Username: admin
 - Key: SSH Key
 - c. Connect to the session.
 - d. In the SSH client session, run the following commands:

```
vThunder (NOLICENSE)>enable <---Execute command--->
Password: <---just press Enter key--->
vThunder (NOLICENSE) #config <---Configuration mode--->
vThunder (config) (NOLICENSE) #
```

The vThunder instances are ready to use.

Configure Server and Client Machine

To test the traffic flow via vThunder, create and configure a server machine and a client machine:

- [Configure a Server Machine](#)
- [Configure a Client Machine](#)

Configure a Server Machine

To configure a server machine, perform the following steps:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select your autoscale server instance name.
Here, `autoscaling-vm` is the server instance name.
3. Click **Connect**.
A **Connect to instance** window with **EC2 Instance Connect** tab is displayed.
4. Click **Connect**.
A **Terminal** window is displayed.
5. Run the following commands in the Terminal window to update all the package information:

```
sudo apt update
```

6. Run the following command in the Terminal window to create an Apache Server virtual machine:

```
sudo apt install apache2
```

While the Apache server is getting installed, you get a prompt to continue further. Enter 'Y' to continue. After the installation is complete, a newline prompt is displayed.

Configure a Client Machine

To configure a client machine, perform the following steps:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Click **Launch Instances**.
A **Launch an instance** window is displayed.
3. In the **Name and tags** section, enter an instance name.
Here, enter `vth-client` as the client instance name.
4. In the **Application and OS Images** section, select **Ubuntu**.
5. In the **Instance type** section, select the required instance type.
6. In the **Key pair (login)** field, select your SSH key.
7. In the **Network settings** section, click **Edit** to edit the following:
 - VPC: *your VPC*
Here, `vth-vpc` is the VPC.
 - Subnet: Data subnet 1
Here, `10.0.2.0/24` is the data subnet value.
 - Auto-assign public IP: Enable
 - Firewall (security groups): Select existing security group
 - Common security groups: *your data security group*
Here, `vth-vThunderSecurityGroupData` is the security group.
8. Click **Launch instance**.

NOTE: The system may take a few minutes to launch the instance.

The client instance is displayed in the **Instances** list with the status as **Running**.

Configure FTP server

In a High Availability (HA) configuration, IP switching occurs between two vThunder instances. The IP switching is enabled when AWS keys are imported on the vThunder instances. To import the AWS Access Keys, configure an FTP server.

To configure an FTP server, perform the following steps:

1. From **AWS Management Console**, navigate to **EC2 > Instances > <FTP_server_name>**.

2. Click **Launch Instances**.

A **Launch an instance** window is displayed.

3. Run the following command in the FTP server instance to create an Apache Server virtual machine:

```
sudo apt install apache2
```

While the Apache server is getting installed, you get a prompt to continue further. Enter 'Y' to continue. After the installation is complete, a newline prompt is displayed.

4. Run the following command in the same instance to open the `aws_access_key.txt` file:

```
sudo vi /var/www/html/aws_access_key.txt
```

5. Press **Esc** and enter **i** to enable edit/insert mode.
6. Locate and open the **credentials** file from downloaded CFT folder.
7. Copy the updated access key ID and secret access key as per your AWS account and paste in the `aws_access_key.txt` file. For more information, see [Prerequisites](#).
8. After the changes, press **Esc** then type **:wq** to save the changes and exit.

Configure vThunder as an SLB with HA

The following topics are covered:

- [Initial Setup](#)
- [Deploy vThunder as an SLB](#)
- [Verify Access Keys on vThunder Instances](#)

Initial Setup

Before deploying vThunder on AWS cloud as an SLB with HA, you need to configure the corresponding parameters in the CFT.

To configure the parameters, perform the following steps:

1. Navigate to the folder where you have downloaded the CFT, and open the CFT_TMPL_3NIC_2VM_HA_GLM_PUBVIP_BACKAUTO_CONFIG_SSL_SLB_HA_GLM_PARAM.json with a text editor.

NOTE: Each parameter has a default value mentioned in the parameter file.

2. Configure the stack name created using CFT.

```
"stackDetails": {
  "value": [
    {
      "stackName": "vth-stack"
    }
  ]
},
```

3. Set the capacity of the autoscale server.

```
"desiredCapacity": 1,
```

4. Configure Service Group.

The Service group name by default is “sg+port_number”. If you may want to change the service group name then after changing the name, change the names in the Virtual servers as well.

```
"serviceGroup": {
  "value": [
    {
      "name": "sg443",
      "protocol": "tcp"
    },
    {
      "name": "sg53",
      "protocol": "udp"
    },
    {
      "name": "sg80",
      "protocol": "tcp"
    }
  ]
},
```

5. Configure a Virtual Server and its ports.

The virtual server default name is “vip”. It is the Private IP address of Ethernet1.

```

"virtualServerList": {
  "virtual-server-name": "vip",
  "metadata": {
    "description": "virtual server is using ethernet 1 ip
address"
  },
  "value": [
    {
      "port-number": 53,
      "protocol": "udp",
      "auto": 1,
      "service-group": "sg53"
    },
    {
      "port-number": 80,
      "protocol": "http",
      "auto": 1,
      "service-group": "sg80"
    },
    {
      "port-number": 443,
      "protocol": "https",
      "auto": 1,
      "service-group": "sg443"
    }
  ]
},

```

6. Configure DNS.

```

"dns": {
  "value": "8.8.8.8"
},

```

7. Configure a Network Gateway IP.

The default value of network gateway IP address for data subnet is 10.0.2.1 and

for management subnet, it is 10.0.1.1.

```
"rib-list": [
  {
    "ip-dest-addr": "0.0.0.0",
    "ip-mask": "/0",
    "ip-nexthop-ipv4": [
      {
        "ip-next-hop": "10.0.2.1"
      },
      {
        "ip-next-hop": "10.0.1.1"
      }
    ]
  }
],
```

8. Set VRRP-A.

```
"vrrp-a": {
  "set-id": 1
},
```

9. Set a Terminal Idle Timeout.

```
"terminal": {
  "idle-timeout": 0
},
```

10. Set VRID.

The default value of VRID is 0. For vThunder instance 1, the default priority is 100, and for vThunder instance 2, it is 99 (100-1).

```
"vrid-list": [
  {
    "vrid-val": 0,
    "blade-parameters": {
      "priority": 100
    }
  }
],
```

11. Configure SSL.

```
"sslConfig": {
  "requestTimeout": 40,
  "Path": "server.pem",
  "File": "server",
  "certificationType": "pem"
},
```

NOTE: By default, SSL configuration is disabled i.e., no SSL configuration is applied.

The server.pem file is included in the downloaded CFT folder. If necessary, you can edit this file or use a different certificate file. When you use a different certificate file, remember to change the path accordingly.

12. Configure your GLM account details.

To obtain the **Entitlement Token**, go to [GLM](#) > **Licenses** > select your license > **Overview** > **Info** tab.

```
"user_name": {
  "value": "xxxxxxx@a10networks.com"
},
"user_password": {
  "value": "user_password"
},
"entitlement_token": {
  "value": "XXXXXXXXXXXX"
},
```

13. Configure SLB server ports.

```
"port-list": [
{
  "port-number": 53,
  "protocol": "udp"
},
{
```

```

        "port-number": 80,
        "protocol": "tcp"
    },
    {
        "port-number": 443,
        "protocol": "tcp"
    }
],

```

14. Verify if all the configurations in the CFT_TMPL_3NIC_2VM_HA_GLM_PUBVIP_BACKAUTO_CONFIG_SSL_SLB_HA_GLM_PARAM.json file are correct and then save the changes.

Deploy vThunder as an SLB

To deploy vThunder on AWS cloud as an SLB, perform the following steps:

1. From your command prompt, navigate to the folder where you have downloaded the CFT.
2. Run the following command to create vThunder SLB instance:

```
python ./CFT_TMPL_3NIC_2VM_HA_GLM_PUBVIP_BACKAUTO_CONFIG_SSL_SLB_HA_GLM_3.py
```

A message is prompted to upload the SSL certificate and configure GLM and HA.

3. Enter **yes** if you want to upload the SSL certificate and configure GLM and HA.

```

Do you want to upload ssl certificate(yes/no)?yes
Do you want to configure GLM (yes/no)?yes
Do you want to configure HA (yes/no)?yes

```

A message is prompted that displays the criteria for a valid admin user password.

```

Primary conditions for password validation, user should provide the new
password according to the given combination:

```

```

Minimum length of 9 characters
Minimum lowercase character should be 1
Minimum uppercase character should be 1
Minimum number should be 1

```

```
Minimum special character should be 1
Should not include repeated characters
Should not include more than 3 keyboard consecutive characters.
```

A message is prompted to enter a new password.

4. Provide the new password for the admin user and then confirm the new password.

The new password should follow the valid admin user password criteria mentioned in the above step. For more information, see [Default Password Policy](#).

```
Enter vThunder's new password:
Confirm new password:
```

If both the password and confirm password field values match and complies to the valid admin user password criteria, a message 'Password changed successfully' is displayed along with the other configurations.

```
Password changed successfully.
-----
-----
Configuring vThunder with instance id i-078c5d5eaad0958cf
AWS access key uploaded to vThunder.
configured ethernet ip
configured ethernet ip
Configure service group
Configured virtual servers
SSL Configured.
License activation completed.
Configured primary dns
configure glm.
Glm license request sent successfully.
Successfully Configured IP route
Configured Vrrp A common configuration
Configured idle timeout
Configured vrrp rid
Configured peer group
Configurations are saved on partition: shared
-----
```



```

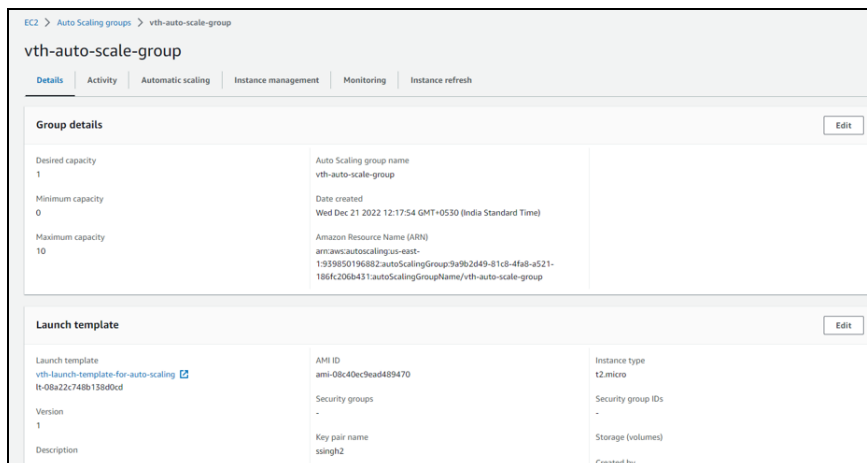
-----
Configuring vThunder with instance id i-069a42154aeac5c76
AWS access key uploaded to vThunder.
configured ethernet ip
configured ethernet ip
Configure service group
Configured virtual servers
SSL Configured.
License activation completed.
Configured primary dns
configure glm.
Glm license request sent successfully.
Successfully Configured IP route
Configured Vrrp A common configuration
Configured idle timeout
Configured vrrp rid
Configured peer group
Configurations are saved on partition: shared
Updated desired capacity of autoscale group to 1

```

To change this password in future, see [On-demand Password Change](#).

5. Verify the desired capacity of the autoscale server from **AWS Management Console > EC2 > AutoScaling Group > <autoscale_group_name>**.

Figure 54 : AutoScaling Group Details



vth-auto-scale-group		
Details Activity Automatic scaling Instance management Monitoring Instance refresh		
Group details Edit		
Desired capacity	Auto Scaling group name	
1	vth-auto-scale-group	
Minimum capacity	Date created	
0	Wed Dec 21 2022 12:17:54 GMT+0530 (India Standard Time)	
Maximum capacity	Amazon Resource Name (ARN)	
10	arn:aws:autoscaling:us-east-1:939650196882:autoScalingGroup:39b2d49-81c8-4fab-a521-186fc206b431:autoScalingGroupName/vth-auto-scale-group	
Launch template Edit		
Launch template	AMI ID	Instance type
vth-launch-template-for-auto-scaling It-69ba22c74bb13860d	ami-08c40cead489470	t2.micro
Version	Security groups	Security group IDs
1	-	-
Description	Key pair name	Storage (volumes)
-	singh2	Created by

Verify Access Keys on vThunder Instances

To verify if the access keys are uploaded on both vThunder instances, perform the following step:

1. Run the following commands on each vThunder instance:

```
vThunder(config)#admin admin
vThunder(config-admin:admin)#aws-accesskey show
[default]
aws_access_key_id = your_aws_access_key_id
aws_secret_access_key = your_aws_secret_access_key
vThunder(config-admin:admin)#exit
vThunder(config)#
vThunder(config)#
vThunder-Active(config)#
```

The access keys are displayed.

NOTE: Once the AWS access keys are imported to vThunder, the FTP server gets deleted automatically.

Verify Deployment

To verify vThunder SLB deployment using CFT, perform the following steps:

1. Access the vThunder instances using CLI.
2. Run the following command on Master controller:

```
vThunder-Active(config)#show running-config
```

If the deployment is successful, the following SLB configuration is displayed:

```
!Current configuration: 349 bytes
!Configuration last updated at 10:56:58 GMT Fri Jan 6 2023
!Configuration last saved at 10:53:34 GMT Fri Jan 6 2023
!64-bit Advanced Core OS (ACOS) version 5.2.1, build 153 (Dec-11-2020,16:36)
!
```

```
vrrp-a common
  device-id 1
  set-id 1
  enable
!
terminal idle-timeout 0
!
ip dns primary 8.8.8.8
!
!
glm use-mgmt-port
glm enable-requests
glm token vTh205fe920b
!
interface ethernet 1
  enable
  ip address dhcp
!
interface ethernet 2
  enable
  ip address dhcp
!
vrrp-a vrid 0
  floating-ip 10.0.3.12
  blade-parameters
    priority 99
!
vrrp-a peer-group
  peer 10.0.2.117
  peer 10.0.2.173
!
ip route 0.0.0.0 /0 10.0.2.1
ip route 0.0.0.0 /0 10.0.1.1
!
slb server i-0b3aad427ca3c2f23 10.0.3.23
  port 53 udp
  port 80 tcp
```

```

    port 443 tcp
!
slb service-group sg443 tcp
    member i-0b3aad427ca3c2f23 443
!
slb service-group sg53 udp
    member i-0b3aad427ca3c2f23 53
!
slb service-group sg80 tcp
    member i-0b3aad427ca3c2f23 80
!
slb virtual-server vip 10.0.2.121
    port 53 udp
        source-nat auto
        service-group sg53
    port 80 http
        source-nat auto
        service-group sg80
    port 443 https
        source-nat auto
        service-group sg443
!
!
end
!Current config commit point for partition 0 is 0 & config mode is
classical-mode
vThunder-Active(config)#

```

3. Run the following command on vThunder instance 1:

```
vThunder-Active(config)#show pki cert
```

If the deployment is successful, the following SSL configuration is displayed:

Name	Type	Expiration	Status

server certificate	Jan 28 12:00:00 2028 GMT	[Unexpired, Bound]	

At this point, the vThunder instance 2 has the following prompt:

```
vThunder-Standby (config) #
```

Figure 55 : vThunder instance 1 - Active

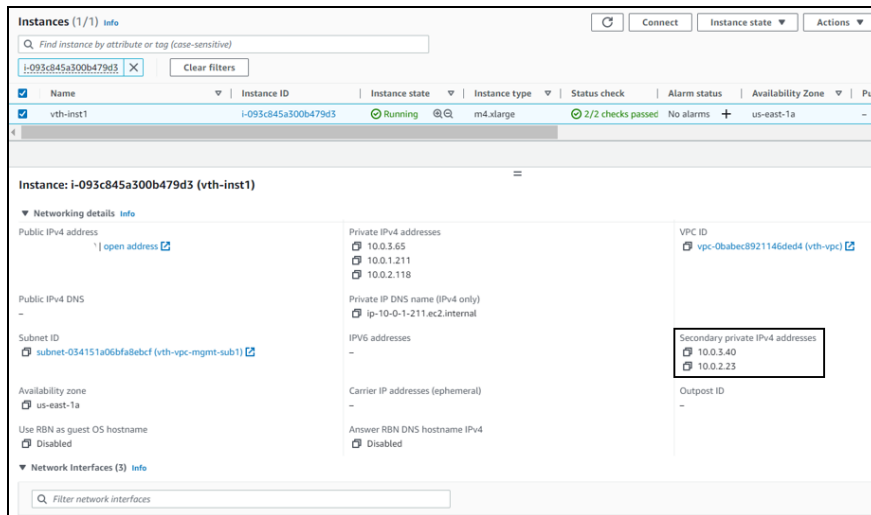
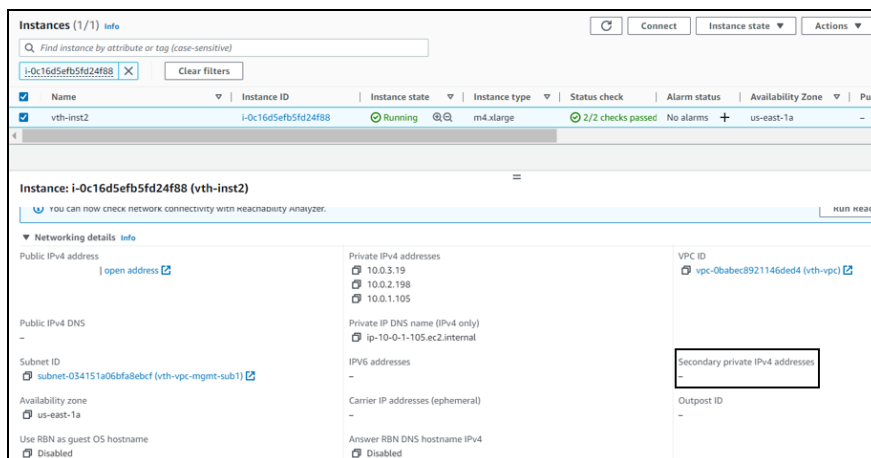


Figure 56 : vThunder instance 2 - Standby



4. Run the following command on vThunder instance 1:

```
vThunder-Active (config) #vrrp-a force-self-standby enable
vThunder-Active (config) #
vThunder-ForcedStandby (config) #
```

At this point, IP switching occurs and the vThunder instance 2 prompt becomes:

```
vThunder-Active (config) #
```

5. Run the following command on vThunder instance 2:

```
vThunder-Active(config)#show running-config
```

If the deployment is successful, the following SLB configuration is displayed:

```
!Current configuration: 282 bytes
!Configuration last updated at 10:53:35 GMT Fri Jan 6 2023
!Configuration last saved at 10:53:37 GMT Fri Jan 6 2023
!64-bit Advanced Core OS (ACOS) version 5.2.1, build 153 (Dec-11-
2020,16:36)
!
vrrp-a common
    device-id 2
    set-id 1
    enable
!
terminal idle-timeout 0
!
ip dns primary 8.8.8.8
!
!
glm use-mgmt-port
glm enable-requests
glm token vTh205fe920b
!
interface ethernet 1
    enable
    ip address dhcp
!
interface ethernet 2
    enable
    ip address dhcp
!
vrrp-a vrid 0
    floating-ip 10.0.3.12
    blade-parameters
        priority 98
!
vrrp-a peer-group
```

```
peer 10.0.2.117
peer 10.0.2.173
!
ip route 0.0.0.0 /0 10.0.2.1
ip route 0.0.0.0 /0 10.0.1.1
!
slb server i-0b3aad427ca3c2f23 10.0.3.23
    port 53 udp
    port 80 tcp
    port 443 tcp
!
slb service-group sg443 tcp
    member i-0b3aad427ca3c2f23 443
!
slb service-group sg53 udp
    member i-0b3aad427ca3c2f23 53
!
slb service-group sg80 tcp
    member i-0b3aad427ca3c2f23 80
!
slb virtual-server vip 10.0.2.121
    port 53 udp
        source-nat auto
        service-group sg53
    port 80 http
        source-nat auto
        service-group sg80
    port 443 https
        source-nat auto
        service-group sg443
!
!
end
!Current config commit point for partition 0 is 0 & config mode is
classical-mode
vThunder-Active(config)#
```

6. Run the following command on vThunder instance 2:

```
vThunder-Active (config) #show pki cert
```

If the deployment is successful, the following SSL configuration is displayed:

Name	Type	Expiration	Status
server certificate	Jan 28 12:00:00 2028 GMT	[Unexpired, Bound]	

Figure 57 : vThunder instance 1 - Standby

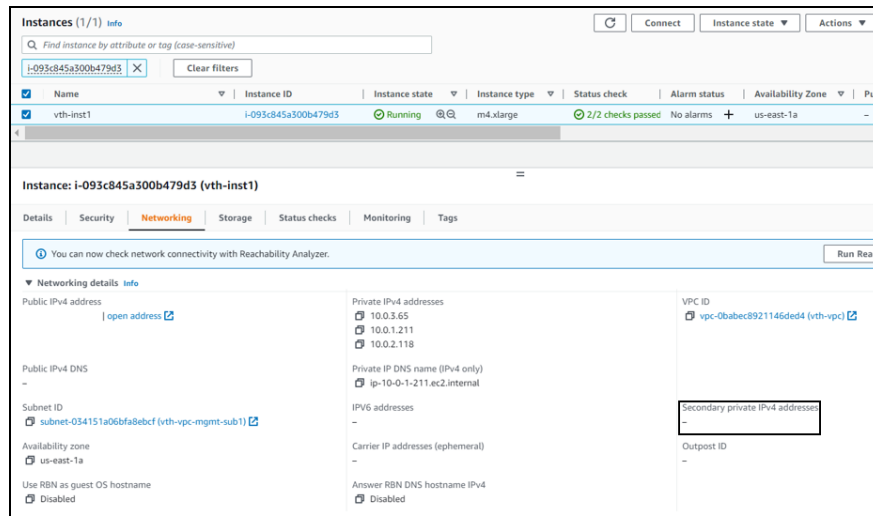
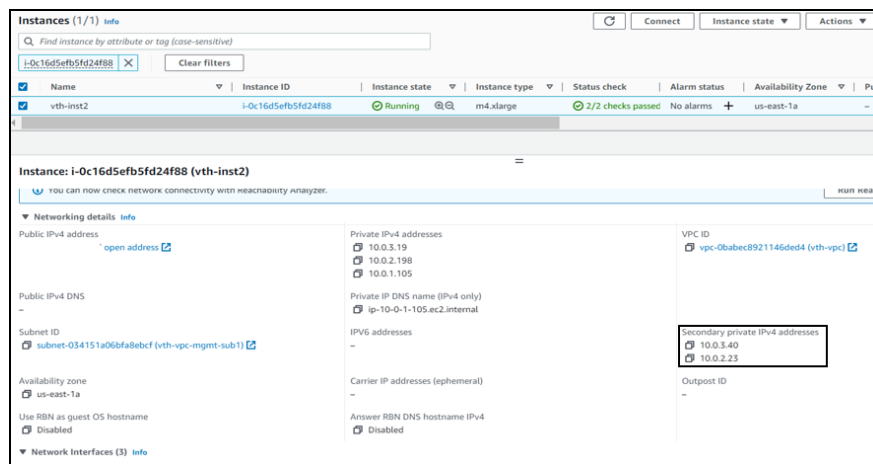


Figure 58 : vThunder instance 2 - Active



- If you want to make vThunder instance 1 active, run the following command on vThunder instance 2:


```
vThunder-Active(config)#vrrp-a force-self-standby disable
vThunder-Active(config)#
vThunder-ForcedStandby(config)#
```

At this point, IP switching occurs and the vThunder instance 1 prompt becomes:

```
vThunder-Active(config)#
```

NOTE: If you want to access the vThunder instances using GUI, re-login the instances using the new admin user password and NOT the *EC2 Instance ID*.

Verify GLM

The application of license can be verified using any of the following ways:

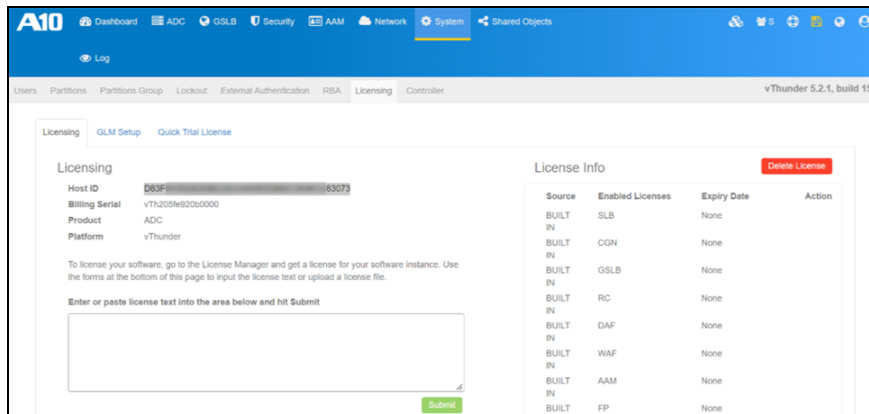
- [Verify License using GUI](#)
- [Verify License using CLI](#)

Verify License using GUI

To verify license using GUI, perform the following steps:

1. Log in to your vThunder instance using Public IPv4 address.
Here, `vth-inst1` is the vThunder instance.
2. Navigate to **Profile > Setting > Licensing**.
3. Click the **Licensing** tab.
If the license is successfully applied on vThunder, the Host ID is displayed.

Figure 59 : GLM > Licensing window



Verify License using CLI

To verify license using CLI, perform the following steps:

1. Verify if NOLICENSE is removed from the vThunder prompt.
2. Run the following command on vThunder:

```
vThunder(config)#show license
```

If the license is successfully applied on vThunder, the Host ID is displayed:

```
Host ID      : D83F****82EB633D****067DB84136****83073
```

3. Run the following command on vThunder instance:

```
vThunder(config)#show license-info
```

If the license is successfully applied on vThunder, the following GLM configuration is displayed:

```
Host ID      : D83F****82EB633D****067DB84136****83073
USB ID       : Not Available
Billing Serials: vTh205fe920b0000
Token        : Not Available
Product      : ADC
Platform     : vThunder
Burst        : Disabled
GLM Ping Interval In Hours : 24
```

Enabled Licenses	Expiry Date (UTC)	Notes
SLB	None	
CGN	None	
GSLB	None	
RC	None	
DAF	None	
WAF	None	
AAM	None	
FP	None	
WEBROOT	N/A	Requires an additional Webroot license.
THREATSTOP	N/A	Requires an additional ThreatSTOP license.
QOSMOS	N/A	Requires an additional QOSMOS license.
WEBROOT_TI	N/A	Requires an additional Webroot Threat Intel license.
IPSEC_VPN	N/A	Requires an additional IPsec VPN license.
500 Mbps Bandwidth 20-January-2023		

Verify Traffic Flow

For this template, the traffic flow is verified using the following:

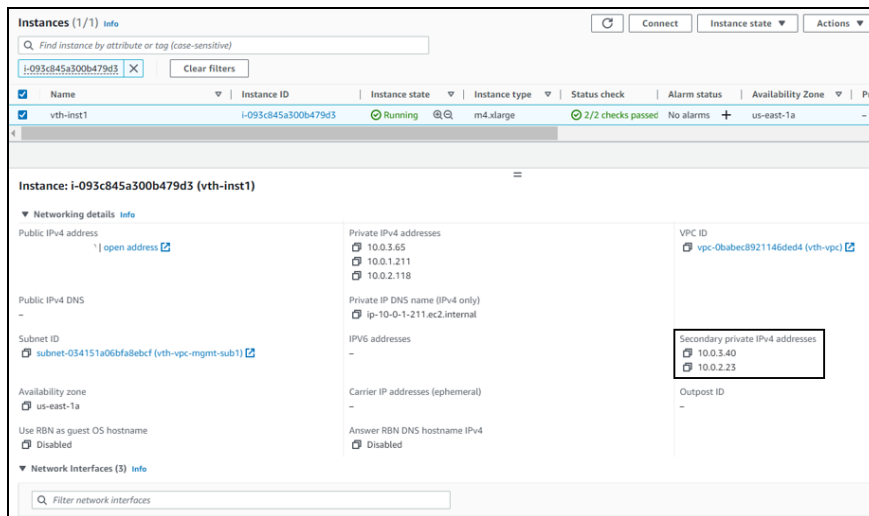
- [Secondary Private IPv4 Address](#)
- [Allocated Public IPv4 address](#)

Secondary Private IPv4 Address

To verify the traffic flow from client machine to server machine via vThunder instance secondary private IPv4 address, perform the following:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select the active vThunder instance name and then click the **Networking** tab.
3. Copy the IP address of the vThunder Secondary IPv4 Address under the **Private IPv4 address**.

Figure 60 : vThunder instance 1 - Networking tab



4. Select your client instance from the **Instances** list.
Here, **vth-client** is the client instance name.
5. Click **Connect**.
A **Connect to instance** window with **EC2 Instance Connect** tab is displayed.
6. Click **Connect**.
A **Terminal** window is displayed.
7. Run the following command in the Terminal window to send the traffic from the client machine:

```
curl <vThunder_instance_secondary_private_IPv4_Address>
```

Example

```
curl 10.0.2.23
```

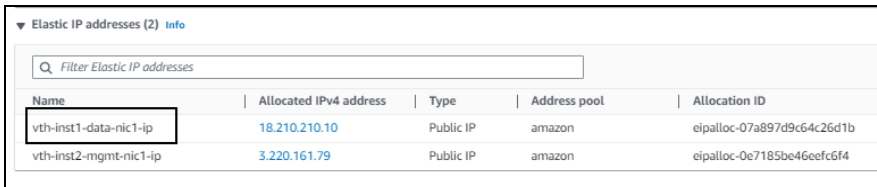
8. Verify if a response is received.

Allocated Public IPv4 address

To verify the traffic flow from client machine to server machine via vThunder instance allocated public IPv4 address, perform the following:

1. Select the active vThunder instance name and then click its **Networking** tab.
2. Copy the Allocated IPv4 address of data interface under the **Elastic IP address**. Here, `vth-inst1-data-nic1-ip` is the Elastic Public IP of data interface.

Figure 61 : Elastic IP addresses



Name	Allocated IPv4 address	Type	Address pool	Allocation ID
vth-inst1-data-nic1-ip	18.210.210.10	Public IP	amazon	eipalloc-07a897d9c64c26d1b
vth-inst2-mgmt-nic1-ip	3.220.161.79	Public IP	amazon	eipalloc-0e7185be46eefc6f4

3. Select your client instance from the **Instances** list. Here, `vth-client` is the client instance name.
4. Click **Connect**. A **Connect to instance** window with **EC2 Instance Connect** tab is displayed.
5. Click **Connect**. A **Terminal** window is displayed.
6. Run the following command in the Terminal window to send the traffic from client machine:

```
curl <Allocated_IPv4_address_of_data_interface>
```

Example

```
curl 18.210.210.10
```

7. Verify if a response is received.

On-demand Password Change

For the vThunder instances that are created using this template, you can change the existing admin user password anytime in future using the on-demand password change feature.

To change the on-demand password, perform the following steps:

1. From your command prompt, navigate to the folder where you have downloaded the CFT.
2. Run the following command to change the password:

```
python ./CFT_TMPL_3NIC_2VM_HA_GLM_PUBVIP_BACKAUTO_ON-DEMAND_PASSWORD_CHANGE.py
```

A message is prompted to enter the existing password.

3. Provide the existing vThunder admin user password.

```
Enter vThunder's existing password:
```

If the entered password is correct, a message is prompted that displays the criteria for a valid admin user password.

```
Primary conditions for password validation, user should provide the new password according to the given combination:
```

```
Minimum length of 9 characters
```

```
Minimum lowercase character should be 1
```

```
Minimum uppercase character should be 1
```

```
Minimum number should be 1
```

```
Minimum special character should be 1
```

```
Should not include repeated characters
```

```
Should not include more than 3 keyboard consecutive characters.
```

If the entered password is incorrect after three consecutive attempts, re-run the on-demand password change script.

4. Provide the new password for admin user and then confirm the new password when prompted.

```
Enter vThunder's new password:
```

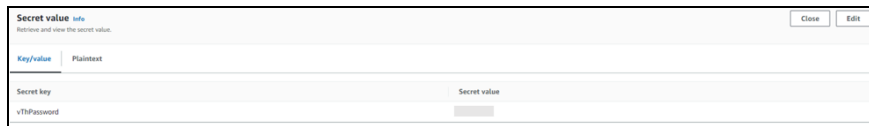
```
Confirm new password:
```

The new password should follow the Default password policy. For more information, see [Default Password Policy](#).

If the password change is successful, a message 'Password changed successfully' is displayed.

5. Access the vThunder instances using GUI to verify if the vThunder admin user password is updated with the new password.
6. Navigate to **AWS Management Console > Secrets Manager > Secrets > <secret_manager_name>** and verify if the secret value key “vThPassword” is updated with new password.

Figure 62 : Secret Value window



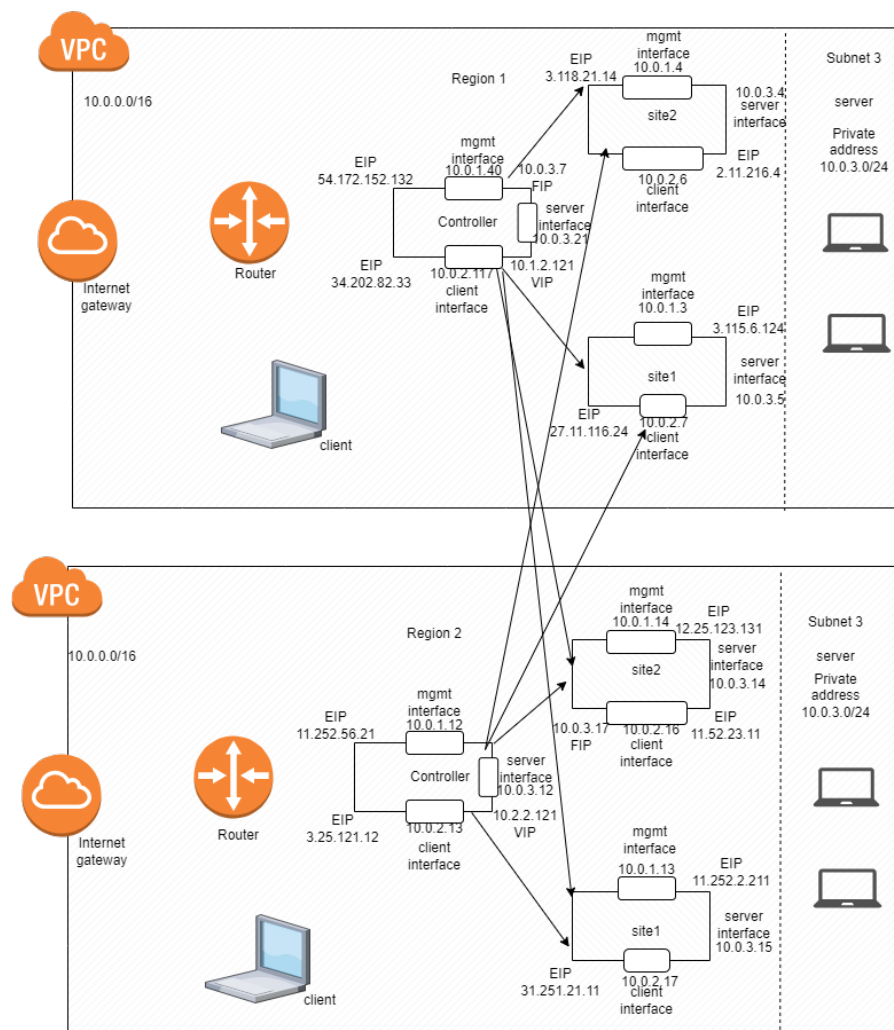
7. Access the vThunder instances using the updated password for the admin user.

Deploy CFT A10-vThunder_ADC-3NIC-6VM-2RG-GSLB

Using this template, you can deploy two regions each containing one GSLB controller and two site devices. After the deployment of site devices, a server is assigned to each site devices.

[Figure 63](#) illustrates the deployment topology for the 3NIC-6VM-2RG-GSLB template.

Figure 63 : 3NIC-6VM-2RG-GSLB Deployment Topology



The following topics are covered:

System Requirements	161
Supported Instance Types	165
Create vThunder Instances	168
Access vThunder using GUI or CLI	175
Configure vThunder as an SLB	178
Verify Deployment	190
Verify Traffic Flow	208

System Requirements

When the files from the CFT are downloaded and saved on your local machine, the resources and their default values are displayed as shown in [Table 13](#). You can configure the resources and modify these default values based on your deployment requirements.

Table 13 : System Requirements

Resource Name	Description	Default Value
Stack	Two stacks are created with the specified name and location.	In this template, <code>vth-stack1</code> and <code>vth-stack2</code> are used as the default stack names. Therefore, the respective resources has <code>vth-stack1</code> and <code>vth-stack2</code> as the prefixes.
Network Interface Card [NIC]	Total nine interfaces are created for each region. Two types of interfaces are created for each vThunder instance: • One management interface • Two data interfaces	<code>vth-stack1-inst1-mgmt-nic1</code> <code>vth-stack1-inst1-data-nic1</code> <code>vth-stack1-inst1-data-nic2</code> <code>vth-stack1-inst2-mgmt-</code>

Resource Name	Description	Default Value
		nic1 vth-stack1-inst2-data-nic1 vth-stack1-inst2-data-nic2 vth-stack1-inst3-mgmt-nic1 vth-stack1-inst3-data-nic1 vth-stack1-inst3-data-nic2 vth-stack2-inst1-mgmt-nic1 vth-stack2-inst1-data-nic1 vth-stack2-inst1-data-nic2 vth-stack2-inst2-mgmt-nic1 vth-stack2-inst2-data-nic1 vth-stack2-inst2-data-nic2 vth-stack2-inst3-mgmt-nic1 vth-stack2-inst3-data-nic1

Resource Name	Description	Default Value
		<code>vth-stack2-inst3-data-nic2</code>
Subnet	Three subnets are created with an address prefix each, for a region.	<code>vth-stack1-vpc-mgmt-sub1</code> <code>vth-stack1-vpc-data-sub1</code> <code>vth-stack1-vpc-data-sub2</code> <code>vth-stack2-vpc-mgmt-sub1</code> <code>vth-stack2-vpc-data-sub1</code> <code>vth-stack2-vpc-data-sub2</code>
Virtual Private Network [VCN]	A virtual private network is assigned to the virtual machine instance.	<code>vth-stack1-vpc</code> <code>vth-stack2-vpc</code> Address prefix for virtual network: 10.0.0.0/16
Elastic Public IP	Six Elastic Public IPs addresses are created and attached to the management and data interfaces of the vThunder instances for each region.	<code>vth-stack1-inst1-mgmt-nic1-ip</code> <code>vth-stack1-inst1-data1-nic1-ip</code> <code>vth-stack1-inst2-mgmt-nic1-ip</code> <code>vth-stack1-inst2-data1-nic1-ip</code> <code>vth-stack1-inst3-mgmt-</code>

Resource Name	Description	Default Value
		nic1-ip vth-stack1-inst3-data1-nic1-ip vth-stack2-inst1-mgmt-nic1-ip vth-stack2-inst1-data1-nic1-ip vth-stack2-inst2-mgmt-nic1-ip vth-stack2-inst2-data1-nic1-ip vth-stack2-inst3-mgmt-nic1-ip vth-stack2-inst3-data1-nic1-ip
Security Group	Two security groups are created for each region: - One security group for management interface. - One security group for client-side data interface and server-side data interface. Logical name: - vThunderSecurityGroupMgmt - vThunderSecurityGroupData	Here, the tag names are: vth-stack1-sg-mgmt vth-stack1-sg-data vth-stack2-sg-mgmt vth-stack2-sg-data
vThunder Instance	Three vThunder EC2 instances are created for each region.	vth-stack1-controller-region1

Resource Name	Description	Default Value
	Default type: m4.xlarge (40 Gb memory) Table 14 lists the supported instance types.	vth-stack1-site1-region1 vth-stack1-site2-region1 vth-stack2-controller-region2 vth-stack2-site1-region2 vth-stack2-site2-region2
Server Instance	Two Ubuntu Server instances are created for each region. Default Size: t2.micro	vth-stack1-server1 vth-stack1-server2 vth-stack2-server1 vth-stack2-server2

Supported Instance Types

[Table 14](#) provides detailed information about the supported instance types.

Table 14 : List of Supported Instance Types

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

Instance	vCPU	Memory	Number of Network Interfaces
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

Instance	vCPU	Memory	Number of Network Interfaces
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

Create vThunder Instances

The vThunder instances are created in two regions:

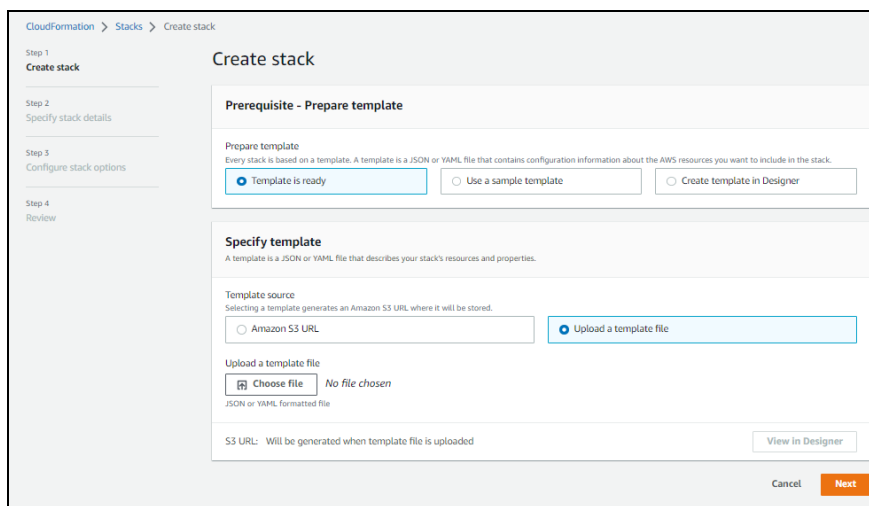
- [vThunder instances for Region1](#)
- [vThunder instances for Region2](#)

vThunder instances for Region1

To create vThunder instances for region1, perform the following steps:

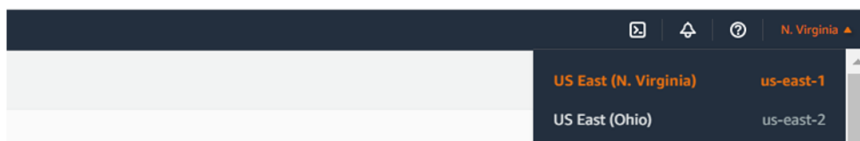
1. From **AWS Management Console**, navigate to **CloudFormation > Stacks > Create Stack > With new resources (standard)**.
The **Create stack** window is displayed.

Figure 64 : Create stack window



2. Select region1 from the dropdown on the **AWS Management Console** top ribbon.

Figure 65 : Region dropdown



3. In the **Prerequisite - Prepare template** section, select **Template is ready**.
After the selecting this option, the **Specify template** section is displayed.

4. In the **Specify template** section, select **Upload a template file** and click **Choose file** to browse and upload each of the following template files from the downloaded CFT folder:

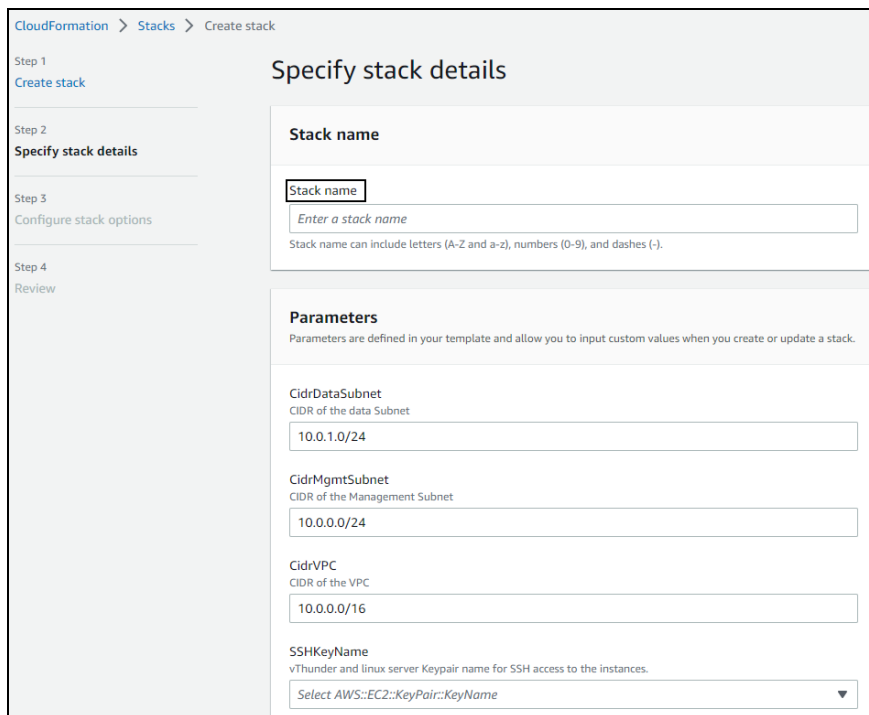
CFT_TMPL_3NIC_6VM_2RG_GSLB_REGION_1

The selected template file name is displayed as the chosen file.

5. Click **Next**.

The **Specify stack details** window is displayed.

Figure 66 : Specify stack details window



6. In the **Specify stack details** window, enter or select the following:
 - a. In the **Stack name** section, enter a **Stack name**.
Here, **vth-stack1** is provided as the default stack name.
 - b. In the **Parameters** section, enter or select the required values in the following fields:
 - **KeyPairName**: *<your SSH key>*
 - **TagValue**: **a10-vthunder-adc**

- **Zone:** *<your availability zone>*
- c. Verify the other fields and change the values appropriately. (Optional)
If the default IP addresses of the following fields changed then the updated IP addresses must be within CIDR range of Data Subnet 1:
 - PrimaryIpGSLBMaster
 - PrimaryIpSite1
 - PrimaryIpSite2
 - VIPGSLBMaster
 - VIPSite1
 - VIPSite2

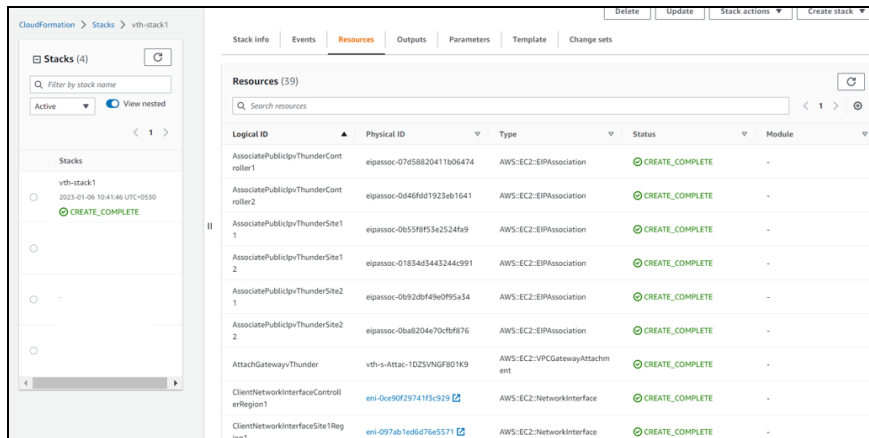
NOTE: 10.1.2.0/24 is default CIDR of region1.

7. Click **Next**.
The **Configure stack options** window is displayed.
8. Verify the fields and change the values appropriately. (Optional)
9. Click **Next**.
The **Review** window is displayed.
10. Verify if all the stack configurations are correct and then click **Submit**.

NOTE: The system may take a few minutes to create the resources and display the stack status as **CREATE_COMPLETE**.

11. Verify if all the above resources are created in the **AWS Management Console** > **CloudFormation** > **Stacks** > *<stack_name>* > **Resources** tab.

Figure 67 : Resource listing in the resource group



Logical ID	Physical ID	Type	Status	Module
AssociatePublicIpvThunderController1	eipassoc-07558820411b06474	AWS::EC2::EIPAssociation	CREATE_COMPLETE	-
AssociatePublicIpvThunderController2	eipassoc-0d48f6d1923eb1641	AWS::EC2::EIPAssociation	CREATE_COMPLETE	-
AssociatePublicIpvThunderSite1	eipassoc-0b55f8f53a2524fa9	AWS::EC2::EIPAssociation	CREATE_COMPLETE	-
AssociatePublicIpvThunderSite1	eipassoc-0183463443244c991	AWS::EC2::EIPAssociation	CREATE_COMPLETE	-
AssociatePublicIpvThunderSite2	eipassoc-0b92dbf43b0f95a34	AWS::EC2::EIPAssociation	CREATE_COMPLETE	-
AssociatePublicIpvThunderSite2	eipassoc-0ba8204a70c0b1876	AWS::EC2::EIPAssociation	CREATE_COMPLETE	-
AttachGatewayvThunder	vth-s-Attac-1025vngf801k9	AWS::EC2::VPCEGatewayAttachment	CREATE_COMPLETE	-
ClientNetworkInterfaceControllerRegion1	eni-0ce9f29741f5c929	AWS::EC2::NetworkInterface	CREATE_COMPLETE	-
ClientNetworkInterfaceSite1Region1	eni-087ab1ed6e76c5571	AWS::EC2::NetworkInterface	CREATE_COMPLETE	-

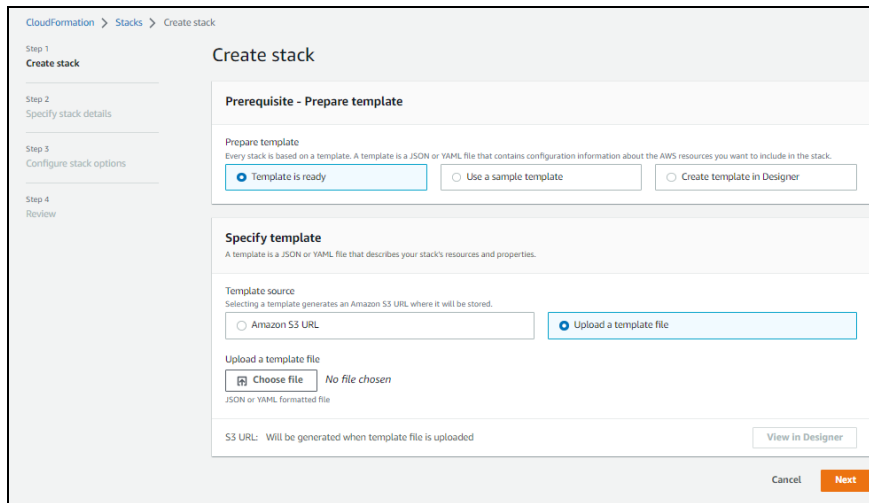
The three vThunder instances with the following Logical IDs for region1 are listed under **Resources** tab:

- vThunderControllerRegion1 (**vth-stack1-controller-region1**)
- vThunderSite1Region1 (**vth-stack1-site1-region1**)
- vThunderSite2Region1 (**vth-stack1-site2-region1**)

vThunder instances for Region2

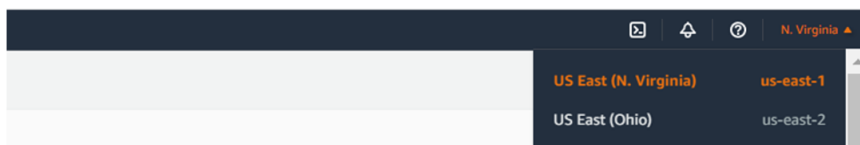
1. From **AWS Management Console**, navigate to **CloudFormation > Stacks > Create Stack > With new resources (standard)**.
The **Create stack** window is displayed.

Figure 68 : Create stack window



2. Select region2 from the dropdown on the **AWS Management Console** top ribbon.

Figure 69 : Region dropdown



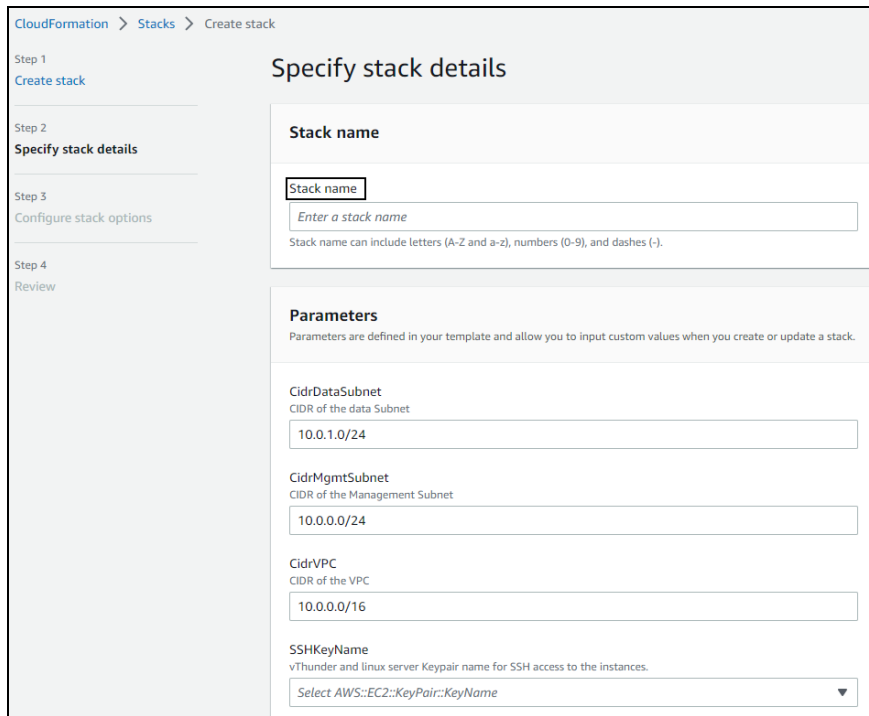
3. In the **Prerequisite - Prepare template** section, select **Template is ready**. After the selection of this option, the **Specify template** section is displayed.
4. In the **Specify template** section, select **Upload a template file** and click **Choose file** to browse and upload each of the following template files from the downloaded CFT folder:

CFT_TMPL_3NIC_6VM_2RG_GSLB_REGION_2

The selected template file name is displayed as the chosen file.

5. Click **Next**. The **Specify stack details** window is displayed.

Figure 70 : Specify stack details window



6. In the **Specify stack details** window, enter or select the following:
 - a. In the **Stack name** section, enter a **Stack name**.
Here, `vth-stack2` is provided as the default stack name.
 - b. In the **Parameters** section, enter or select the required values in the following fields:
 - **KeyPairName:** *<your SSH key>*
 - **TagValue:** `a10-vthunder-adc`
 - **Zone:** *<your availability zone>*
 - c. Verify the other fields and change the values appropriately. (Optional)
If the default IP addresses of the following fields changed then the updated IP addresses must be within CIDR range of Data Subnet 1:
 - PrimaryIpGSLBMaster
 - PrimaryIpSite1
 - PrimaryIpSite2

- VIPGSLBMaster
- VIPSite1
- VIPSite2

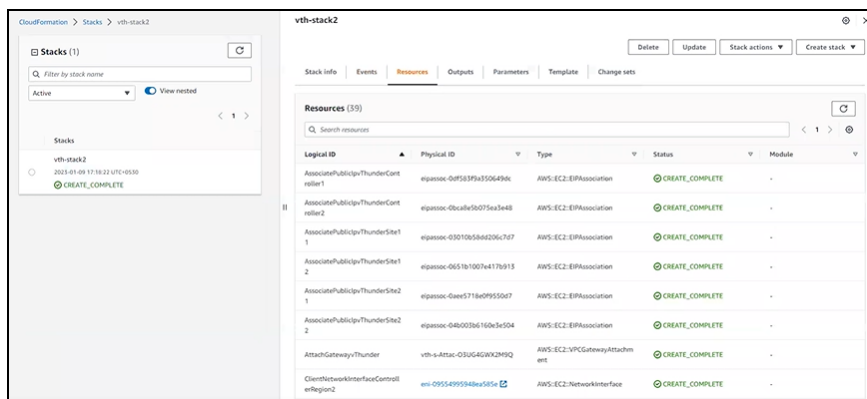
NOTE: 10.2.2.0/24 is default CIDR of region2.

7. Click **Next**.
The **Configure stack options** window is displayed.
8. Verify the fields and change the values appropriately. (Optional)
9. Click **Next**.
The **Review** window is displayed.
10. Verify if all the stack configurations are correct and then click **Submit**.

NOTE: It may take the system a few minutes to create the resources and display the stack status as **CREATE_COMPLETE**.

11. Verify if all the above resources are created in the **AWS Management Console > CloudFormation > Stacks > <stack_name> > Resources** tab.

Figure 71 : Resource listing in the resource group



The three vThunder instances with the following Logical IDs for region2 are listed under **Resources** tab:

- vThunderControllerRegion2 (**vth-stack2-controller-region2**)
- vThunderSite1Region2 (**vth-stack2-site1-region2**)
- vThunderSite2Region2 (**vth-stack2-site2-region2**)

Access vThunder using GUI or CLI

vThunder instances can be accessed using any of the following ways:

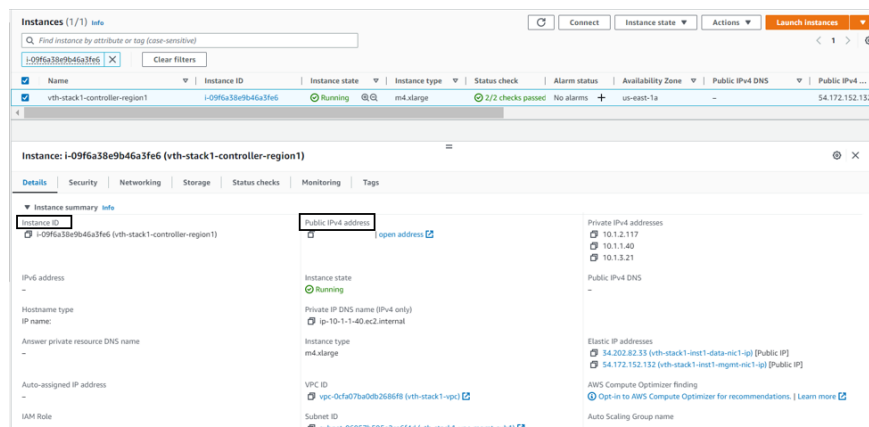
- [Access vThunder using GUI](#)
- [Access vThunder using CLI](#)

Access vThunder using GUI

To access vThunder instances using GUI, perform the following steps:

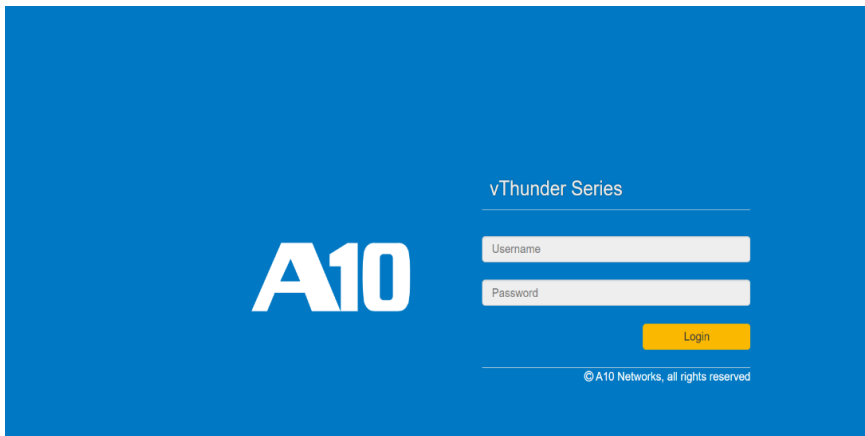
1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select your master controller instance name.
Here, **vth-stack1-controller-region1** is the master controller instance name.

Figure 72 : Master Controller Instance Name



3. Copy the **Public IPv4 address** from the **Details** tab and replace the IP address in the below link:
http://<vThunder_public_IPv4_address>
4. Open the updated link in any browser.
The vThunder login window is displayed.

Figure 73 : vThunder GUI



5. Enter the following credentials:

- Username – admin
- Password – *EC2 Instance ID*

The home page is displayed if the entered credentials are correct.

6. Similarly, verify the following vThunder instances of region1 and region2 are up and running:

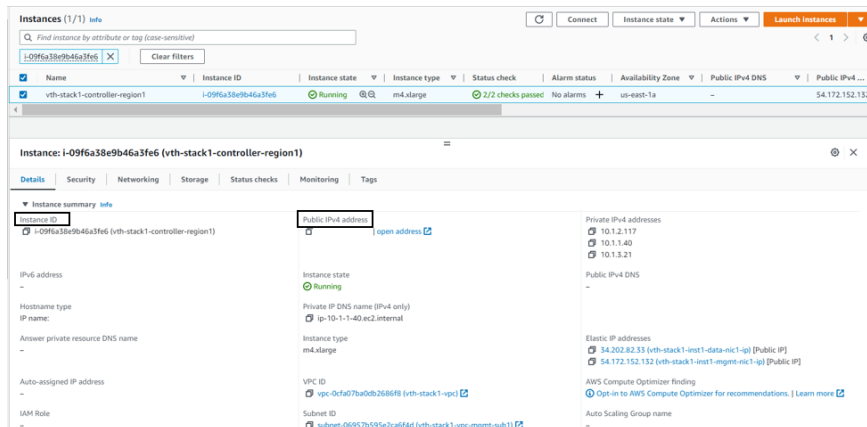
- `vth-stack1-site1-region1`
- `vth-stack1-site2-region1`
- `vth-stack2-controller-region2`
- `vth-stack2-site1-region2`
- `vth-stack2-site2-region2`

Access vThunder using CLI

To access vThunder instances using CLI, perform the following steps:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select your master controller instance name.
Here, `vth-stack1-controller-region1` is the master controller instance name.

Figure 74 : Master Controller Instance Name



3. Copy the **Public IPv4 address** from the **Details** tab.
4. Open any SSH client and provide the following to establish a connection:
 - Hostname: Public IPv4 address
 - Username: admin
 - Key: SSH Key
5. Connect to the session.
6. In the SSH client session, run the following commands:

```
vThunder (NOLICENSE) >enable <---Execute command--->
Password: <---just press Enter key--->
vThunder (NOLICENSE) #config <---Configuration mode--->
vThunder (config) (NOLICENSE) #
```

7. Similarly, verify the following vThunder instances of region1 and region2 are up and running:
 - vth-stack1-site1-region1
 - vth-stack1-site2-region1
 - vth-stack2-controller-region2
 - vth-stack2-site1-region2
 - vth-stack2-site2-region2

Configure vThunder as an SLB

The following topics are covered:

- [Initial Setup](#)
- [Deploy vThunder as an SLB](#)

Initial Setup

Before deploying vThunder on AWS cloud as an SLB with HA, you need to configure the corresponding parameters in the CFT.

To configure the parameters, perform the following steps:

1. Navigate to the folder where you have downloaded the CFT, and open the CFT_TMPL_3NIC_6VM_2RG_GSLB_CONFIG_GSLB_PARAM.json with a text editor.

NOTE: Each parameter has a default value mentioned in the parameter file.

2. Configure the stack names for the two regions created using CFT.
If the default value of the region fields are changed, then change the geo-location values accordingly.

```
"stackRegionDetails":{
  "value" : [
    {
      "region": "us-east-1",
      "stackName":"vth-stack1"
    },
    {
      "region": "us-east-2",
      "stackName":"vth-stack2"
    }
  ]
},
```

3. Configure SLB server port for site devices.

```
"parameters": {
  "slbServerPortList1": {
    "value": [
      {
        "port-number": 80,
        "protocol": "tcp",
        "health-check-disable": 0
      }
    ]
  },
  "slbServerPortList2": {
    "value": [
      {
        "port-number": 80,
        "protocol": "tcp",
        "health-check-disable": 0
      }
    ]
  },
  "slbServerPortList3": {
    "value": [
      {
        "port-number": 80,
        "protocol": "tcp",
        "health-check-disable": 0
      }
    ]
  },
  "slbServerPortList4": {
    "value": [
      {
        "port-number": 80,
        "protocol": "tcp",
        "health-check-disable": 0
      }
    ]
  },
}
```

4. Configure service group port for site devices.

```
"serviceGroupList1": {
  "value": [
    {
      "name": "sg",
      "protocol": "tcp",
      "health-check-disable": 0,
      "member-list": [
        {
          "port": 80
        }
      ]
    }
  ]
},
"serviceGroupList2": {
  "value": [
    {
      "name": "sg",
      "protocol": "tcp",
      "health-check-disable": 0,
      "member-list": [
        {
          "port": 80
        }
      ]
    }
  ]
},
"serviceGroupList3": {
  "value": [
    {
      "name": "sg",
      "protocol": "tcp",
      "health-check-disable": 0,
      "member-list": [
        {
```

```

        "port":80
      }
    ]
  }
],
"serviceGroupList4": {
  "value": [
    {
      "name":"sg",
      "protocol":"tcp",
      "health-check-disable":0,
      "member-list": [
        {
          "port":80
        }
      ]
    }
  ]
},

```

5. Configure SLB virtual server for site devices.
The virtual server default name is “vs1”.

```

"virtualServerList1": {
  "virtual-server-name": "vs1",
  "metadata": {
    "description": "virtual server is using VIP from
ethernet 1 secondary subnet"
  },
  "value": [
    {
      "port-number":80,
      "protocol":"tcp",
      "auto":1,
      "service-group":"sg"
    }
  ]
}

```

```

    },
    "virtualServerList2": {
      "virtual-server-name": "vs1",
      "metadata": {
        "description": "virtual server is using VIP from
ethernet 1 secondary subnet"
      },
      "value": [
        {
          "port-number": 80,
          "protocol": "tcp",
          "auto": 1,
          "service-group": "sg"
        }
      ]
    },
    "virtualServerList3": {
      "virtual-server-name": "vs1",
      "metadata": {
        "description": "virtual server is using VIP from
ethernet 1 secondary subnet"
      },
      "value": [
        {
          "port-number": 80,
          "protocol": "tcp",
          "auto": 1,
          "service-group": "sg"
        }
      ]
    },
    "virtualServerList4": {
      "virtual-server-name": "vs1",
      "metadata": {
        "description": "virtual server is using VIP from
ethernet 1 secondary subnet"
      },

```

```
    "value": [
      {
        "port-number":80,
        "protocol":"tcp",
        "auto":1,
        "service-group":"sg"
      }
    ]
  },
```

6. Configure GSLB service IP address for controller.

```
"serviceipList1": {
  "node-name": "vs1",
  "value": [
    {
      "port-num": 80,
      "port-proto": "tcp"
    }
  ]
},
"serviceipList2": {
  "node-name": "vs2",
  "value": [
    {
      "port-num": 80,
      "port-proto": "tcp"
    }
  ]
},
"serviceipList3": {
  "node-name": "vs3",
  "value": [
    {
      "port-num": 80,
      "port-proto": "tcp"
    }
  ]
}
```

```

    },
    "serviceipList4": {
      "node-name": "vs4",
      "value": [
        {
          "port-num": 80,
          "port-proto": "tcp"
        }
      ]
    }
  },

```

7. Configure GSLB site details for controller.

```

"siteList1": {
  "site-name": "eastus_1",
  "vip-name": "vs1",
  "device-name": "slb1",
  "geo-location": "North America,United States"
},
"siteList2": {
  "site-name": "eastus_2",
  "vip-name": "vs2",
  "device-name": "slb2",
  "geo-location": "North America,United States"
},
"siteList3": {
  "site-name": "eastus2_1",
  "vip-name": "vs3",
  "device-name": "slb3",
  "geo-location": "North America.United States.California.San
Jose"
},
"siteList4": {
  "site-name": "eastus2_2",
  "vip-name": "vs4",
  "device-name": "slb4",
  "geo-location": "North America.United States.California.San

```



```
Jose"
    },
```

8. Configure system geo location details for controller.

```
"geolocation": {
    "geo-location-iana": "0",
    "geo-location-geolite2-city": "1",
    "geolite2-city-include-ipv6": "0",
    "geo-location-geolite2-country": "0"
},
```

9. Configure GSLB DNS policy for controller.

```
"dnsPolicy": {
    "policy-name": "a10",
    "type": "health-check, geographic"
},
```

10. Configure GSLB virtual server for controller.

```
"gslbserverList1": {
    "virtual-server-name": "gslb-server",
    "use-if-ip": 1,
    "ethernet": 1,
    "metadata": {
        "description": "gslb virtual server is using VIP from
ethernet 1 secondary subnet"
    },
    "value": [
        {
            "port-number": 53,
            "protocol": "udp",
            "gslb-enable": 1
        }
    ]
},
"gslbserverList2": {
    "virtual-server-name": "gslb-server",
    "use-if-ip": 1,
    "ethernet": 1,
```

```

        "metadata": {
            "description": "gslb virtual server is using VIP from
ethernet 1 secondary subnet"
        },
        "value": [
            {
                "port-number": 53,
                "protocol": "udp",
                "gslb-enable": 1
            }
        ]
    },

```

11. Configure GSLB protocol status for controller.

```

    "gslbprotocolStatus": {
        "status-interval": 1
    },

```

12. Configure GSLB group for controller.

```

    "gslbcontrollerGroup1": {
        "name": "default",
        "priority": 255
    },
    "gslbcontrollerGroup2": {
        "name": "default",
        "priority": 100
    },

```

13. Configure GSLB zone for controller.

```

    "gslbzone": {
        "service-port": 80,
        "service-name": "www",
        "name" : "gslb.a10.com"
    },

```

14. Configure default route for vThunder instances.

```
"defaultroute1":
{
    "next-hop": "10.1.2.1"
},
"defaultroute2":
{
    "next-hop": "10.2.2.1"
}
```

15. Verify if all the configurations in the CFT_TMPL_3NIC_6VM_2RG_GSLB_CONFIG_GSLB_PARAM.json file are correct and then save the changes.

Deploy vThunder as an SLB

To deploy vThunder on AWS cloud as an SLB, perform the following steps:

1. From your command prompt, navigate to the folder where you have downloaded the CFT.
2. Run the following command to create vThunder SLB instances:

```
python ./CFT_TMPL_3NIC_6VM_2RG_GSLB_CONFIG_GSLB_3.py
```

A message is prompted that displays the criteria for a valid admin user password.

```
Primary conditions for password validation, user should provide the new
password according to the given combination:
```

```
Minimum length of 9 characters
Minimum lowercase character should be 1
Minimum uppercase character should be 1
Minimum number should be 1
Minimum special character should be 1
Should not include repeated characters
Should not include more than 3 keyboard consecutive characters.
```

A message is prompted to enter a new password.

3. Provide the new password for the admin user and then confirm the new password.

The new password should follow the valid admin user password criteria mentioned in the above step. For more information, see [Default Password Policy](#).

```
Enter vThunder's new password:
Confirm new password:
```

If both the password and confirm password field values match and complies to the valid admin user password criteria, a message 'Password changed successfully' is displayed along with the other configurations.

```
Password changed successfully for site devices.
Password changed successfully for controller devices.
-----
-----
configured ethernet- 1 ip
configured ethernet- 2 ip
Configuring slb server for site: i-0860344bc58ad0a15
Successfully Configured slb server for site: i-0860344bc58ad0a15
Configuring service group for site: i-0860344bc58ad0a15
Successfully Configured service group for site:i-0860344bc58ad0a15
Successfully Configured virtual server for site: i-0860344bc58ad0a15
Successfully Configured gslb site: i-0860344bc58ad0a15
Successfully Configured default route:i-0860344bc58ad0a15
Configurations are saved on partition: shared
-----
-----
configured ethernet- 1 ip
configured ethernet- 2 ip
Configuring slb server for site: i-0e794e0e05b0b08db
Successfully Configured slb server for site: i-0e794e0e05b0b08db
Configuring service group for site: i-0e794e0e05b0b08db
Successfully Configured service group for site:i-0e794e0e05b0b08db
Successfully Configured virtual server for site: i-0e794e0e05b0b08db
Successfully Configured gslb site: i-0e794e0e05b0b08db
Successfully Configured default route:i-0e794e0e05b0b08db
Configurations are saved on partition: shared
-----
-----
configured ethernet- 1 ip
```

```
configured ethernet- 2 ip
Configuring slb server for site: i-0cd1ea57124cd8bd8
Successfully Configured slb server for site: i-0cd1ea57124cd8bd8
Configuring service group for site: i-0cd1ea57124cd8bd8
Successfully Configured service group for site:i-0cd1ea57124cd8bd8
Successfully Configured virtual server for site: i-0cd1ea57124cd8bd8
Successfully Configured gslb site: i-0cd1ea57124cd8bd8
Successfully Configured default route:i-0cd1ea57124cd8bd8
Configurations are saved on partition: shared
-----

configured ethernet- 1 ip
configured ethernet- 2 ip
Configuring slb server for site: i-0807dclb0152de941
Successfully Configured slb server for site: i-0807dclb0152de941
Configuring service group for site: i-0807dclb0152de941
Successfully Configured service group for site:i-0807dclb0152de941
Successfully Configured virtual server for site: i-0807dclb0152de941
Successfully Configured gslb site: i-0807dclb0152de941
Successfully Configured default route:i-0807dclb0152de941
Configurations are saved on partition: shared
-----

Configuring controller devices
configured ethernet- 1 ip
configured ethernet- 2 ip
Successfully Configured gslb server for controller: i-095a405f97fbef0e2
Successfully Configured ServiceIp for site: i-095a405f97fbef0e2
Successfully Configured ServiceIp for site: i-095a405f97fbef0e2
Successfully Configured ServiceIp for site: i-095a405f97fbef0e2
Successfully Configured ServiceIp for site: i-095a405f97fbef0e2
Successfully Configured site information for: i-095a405f97fbef0e2
Successfully Configured site information for: i-095a405f97fbef0e2
Successfully Configured site information for: i-095a405f97fbef0e2
Successfully Configured site information for: i-095a405f97fbef0e2
Successfully Configured gslb policy for :i-095a405f97fbef0e2
Successfully Configured gslb zone for :i-095a405f97fbef0e2
```

```
Successfully Configured gslb controller and status interval: i-095a405f97fbef0e2
Successfully Configured gslb controller group: i-095a405f97fbef0e2
Successfully Configured geo location: i-095a405f97fbef0e2
Successfully Configured default route:i-095a405f97fbef0e2
Configurations are saved on partition: shared
configured ethernet- 1 ip
configured ethernet- 2 ip
Successfully Configured gslb server for controller: i-043bc788995faa4c6
Successfully Configured gslb controller group: i-043bc788995faa4c6
Successfully Configured default route:i-043bc788995faa4c6
Configurations are saved on partition: shared
```

4. Verify the following for each site devices:

- Interfaces are enabled
- SLB is configured
- Site device is enabled to be a GSLB device.
- Default route is configured pointing to the client-side data interface for traffic to exit the vThunder.

5. Verify the following for each GSLB controller:

- Interfaces are enabled
- vThunder device is configured with the required GSLB configuration
- Geo location is enabled.
- Default route is configured pointing to the client-side data interface for traffic to exit the vThunder.

Verify Deployment

To verify deployment using CFT, perform the following steps:

1. Access the vThunder instances using CLI.
2. Verify SLB configuration on the following vThunder instances:

CONTROLLER - Master configuration

Run the following command:

```
vThunder-gslb:Master(config) (NOLICENSE) #show running-config
```

If the deployment is successful, the following controller and site configuration is displayed on vThunder master controller:

```
!Current configuration: 246 bytes
!Configuration last updated at 11:58:47 GMT Mon Jan 9 2023
!Configuration last saved at 11:58:51 GMT Mon Jan 9 2023
!64-bit Advanced Core OS (ACOS) version 5.2.1, build 153 (Dec-11-
2020,16:36)
!
!
no system geo-location load iana
system geo-location load GeoLite2-City
!
!
interface ethernet 1
    enable
    ip address dhcp
!
interface ethernet 2
    enable
    ip address dhcp
!
!
ip route 0.0.0.0 /0 10.1.2.1
ip route 0.0.0.0 /0 10.1.1.1
!
slb virtual-server gslb-server 10.1.2.121
    port 53 udp
    gslb-enable
!
```

```
gslb service-ip vs1 10.1.2.123
    external-ip 35.153.250.242
    port 80 tcp
!
gslb service-ip vs2 10.1.2.124
    external-ip 44.208.102.77
    port 80 tcp
!
gslb service-ip vs3 10.2.2.123
    external-ip 18.188.27.110
    port 80 tcp
!
gslb service-ip vs4 10.2.2.124
    external-ip 3.14.157.128
    port 80 tcp
!
gslb group default
    enable
    priority 255
!
gslb site eastus_1
    geo-location "North America,United States"
    slb-dev slb1 107.21.185.247
    vip-server vs1
!
gslb site eastus_2
    geo-location "North America,United States"
    slb-dev slb2 3.220.107.48
    vip-server vs2
!
gslb site eastus2_1
    geo-location "North America.United States.California.San Jose"
    slb-dev slb3 18.116.22.194
    vip-server vs3
!
gslb site eastus2_2
```



```

    geo-location "North America.United States.California.San Jose"
    slb-dev slb4 3.134.234.243
        vip-server vs4
!
gslb policy a10
    metric-order health-check geographic
    dns server authoritative
!
gslb zone gslb.a10.com
    policy a10
        service 80 www
            dns-a-record vs1 static
            dns-a-record vs2 static
            dns-a-record vs3 static
            dns-a-record vs4 static
!
gslb protocol status-interval 1
!
gslb protocol enable controller
!
!
end

!Current config commit point for partition 0 is 0 & config mode is
classical-mode

```

CONTROLLER - Member configuration

Run the following command:

```
vThunder-gslb:Member(config) (NOLICENSE) #show running-config
```

If the deployment is successful, the following controller and site configuration is displayed on vThunder member controller:

```

!Current configuration: 182 bytes
!Configuration last updated at 11:59:07 GMT Mon Jan 9 2023
!Configuration last saved at 11:59:05 GMT Mon Jan 9 2023
!64-bit Advanced Core OS (ACOS) version 5.2.1, build 153 (Dec-11-
2020,16:36)

```

```
!  
!  
interface ethernet 1  
    enable  
    ip address dhcp  
!  
interface ethernet 2  
    enable  
    ip address dhcp  
!  
!  
ip route 0.0.0.0 /0 10.2.2.1  
ip route 0.0.0.0 /0 10.2.1.1  
!  
slb virtual-server gslb-server 10.2.2.121  
    port 53 udp  
    gslb-enable  
!  
gslb service-ip vs1 10.1.2.123  
    external-ip 35.153.250.242  
    port 80 tcp  
!  
gslb service-ip vs2 10.1.2.124  
    external-ip 44.208.102.77  
    port 80 tcp  
!  
gslb service-ip vs3 10.2.2.123  
    external-ip 18.188.27.110  
    port 80 tcp  
!  
gslb service-ip vs4 10.2.2.124  
    external-ip 3.14.157.128  
    port 80 tcp  
!  
gslb group default  
    enable
```

```
primary 18.209.143.246
!
gslb site eastus_1
  geo-location "North America,United States"
  slb-dev slb1 107.21.185.247
  vip-server vs1
!
gslb site eastus_2
  geo-location "North America,United States"
  slb-dev slb2 3.220.107.48
  vip-server vs2
!
gslb site eastus2_1
  geo-location "North America.United States.California.San Jose"
  slb-dev slb3 18.116.22.194
  vip-server vs3
!
gslb site eastus2_2
  geo-location "North America.United States.California.San Jose"
  slb-dev slb4 3.134.234.243
  vip-server vs4
!
gslb policy a10
  metric-order health-check geographic
  dns server authoritative
!
gslb zone gslb.a10.com
  policy a10
  service 80 www
    dns-a-record vs1 static
    dns-a-record vs2 static
    dns-a-record vs3 static
    dns-a-record vs4 static
!
gslb protocol status-interval 1
!
```

```
gslb protocol enable controller
!
!
end
!Current config commit point for partition 0 is 0 & config mode is
classical-mode
```

SITE1 REGION1 configuration

Run the following command:

```
vThunder(config) (NOLICENSE) #show running-config
```

If the deployment is successful, the following controller and site configuration is displayed on vThunder site1 region1:

```
!Current configuration: 89 bytes
!Configuration last updated at 11:57:22 GMT Mon Jan 9 2023
!Configuration last saved at 11:57:26 GMT Mon Jan 9 2023
!64-bit Advanced Core OS (ACOS) version 5.2.1, build 153 (Dec-11-
2020,16:36)
!
!
interface ethernet 1
    enable
    ip address dhcp
!
interface ethernet 2
    enable
    ip address dhcp
!
!
ip route 0.0.0.0 /0 10.1.2.1
ip route 0.0.0.0 /0 10.1.1.1
!
slb server vth-stack1-server1 10.1.3.52
    health-check-disable
    port 80 tcp
    health-check-disable
```

```

!
slb service-group sg tcp
    member vth-stack1-server1 80
!
slb virtual-server vs1 10.1.2.123
    port 80 tcp
    source-nat auto
    service-group sg
!
!
gslb protocol enable device
!
!
end
!Current config commit point for partition 0 is 0 & config mode is
classical-mode

```

SITE2 REGION1 configuration

Run the following command:

```
vThunder(config) (NOLICENSE) #show running-config
```

If the deployment is successful, the following controller and site configuration is displayed on vThunder site2 region1:

```

!Current configuration: 89 bytes
!Configuration last updated at 11:57:39 GMT Mon Jan 9 2023
!Configuration last saved at 11:57:43 GMT Mon Jan 9 2023
!64-bit Advanced Core OS (ACOS) version 5.2.1, build 153 (Dec-11-
2020,16:36)
!
!
interface ethernet 1
    enable
    ip address dhcp
!
interface ethernet 2
    enable

```

```

ip address dhcp
!
!
ip route 0.0.0.0 /0 10.1.2.1
ip route 0.0.0.0 /0 10.1.1.1
!
slb server vth-stack1-server2 10.1.3.250
    health-check-disable
    port 80 tcp
        health-check-disable
!
slb service-group sg tcp
    member vth-stack1-server2 80
!
slb virtual-server vs1 10.1.2.124
    port 80 tcp
        source-nat auto
        service-group sg
!
!
gslb protocol enable device
!
!
end
!Current config commit point for partition 0 is 0 & config mode is
classical-mode

```

SITE1 REGION2 configuration

Run the following command:

```
vThunder(config) (NOLICENSE) #show running-config
```

If the deployment is successful, the following controller and site configuration is displayed on vThunder site1 region2 :

```

!Current configuration: 89 bytes
!Configuration last updated at 11:57:55 GMT Mon Jan 9 2023
!Configuration last saved at 11:57:59 GMT Mon Jan 9 2023

```

```
!64-bit Advanced Core OS (ACOS) version 5.2.1, build 153 (Dec-11-
2020,16:36)
!
!
interface ethernet 1
    enable
    ip address dhcp
!
interface ethernet 2
    enable
    ip address dhcp
!
!
ip route 0.0.0.0 /0 10.2.2.1
ip route 0.0.0.0 /0 10.2.1.1
!
slb server vth-stack2-server3 10.2.3.179
    health-check-disable
    port 80 tcp
    health-check-disable
!
slb service-group sg tcp
    member vth-stack2-server3 80
!
slb virtual-server vs1 10.2.2.123
    port 80 tcp
    source-nat auto
    service-group sg
!
!
gslb protocol enable device
!
!
end

!Current config commit point for partition 0 is 0 & config mode is
classical-mode
```

SITE2 REGION2 configuration

Run the following command:

```
vThunder(config) (NOLICENSE) #show running-config
```

If the deployment is successful, the following controller and site configuration is displayed on vThunder site2 region2 :

```
!Current configuration: 89 bytes
!Configuration last updated at 11:58:10 GMT Mon Jan 9 2023
!Configuration last saved at 11:58:14 GMT Mon Jan 9 2023
!64-bit Advanced Core OS (ACOS) version 5.2.1, build 153 (Dec-11-
2020,16:36)
!
!
interface ethernet 1
    enable
    ip address dhcp
!
interface ethernet 2
    enable
    ip address dhcp
!
!
ip route 0.0.0.0 /0 10.2.2.1
ip route 0.0.0.0 /0 10.2.1.1
!
slb server vth-stack2-server4 10.2.3.254
    health-check-disable
    port 80 tcp
    health-check-disable
!
slb service-group sg tcp
    member vth-stack2-server4 80
!
slb virtual-server vs1 10.2.2.124
    port 80 tcp
```



```
source-nat auto
service-group sg
!
!
gslb protocol enable device
!
!
end
!Current config commit point for partition 0 is 0 & config mode is
classical-mode
```

3. Verify GSLB group information on the following vThunder instances:

CONTROLLER - Master configuration

- a. Run the following command:

```
vThunder-gslb:Master(NOLICENSE) #show gslb group
```

- b. Verify if the public ip of member controller is displayed in the response:

```
Pri = Priority, Attrs = Attributes
S-Cfg = Secure Config
S-State = Secure Status
D = Disabled, L = Learn
P = Passive, * = Master
E = Enabled, EF = Enable-Fallback
Unsec = Unsecure, Unkwn = Unknown
Estng = Establishing, Estd = Established
Group: default, Master: local
Member                               Sys-ID  Pri Attrs  Status  S-
Cfg S-State Address
-----
local                               7b8a5001 255 L*      OK
vThunder                             58dd5c28 100 PL     Synced  D
Unsec 13.58.227.170
```

CONTROLLER - Member configuration

a. Run the following command:

```
vThunder-gslb:Member(NOLICENSE) #show gslb group
```

Verify if the public ip of master controller is displayed in the response:

```

Pri = Priority, Attrs = Attributes
S-Cfg = Secure Config
S-State = Secure Status
    D = Disabled, L = Learn
    P = Passive, * = Master
    E = Enabled, EF = Enable-Fallback
    Unsec = Unsecure, Unkwn = Unknown
    Estng = Establishing, Estd = Established
Group: default, Master: vThunder
Member                               Sys-ID   Pri Attrs   Status      S-
Cfg S-State Address
-----
-----
local                               58dd5c28 100 L         OK
vThunder                           7b8a5001 255 L*      Synced      D
    Unsec    18.209.143.246
    
```

4. Verify the GSLB protocol information on the following vThunder instances:

CONTROLLER - Master configuration

Run the following command:

```
vThunder-gslb:Master(NOLICENSE) #show gslb protocol
```

The following configuration is displayed on vThunder master controller:

```

GSLB site: eastus_1
  SLB device: slb1 (10.1.1.170:10578) Established
  Session ID:    7629
  Secure Config:          Disable |Current SSL State:
                        Unsecure
  Connection succeeded:    1 |Connection failed:
                        0
  Open packet sent:       1 |Open packet received:
    
```

```

1
Open session succeeded:      1 |Open session failed:
0
Sessions Dropped:          0 |Update packet received:
1320
Keepalive packet sent:     23 |Keepalive packet
received:                  22
Notify packet sent:        0 |Notify packet received:
0
Message Header Error:      0 |Protocol RDT(ms):
0
GSLB Protocol Version:     2 |Peer ACOS Version:
5.2.1 Build 153
Secure negotiation Success: 0 |Secure negotiation
Failures:                  0
SSL handshake Success:     0 |SSL handshake Failures:
0

GSLB site: eastus_2
  SLB device: slb2 (10.1.1.170:4776) Established
  Session ID: 20481
  Secure Config:           Disable |Current SSL State:
  Unsecure
  Connection succeeded:     1 |Connection failed:
0
  Open packet sent:        1 |Open packet received:
1
  Open session succeeded:   1 |Open session failed:
0
  Sessions Dropped:       0 |Update packet received:
1320
  Keepalive packet sent:   23 |Keepalive packet
received:                  22
  Notify packet sent:      0 |Notify packet received:
0
  Message Header Error:    0 |Protocol RDT(ms):

```

```

0
GSLB Protocol Version:          2 |Peer ACOS Version:
    5.2.1 Build 153
Secure negotiation Success:      0 |Secure negotiation
Failures:                        0
    SSL handshake Success:       0 |SSL handshake Failures:
                                0

GSLB site: eastus2_1
    SLB device: slb3 (10.1.1.170:3352) Established
    Session ID:      25287
    Secure Config:          Disable |Current SSL State:
                            Unsecure
    Connection succeeded:    1 |Connection failed:
                            0
    Open packet sent:       1 |Open packet received:
                            1
    Open session succeeded:  1 |Open session failed:
                            0
    Sessions Dropped:       0 |Update packet received:
                            1320
    Keepalive packet sent:   23 |Keepalive packet
received:                    22
    Notify packet sent:      0 |Notify packet received:
                            0
    Message Header Error:    0 |Protocol RDT(ms):
                            12
    GSLB Protocol Version:  2 |Peer ACOS Version:
        5.2.1 Build 153
    Secure negotiation Success: 0 |Secure negotiation
Failures:                    0
    SSL handshake Success:   0 |SSL handshake Failures:
                            0

GSLB site: eastus2_2
    SLB device: slb4 (10.1.1.170:19222) Established

```

```

Session ID:      6077
Secure Config:      Disable |Current SSL State:
                    Unsecure
Connection succeeded:      1 |Connection failed:
                           0
Open packet sent:      1 |Open packet received:
                           1
Open session succeeded:      1 |Open session failed:
                           0
Sessions Dropped:      0 |Update packet received:
                           1406
Keepalive packet sent:      24 |Keepalive packet
received:      23
Notify packet sent:      0 |Notify packet received:
                           0
Message Header Error:      0 |Protocol RDT(ms):
                           12
GSLB Protocol Version:      2 |Peer ACOS Version:
                           5.2.1 Build 153
Secure negotiation Success:      0 |Secure negotiation
Failures:      0
SSL handshake Success:      0 |SSL handshake Failures:
                           0

GSLB protocol is disabled for site devices.

```

CONTROLLER - Member configuration

Run the following command:

```
vThunder-gslb:Member (NOLICENSE) #show gslb protocol
```

The following configuration is displayed on vThunder member controller:

```

GSLB site: eastus_1
  SLB device: slb1 (0.0.0.0:0) GroupControl
Session ID:      Not Available
Secure Config:      None |Current SSL State:
                    None

```

```

Connection succeeded:          0 |Connection failed:
                                0
Open packet sent:             0 |Open packet received:
                                0
Open session succeeded:       0 |Open session failed:
                                0
Sessions Dropped:             0 |Update packet received:
                                0
Keepalive packet sent:        0 |Keepalive packet
received:                      0
Notify packet sent:           0 |Notify packet received:
                                0
Message Header Error:         0 |Protocol RDT(ms):
                                0
GSLB Protocol Version:        2
Secure negotiation Success:    0 |Secure negotiation
Failures:                     0
SSL handshake Success:         0 |SSL handshake Failures:
                                0

GSLB site: eastus_2
  SLB device: slb2 (0.0.0.0:0) GroupControl
  Session ID:    Not Available
  Secure Config:          None |Current SSL State:
                          None
Connection succeeded:      0 |Connection failed:
                          0
Open packet sent:         0 |Open packet received:
                          0
Open session succeeded:   0 |Open session failed:
                          0
Sessions Dropped:        0 |Update packet received:
                          0
Keepalive packet sent:    0 |Keepalive packet
received:                 0
Notify packet sent:       0 |Notify packet received:

```

```

0
Message Header Error: 0 |Protocol RDT(ms):
0
GSLB Protocol Version: 2
Secure negotiation Success: 0 |Secure negotiation
Failures: 0
SSL handshake Success: 0 |SSL handshake Failures:
0

GSLB site: eastus2_1
  SLB device: slb3 (0.0.0.0:0) GroupControl
  Session ID: Not Available
  Secure Config: None |Current SSL State:
None
Connection succeeded: 0 |Connection failed:
0
Open packet sent: 0 |Open packet received:
0
Open session succeeded: 0 |Open session failed:
0
Sessions Dropped: 0 |Update packet received:
0
Keepalive packet sent: 0 |Keepalive packet
received: 0
Notify packet sent: 0 |Notify packet received:
0
Message Header Error: 0 |Protocol RDT(ms):
0
GSLB Protocol Version: 2
Secure negotiation Success: 0 |Secure negotiation
Failures: 0
SSL handshake Success: 0 |SSL handshake Failures:
0

GSLB site: eastus2_2
  SLB device: slb4 (0.0.0.0:0) GroupControl

```

```

Session ID:      Not Available
Secure Config:      None |Current SSL State:
                        None
Connection succeeded:      0 |Connection failed:
                        0
Open packet sent:      0 |Open packet received:
                        0
Open session succeeded:      0 |Open session failed:
                        0
Sessions Dropped:      0 |Update packet received:
                        0
Keepalive packet sent:      0 |Keepalive packet
received:      0
Notify packet sent:      0 |Notify packet received:
                        0
Message Header Error:      0 |Protocol RDT(ms):
                        0
GSLB Protocol Version:      2
Secure negotiation Success:      0 |Secure negotiation
Failures:      0
SSL handshake Success:      0 |SSL handshake Failures:
                        0

GSLB protocol is disabled for site devices.

```

NOTE: If you want to access the vThunder instances using GUI, re-login the instances using the new admin user password and NOT the *EC2 Instance ID*.

Verify Traffic Flow

The traffic flow can be tested using the following:

- [DNS Lookup](#)
- [WGET](#)

DNS Lookup

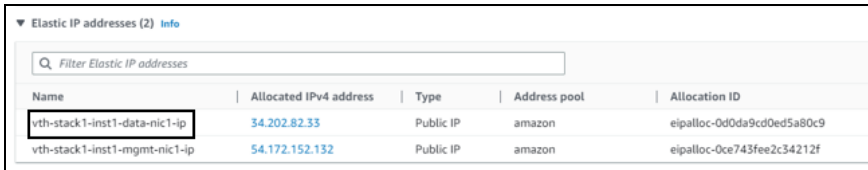
To verify the traffic flow from via vThunder, perform the following:

1. Perform a DNS lookup on server1 of region1 using the master controller's client-side data interface public IP in the following command:

```
$ dig @master_controller_data_public_IP www.gslb.a10.com
```

The master controller's client-side data interface public IP is used as DNS server IP. You can get this data interface public IP from **AWS Management Console** > **EC2** > **Instances** > <stack_name_master_controller_region1> > **Networking** > **Elastic IP address**.

Figure 75 : Master Controller Data Interface Public IP



Name	Allocated IPv4 address	Type	Address pool	Allocation ID
vth-stack1-inst1-data-nic1-ip	34.202.82.33	Public IP	amazon	eipalloc-0d0da9cd0ed5a80c9
vth-stack1-inst1-mgmt-nic1-ip	54.172.152.132	Public IP	amazon	eipalloc-0ce743fee2c34212f

The following response is received:

```
$ dig @34.202.82.33 www.gslb.a10.com
; <<>> DiG 9.18.1-Ubuntu <<>> @34.202.82.33 www.gslb.a10.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 62463
;; flags: qr aa rd; QUERY: 1, ANSWER: 4, AUTHORITY: 0, ADDITIONAL: 1
;; WARNING: recursion requested but not available

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1400
;; QUESTION SECTION:
;www.gslb.a10.com.                IN      A

;; ANSWER SECTION:
www.gslb.a10.com.                10      IN      A      34.199.187.33
www.gslb.a10.com.                10      IN      A      3.232.227.57
```

```

www.gslb.a10.com.      10      IN      A      3.140.125.3
www.gslb.a10.com.      10      IN      A      3.16.234.54

;; Query time: 0 msec
;; SERVER: 34.202.82.33#53(34.202.82.33) (UDP)
;; WHEN: Mon Jan 09 17:43:25 IST 2023
;; MSG SIZE rcvd: 125

```

2. Perform the DNS lookup again.

```

$ dig @34.202.82.33 www.gslb.a10.com
; <<>> DiG 9.18.1-Ubuntu <<>> @34.202.82.33 www.gslb.a10.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<- opcode: QUERY, status: NOERROR, id: 62463
;; flags: qr aa rd; QUERY: 1, ANSWER: 4, AUTHORITY: 0, ADDITIONAL: 1
;; WARNING: recursion requested but not available

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:: udp: 1400
;; QUESTION SECTION:
;www.gslb.a10.com.      IN      A

;; ANSWER SECTION:
www.gslb.a10.com.      10      IN      A      3.232.227.57
www.gslb.a10.com.      10      IN      A      34.199.187.33
www.gslb.a10.com.      10      IN      A      3.140.125.3
www.gslb.a10.com.      10      IN      A      3.16.234.54

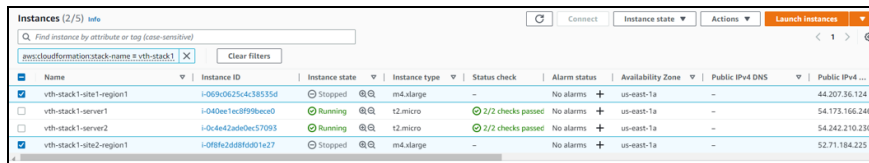
;; Query time: 0 msec
;; SERVER: 34.202.82.33#53(34.202.82.33) (UDP)
;; WHEN: Mon Jan 09 17:44:25 IST 2023
;; MSG SIZE rcvd: 125

```

The response is received with shuffled server IP addresses.

3. Stop the site instances of region1.

Figure 76 : Stopped Site instances



Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4...
vth-stack1-site1-region1	i-069b0625c4c385356	Stopped	m4.large	-	No alarms	us-east-1a	-	44.207.36.124
vth-stack1-server1	i-040ee1ec09990ec00	Running	t2.micro	2/2 checks passed	No alarms	us-east-1a	-	54.173.166.246
vth-stack1-server2	i-0e4e42ade0ec57093	Running	t2.micro	2/2 checks passed	No alarms	us-east-1a	-	54.242.210.230
vth-stack1-site2-region1	i-089b2d89d0d1e27	Stopped	m4.large	-	No alarms	us-east-1a	-	52.71.184.225

- Perform the DNS lookup to verify if you receive a response after stopping the site instances.

```
$ dig @34.202.82.33 www.gslb.a10.com
; <<>> DiG 9.18.1-Ubuntu <<>> @34.202.82.33 www.gslb.a10.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<- opcode: QUERY, status: NOERROR, id: 62463
;; flags: qr aa rd; QUERY: 1, ANSWER: 4, AUTHORITY: 0, ADDITIONAL: 1
;; WARNING: recursion requested but not available

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1400
;; QUESTION SECTION:
;www.gslb.a10.com.                IN      A

;; ANSWER SECTION:
www.gslb.a10.com.                10      IN      A      3.140.125.3
www.gslb.a10.com.                10      IN      A      3.16.234.54
www.gslb.a10.com.                10      IN      A      3.232.227.57
www.gslb.a10.com.                10      IN      A      34.199.187.33

;; Query time: 0 msec
;; SERVER: 34.202.82.33#53(34.202.82.33) (UDP)
;; WHEN: Mon Jan 09 17:46:25 IST 2023
;; MSG SIZE rcvd: 125
```

The response is received with site devices secondary data1 public IPs based on round robin.

WGET

To verify the traffic flow via vThunder, perform the following:

1. From **AWS Management Console**, navigate to **EC2 > Instances**.
2. Select any one of the server instance assigned to the site devices.

Here, `vth-stack1-server1` is the server instance name.

3. Click **Connect**.

A **Connect to instance** window with **EC2 Instance Connect** tab is displayed.

4. Click **Connect**.

A **Terminal** window is displayed.

5. Run the following command in the Terminal window to create an Apache Server virtual machine:

```
$ sudo apt install apache2
```

While the Apache server is getting installed, you get a prompt to continue further. Enter 'Y' to continue. After the installation is complete, a newline prompt is displayed.

6. From **AWS Management Console > EC2 > Instances**, select the site instance of the corresponding server on which Apache was installed.

Here, `vth-stack1-site1-region1` is the site instance.

7. Navigate to **Networking** tab > **Elastic IP addresses** and copy the **Allocated IPv4 address** of site instance data interface.

Figure 77 : Site Instance Data Interface Public IP

Instance: i-0dd6dc4a32d78c35b (vth-stack1-site1-region1)

Public IPv4 DNS
-

Subnet ID
 subnet-0050dd35ac862fd77 (vth-stack1-vpc-mgmt-sub1) 

Availability zone
 us-east-1a

Use RBN as guest OS hostname
 Disabled

► Network Interfaces (3) [Info](#)

▼ Elastic IP addresses (2) [Info](#)

Name	Allocated IPv4 address	Type	Address pool
vth-stack1-inst2-mgmt-nic1-ip	107.21.185.247	Public IP	amazon
vth-stack1-inst2-data1-nic1-ip	35.153.250.242	Public IP	amazon

Additional instance details (right sidebar):

- Public IPv4 address:  10.1.3.119
- Private IP DNS name (IPv4 only):  ip-10-1-1-128.ec2.internal
- IPv6 addresses: -
- Carrier IP addresses (ephemeral): -
- Answer RBN DNS hostname IPv4:  Disabled

8. Run the following command on the server1 of region1:

```
$ wget site_device_data-public-IP
```

The following response is received:

```
$ wget 35.153.250.242
--2023-01-09 17:49:47-- http://35.153.250.242/
Connecting to 35.153.250.242:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 10671 (10K) [text/html]
Saving to: 'index.html.3'

index.html.3                               100%
[=====]
=====] 10.42K --.-KB/s in 0s

2023-01-09 17:49:47 (63.8 MB/s) - 'index.html.3' saved [10671]
```

Appendix

The following topics are covered:

Security Policy for AWS User	214
Default Password Policy	217

Security Policy for AWS User

To deploy the vThunder instance using a CFT template, an AWS user requires certain security policies. The following security policies are recommended:

Predefined

- AmazonEC2FullAccess
- AmazonS3FullAccess
- AmazonS3ObjectLambdaExecutionRolePolicy
- AmazonVPCFullAccess

Custom

- Create and Edit Secrets

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "secretsmanager:CreateSecret",
        "secretsmanager:PutSecretValue"
      ]
    }
  ]
}
```

```

        "Resource": "arn:aws:secretsmanager:us-east-1:939850196882:secret:*"
    }
]
}

```

- **Lambda Update**

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ConfigureFunctions",
      "Effect": "Allow",
      "Action": [
        "lambda:UpdateFunctionConfiguration",
        "lambda:GetFunction"
      ],
      "Resource": "*",
      "Condition": {
        "ForAllValues:StringLike": {
          "lambda:Layer": [
            "arn:aws:lambda:*:939850196882:layer:*:*"
          ]
        }
      }
    }
  ]
}

```

- **Manage Secrets**

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "secretsmanager:GetResourcePolicy",
        "secretsmanager:GetSecretValue",

```

```
        "secretsmanager:DescribeSecret",
        "secretsmanager:ListSecretVersionIds"
    ],
    "Resource": "arn:aws:secretsmanager:us-east-1:939850196882:secret:*"
  },
  {
    "Effect": "Allow",
    "Action": "secretsmanager:ListSecrets",
    "Resource": "*"
  }
]
```

- Cloud Watch Logs and Streams

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "secretsmanager:GetResourcePolicy",
        "secretsmanager:GetSecretValue",
        "secretsmanager:DescribeSecret",
        "secretsmanager:ListSecretVersionIds"
      ],
      "Resource": "arn:aws:secretsmanager:us-east-1:939850196882:secret:*"
    },
    {
      "Effect": "Allow",
      "Action": "secretsmanager:ListSecrets",
      "Resource": "*"
    }
  ]
}
```


Default Password Policy

The default password policy has the following criteria:

- The password should be at least nine characters in length.
- The password should contain at least one number, an uppercase letter (English), a lowercase letter (English), and a special character.
- The password should have at least one letter or number different from the previous password.
- The password should not contain its corresponding username with the same capitalization of letters.
- The password should not contain repeated characters of the same letter or number with the same capitalization of letters.
- The password should not contain the sequential row keyboard input of four letters or numbers with the same capitalization of letters.



©2023 A10 Networks, Inc. All rights reserved. A10 Networks, the A10 Networks logo, ACOS, A10 Thunder, Thunder TPS, A10 Harmony, SSLi and SSL Insight are trademarks or registered trademarks of A10 Networks, Inc. in the United States and other countries. All other trademarks are property of their respective owners. A10 Networks assumes no responsibility for any inaccuracies in this document. A10 Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice. For the full list of trademarks, visit: www.a10networks.com/company/legal/trademarks/.