Installation Manual

AudioCodes Mediant™ Family of Session Border Controllers (SBC)

Mediant Cloud Edition (CE)

Session Border Controller

Version 7.2

audiocodes

Table of Contents

List of Figures

This page is intentionally left blank.

7

Notice

Information contained in this document is believed to be accurate and reliable at the time of printing. However, due to ongoing product improvements and revisions, AudioCodes cannot guarantee accuracy of printed material after the Date Published nor can it accept responsibility for errors or omissions. Updates to this document can be downloaded from https://www.audiocodes.com/library/technical-documents.

This document is subject to change without notice.

Date Published: January-27-2022

WEEE EU Directive

Pursuant to the WEEE EU Directive, electronic and electrical waste must not be disposed of with unsorted waste. Please contact your local recycling authority for disposal of this product.

Customer Support

Customer technical support and services are provided by AudioCodes or by an authorized AudioCodes Service Partner. For more information on how to buy technical support for AudioCodes products and for contact information, please visit our website at https://www.audiocodes.com/services-support/maintenance-and-support.

Stay in the Loop with AudioCodes

Abbreviations and Terminology

Each abbreviation, unless widely used, is spelled out in full when first used.

Related Documentation

Manual Name
Mediant Software SBC User's Manual
SBC Release Notes

Document Revision Record

LTRT	Description
10840	Initial document release for Version 7.2.
10841	Microsoft Azure added.
10842	Installation prerequisites for OpenStack

LTRT	Description
10843	Deployment in non-cloud virtual environments; new deployment via manual installation; network prerequisite diagrams updated
10844	Note removed regarding evaluation for Azure environments.
10845	Azure HA; Activation License (screenshot)
10846	Google Cloud platform and miscellaneous updates.
10847	New section for subscribing Mediant VE in AWS / Azure Marketplace; diagrams updated for network architecture (AWS, Azure / Google Cloud / OpenStack); OAM Subnet changed to Main Subnet; deployment topology updated (AWS / Azure / Google); new sections for private IP addresses (AWS / Azure); section removed – 'Multiple IP Addresses for SC Instances'; new section for installation prerequisites for non-cloud environments; new section for configuring non-standard ports on Azure
10848	Cluster subnet configuration update
10849	IAM Role for Mediant CE (update); Instance Types (update); Public IP Addresses (new); Virtual Machine Sizes (Azure); Machine Types (Google); Default Security Rules (update); Adjusting Security Groups (new), etc.
10855	Section 6.2 updated (VM types, redundancy options and protection)
10858	CentOS 8 for Azure; Google Cloud update; deployment topology for Azure updated; Management Traffic section added for Azure installation; upgrade procedure updated
10862	Typos
10890	Upgrade note for signed .cmp

Documentation Feedback

AudioCodes continually strives to produce high quality documentation. If you have any comments (suggestions or errors) regarding this document, please fill out the Documentation Feedback form on our website at https://online.audiocodes.com/documentation-feedback.

1 Introduction

This document describes initial installation of AudioCodes' **Mediant Cloud Edition** (CE) Session Border Controller (SBC), hereafter referred to as *Mediant CE*.

Mediant CE is a software-based product that is installed and hosted in a cloud computing environment (see note below).

Mediant CE is composed of two component types:

- **Signaling Component (SC):** The SC handles all SIP signaling traffic. It also determines which Media Component (see below) handles the specific media traffic, which is based on load balancing between the Media Components.
- **Media Components (MC):** The MCs handle all media traffic, including transcoding functionality. Up to 21 MCs can be used in the deployed Mediant CE.

Mediant CE provides a unified configuration and management interface, implemented by the SC. This interface provides complete control over all Mediant CE components – both SC and MCs.

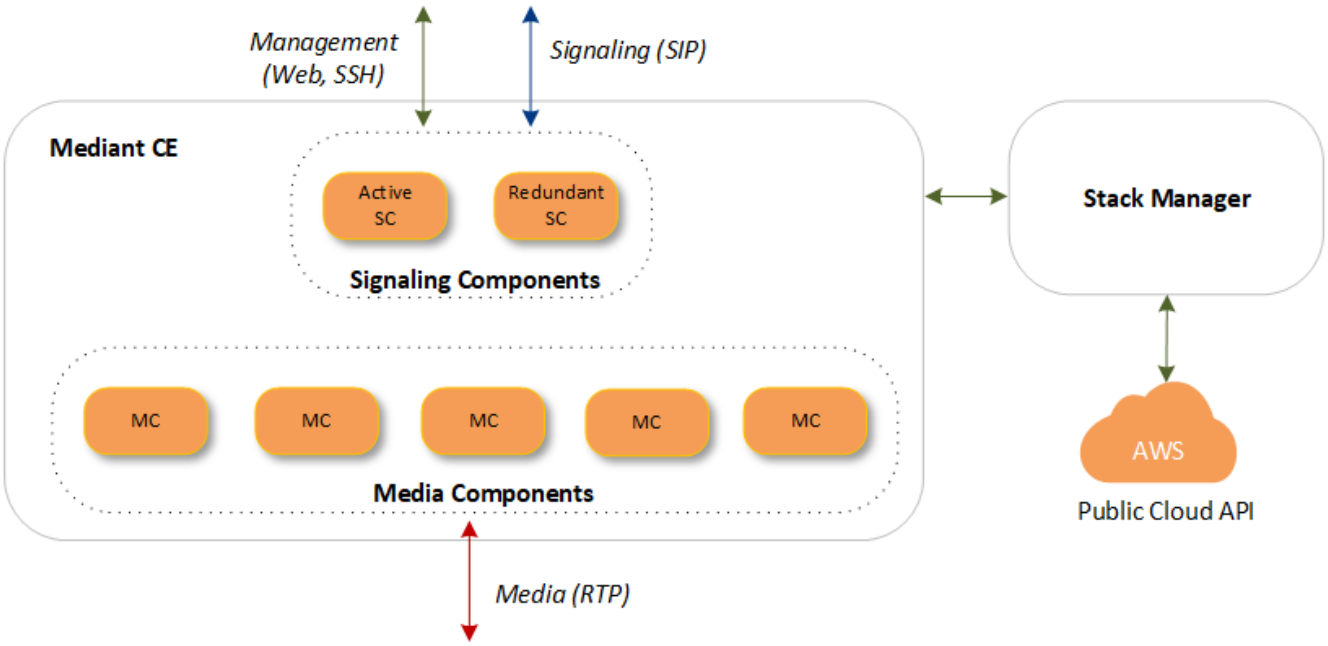
Mediant CE supports High Availability (HA), which is implemented by:

- Employing two SC instances that operate in a 1+1 Active/Standby mode and that provide high availability for management and signaling traffic.
- Employing multiple MC instances that operate in an N+1 Active/Active mode and that provide capacity preservation for media traffic.

The Stack Manager tool is provided as part of the solution. It implements complete lifecycle management of the Mediant CE stack, including initial deployment, manual and automatic scaling, healing and service teardown.

The following figure provides an overview of the Mediant CE architecture.

Figure 1-1: Mediant CE Architecture



Mediant CE currently supports the following cloud computing platforms:

- Amazon Web Services (AWS)
- Microsoft Azure
- OpenStack
- Google Cloud

You may also deploy Mediant CE in non-cloud virtual environments (e.g., VMware), via manual installation and configuration instructions, provided below. Such deployments don't support the Stack Manager component and certain cluster management features. For example, they don't support automatic scaling.

Note:

- Mediant CE deployment in **OpenStack** and **non-cloud virtual environments** is currently available for **evaluation** purposes only.
- Mediant VE and CE products share the same software image. In the AWS Marketplace, AudioCodes has published the image for these products on AWS Marketplace under the name "Mediant VE Session Border Controller (SBC)". Therefore, in some places in this document, this product name is referenced even though the document concerns Mediant CE.
- The scope of this document does not fully cover security aspects for deploying the product in the cloud. Security measures should be done in accordance with specific cloud security policies and recommendations.
- For configuring Mediant CE, refer to the *Mediant Software SBC User's Manual*.

2 Installation Prerequisites for Amazon Web Services (AWS) Environment

Prior to installing Mediant CE in the Amazon Web Services (AWS) environment, make sure that you meet the following prerequisites:

- You have an AWS account. If you don't have an AWS account, you can sign up for one on Amazon's website at http://aws.amazon.com/.
- You have subscribed to the AudioCodes Mediant VE offer in AWS Marketplace. For more information, see Section Subscribing to AudioCodes Mediant VE Product in AWS Marketplace on page 11.
- You have created an Identity and Access Management (IAM) role that enables Mediant CE to manage its network interfaces. For more information, see Section IAM Role for Mediant CE on page 12.
- You have created all subnets needed for Mediant CE deployment, including the Cluster subnet with a private EC2 endpoint or NAT gateway. For more information, see Section Network Prerequisites on page 14.

2.1 Subscribing to AudioCodes Mediant VE Product in AWS Marketplace

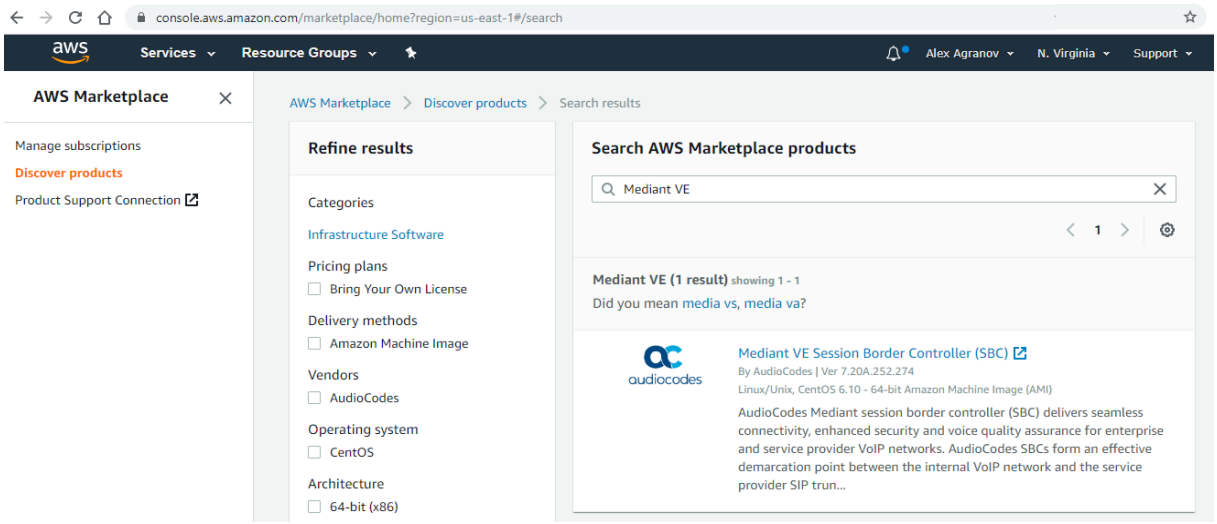
Mediant VE and CE products share the same software image. AudioCodes distributes Mediant VE/CE software images by publishing them in the AWS Marketplace.

Note: As Mediant VE and CE products share the same software image, AudioCodes has published the image for these products on AWS Marketplace under the name "Mediant VE Session Border Controller (SBC)".

Prior to deploying the Mediant CE you must subscribe to the AudioCodes Mediant VE product in AWS Marketplace as follows:

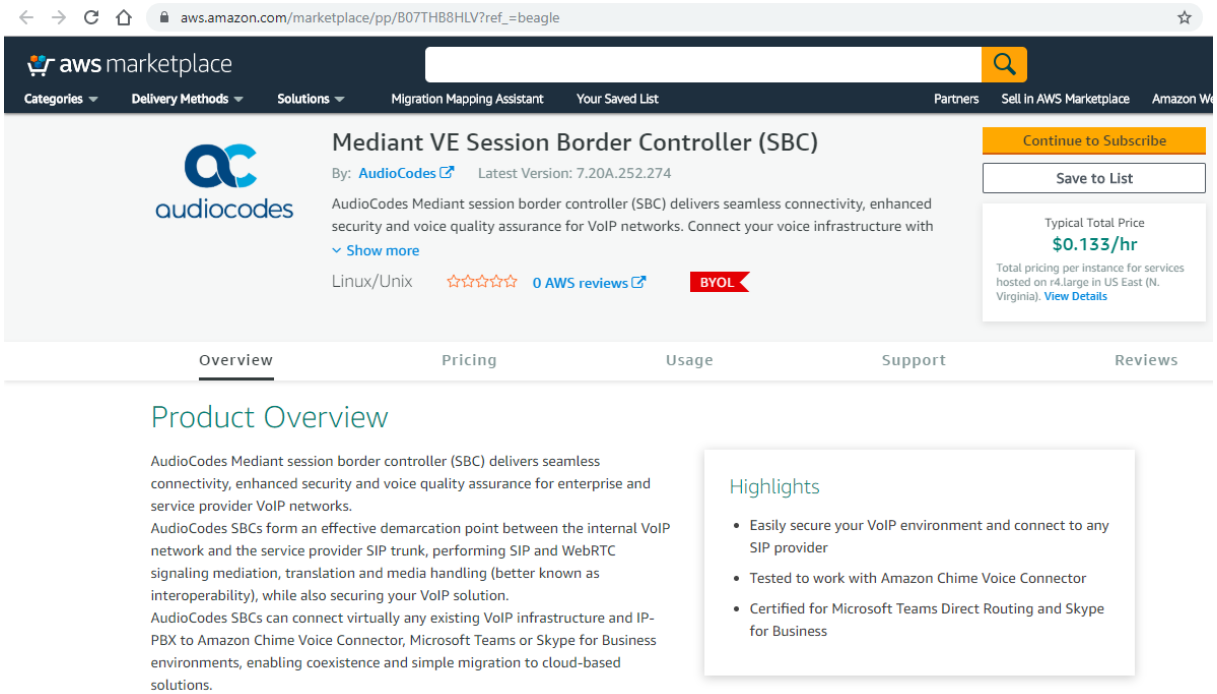
1. Open the AWS Marketplace console at https://console.aws.amazon.com/marketplace.
2. In the **Discover Products** tab, search for the "Mediant VE" product.

Figure 2-1: Searching for Mediant VE Product in the AWS Marketplace



3. Click the **Mediant VE Session Border Controller (SBC)** product.

Figure 2-2: Mediant VE Product in AWS Marketplace



4. Click **Continue to Subscribe** to subscribe to the Mediant VE product.

2.2 IAM Role for Mediant CE

The following IAM role must be created prior to creating the Mediant CE stack. This role ensures that Mediant CE components can manage their network interfaces and re-assign IP addresses in case of a switchover.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "ec2:AssignPrivateIpAddresses",
                "ec2:UnassignPrivateIpAddresses",
                "ec2:AssociateAddress",
                "ec2:DescribeAddresses",
                "ec2:DescribeNetworkInterfaceAttribute",
                "ec2:DescribeNetworkInterfaces"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

➢ **To create an IAM Role:**

1. Open the AWS IAM management console at https://console.aws.amazon.com/iam.
2. Navigate to the **Policies** screen:
   a. Click **Create**.
   b. Select the **JSON** tab, copy-and-paste the IAM policy rules listed above, and then click **Review policy**.
   c. Enter the IAM policy name (e.g. "SBC_HA"), and then click **Create policy**.
3. Navigate to the **Roles** screen:
   a. Click **Create role**.
   b. Choose **EC2** use case, and then click **Next: permissions**.
   c. Search for the IAM policy created in the previous step, select it, and then click **Next: tags**.
   d. Click **Next: review**.
   e. Enter the IAM role name (e.g. "SBC_HA"), and then click **Create role**.

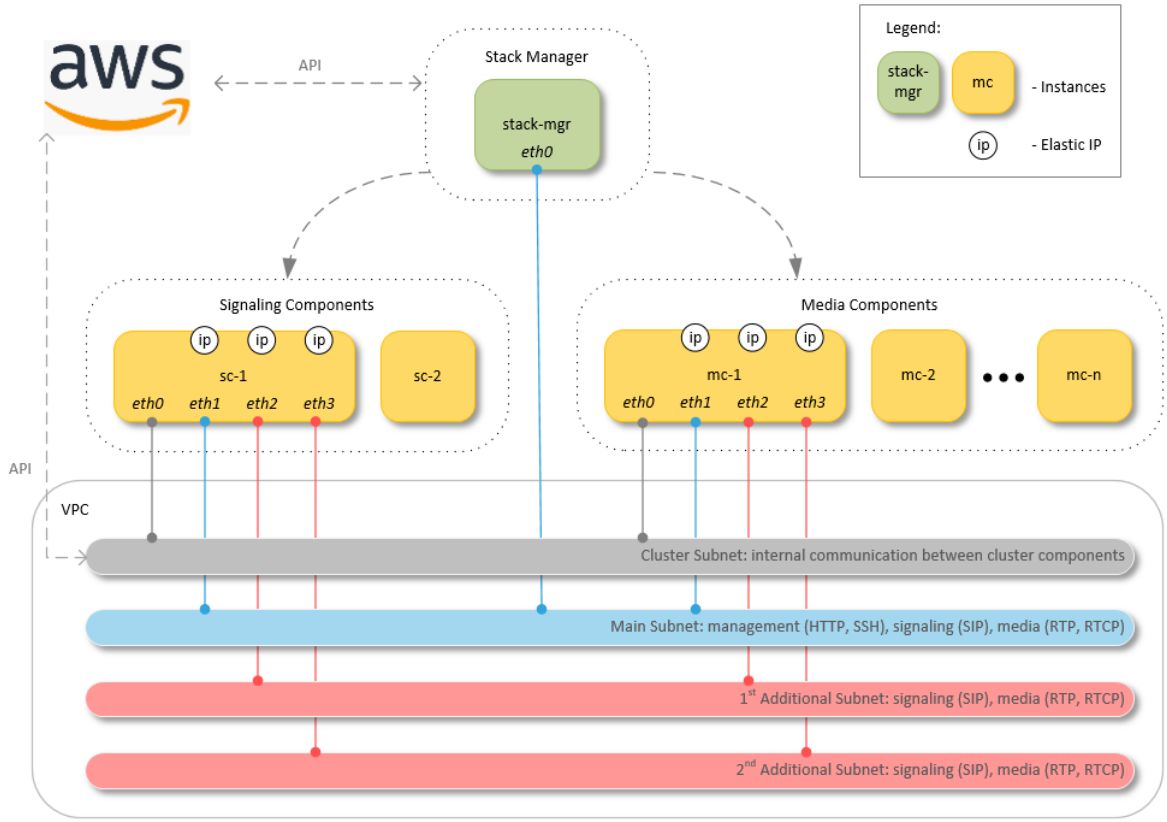
If you want to perform automatic Mediant CE provisioning using a configuration file stored on the AWS S3 service, add the corresponding statements to the IAM role, for example:

```
        {
            "Effect": "Allow",
            "Action": [
                "s3:ListBucket"
            ],
            "Resource": "arn:aws:s3:::sbc"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
            ],
            "Resource": "arn:aws:s3:::sbc/*"
        }
```

2.3 Network Prerequisites

Mediant CE on Amazon Web Services (AWS) uses the following network architecture:

Figure 2-3: Mediant CE Network Architecture – AWS



Up to four subnet may be used:

- **Cluster Subnet:** For internal communication between Mediant CE components and for accessing the AWS API; connected to both SC and MC instances as the first network interface (eth0); it must have a private EC2 endpoint or NAT gateway attached (for more information, see Section Cluster Subnet on page 15)
- **Main Subnet:** Carries management (HTTP, SSH, etc.), signaling (SIP) and media (RTP, RTCP) traffic; connected to both SC and MC instances as the second network interface (eth1) and to the Stack Manager instance
- **1st and 2nd Additional Subnets:** Carry signaling (SIP) and media (RTP, RTCP) traffic; connected to MC instances as the third and fourth network interfaces (eth2 and eth3) correspondingly; theses subnets are optional, as the Main Subnet may carry all types of traffic.

All subnets must reside in the same Availability Zone of the Virtual Private Cloud (VPC).

All needed subnets must be created prior to the Mediant CE deployment.

During the deployment, Stack Manager creates all relevant Mediant CE components, including SC and MC instances and public IP addresses.

SCs operate in 1+1 Active/Standby mode and use "floating" IP addresses, reassigned via AWS API during activity switchover. Since AWS does not support reassignment of primary IP addresses, SCs never use them, but use secondary IP addresses instead (except for the Cluster subnet).

2.3.1 Cluster Subnet

The Cluster Subnet is used for the following tasks:

- Internal communication between Mediant CE components
- Accessing AWS API (for IP address management)

Mediant CE uses private addresses in the Cluster Subnet. Therefore, to enable Mediant CE to access AWS API via the Cluster subnet, you must do one of the following:

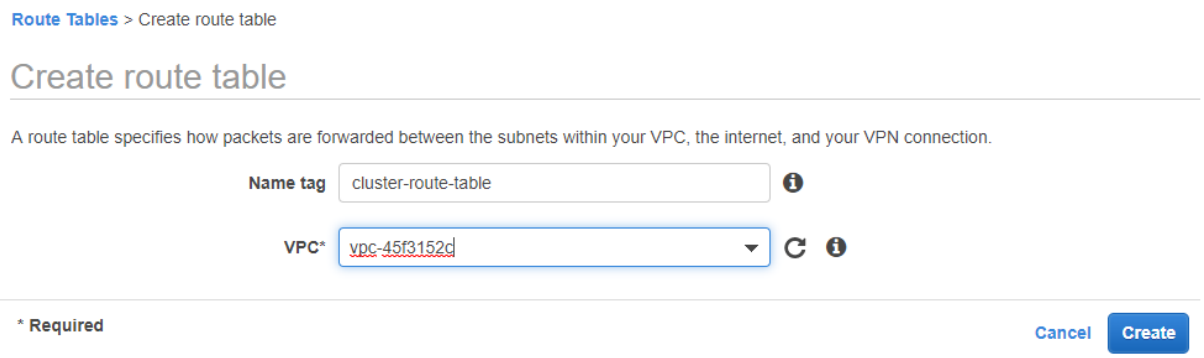
- Create a private EC2 endpoint in the Cluster subnet (recommended method)
- Attach a NAT gateway to the Cluster subnet (alternative method)

In addition, since the Cluster subnet carries sensitive information, it is recommended to create a dedicated subnet and protect it from unauthorized access.

➢ **To create the Cluster subnet:**

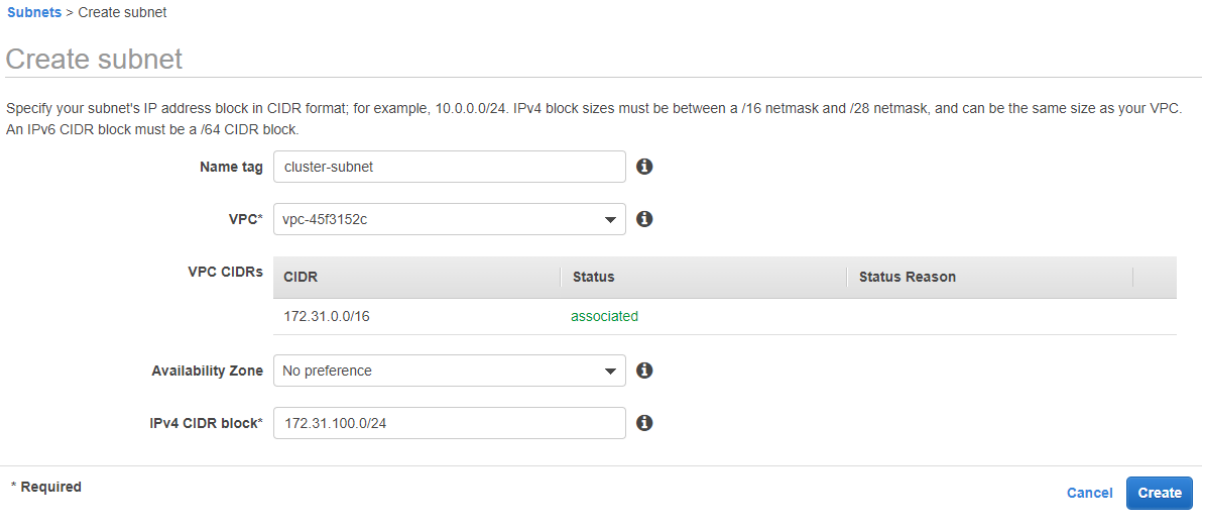
1. Open the AWS VPC management console at https://console.aws.amazon.com/vpc.
2. Open the Route Tables page, and then click **Create route table:**
   a. In the 'Name tag' field, enter the new route table name (e.g. 'cluster-route-table').
   b. From the 'VPC' drop-down list, select the VPC where Mediant CE will be deployed.
   c. Click **Create** to create the route table.

Figure 2-4: Creating Route Table



3. Open the Subnets page, and then click **Create Subnet**.
   a. In the 'Name tag' field, enter the new subnet name (e.g. 'cluster-subnet').
   b. From the 'Availability Zone' drop-down list, select the Availability Zone where Mediant CE will be deployed.
   c. In the 'IPv4 CIDR block' field, enter the IPv4 CIDR for the subnet.
   d. Click **Yes, Create** to create the route table.

Figure 2-5: Creating Cluster Subnet

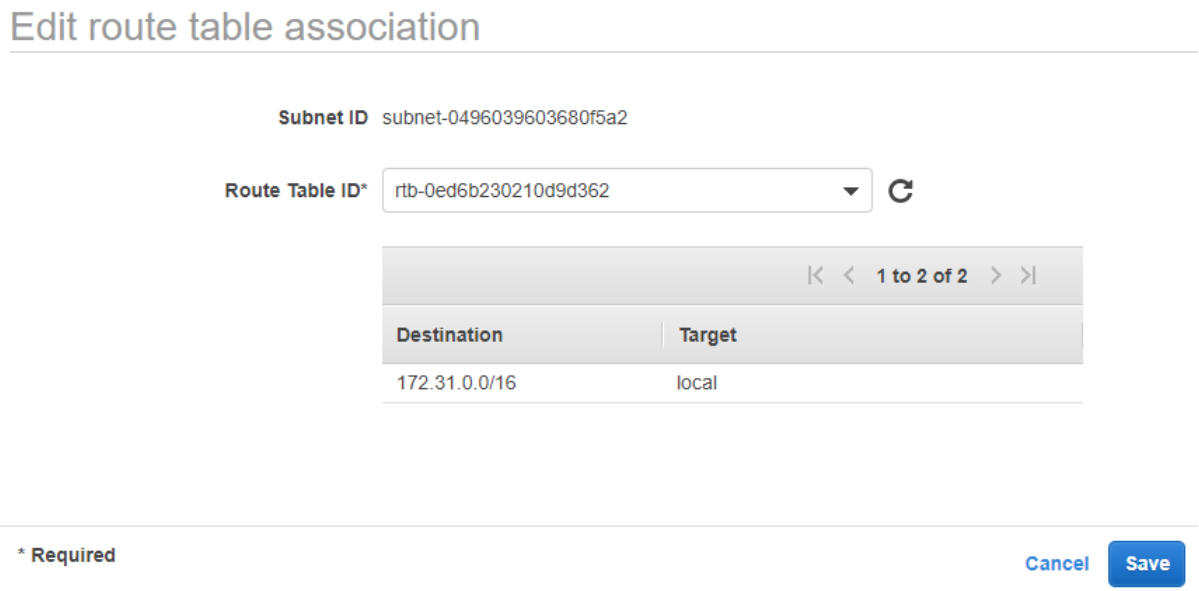


4. Select the created subnet, switch to the **Route Table** tab, and then click **Edit route table association**.

Figure 2-6: Changing Cluster Subnet Route Table

5. Choose the Cluster route table created in the previous steps, and then click **Save**.

Figure 2-7: Editing Route Table Association



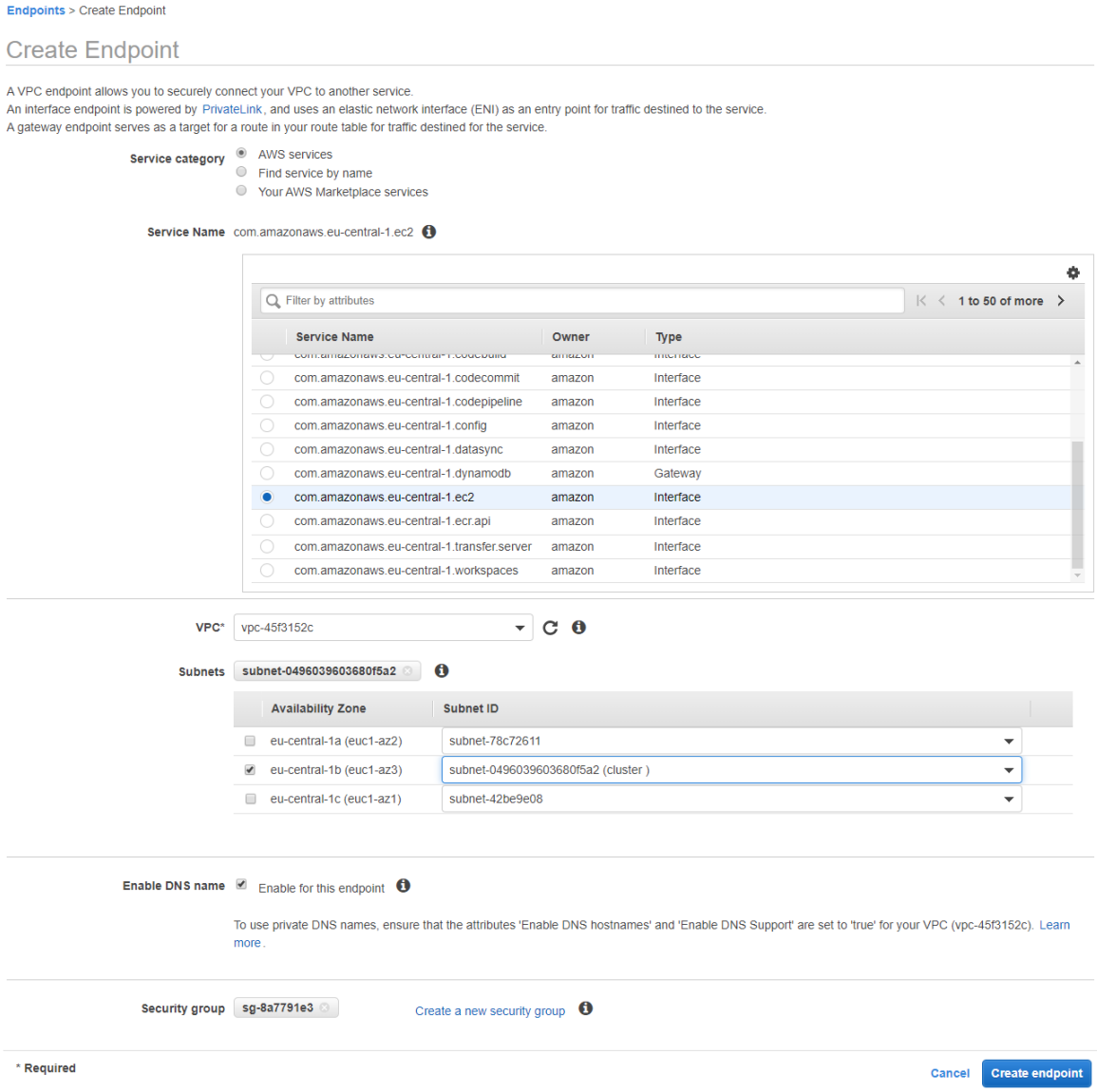
Note: Make sure that Cluster subnet has a dedicated route table. Other subnets (Main subnet, Additional subnets) should be attached to different route table(s), which would typically have the Internet Gateway configured as the default route to ensure proper functionality of Elastic IPs attached to the corresponding network interfaces of EC2 instances.

After you have successfully created the Cluster subnet, you need to enable access to the AWS API via through this subnet. The recommended method is to create a private EC2 endpoint in the Cluster subnet.

➢ **To create the private EC2 endpoint in Cluster subnet:**

1. Open the Endpoints page, and then click **Create Endpoint**.
2. In the 'Service Category' field, select **AWS services**.
3. In the 'Service Name' field, select **com.amazonaws.eu-central-1.ec2**.
4. From the 'VPC' drop-down list, select the VPC where Mediant CE will be deployed.
5. In the 'Subnets' field, select the Cluster subnet.
6. Select the **Enable DNS name** checkbox.
7. From the 'Security group' drop-down list, select the security group that will allow the private endpoint to communicate with the public AWS APIs.
8. Click **Create Endpoint** to create the new endpoint.

Figure 2-8: Creating Private EC2 Endpoint



An alternative method for enabling access to the AWS API through the Cluster subnet is by attaching a NAT Gateway to the Cluster subnet.

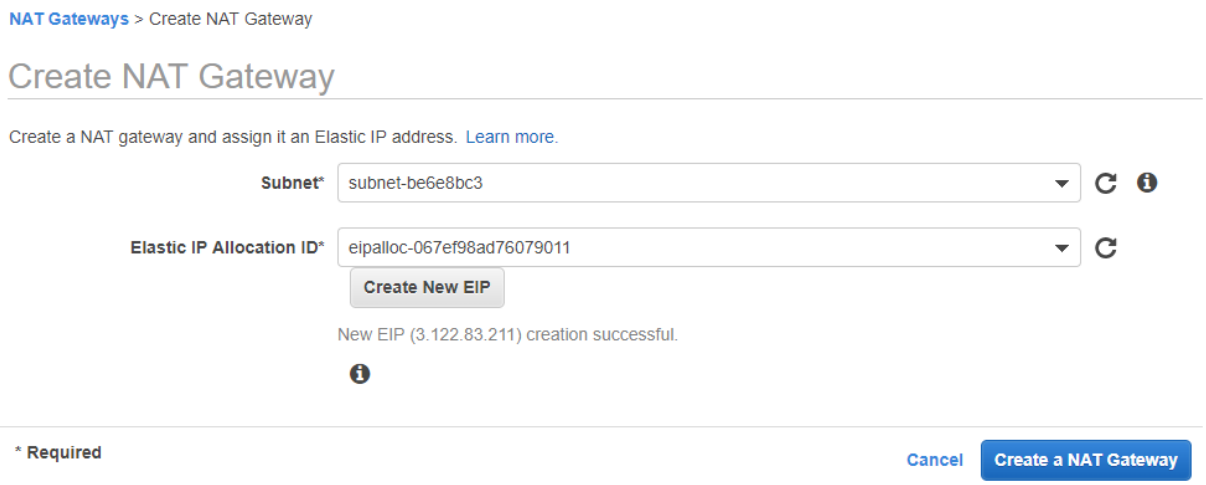
➢ **To create NAT Gateway and attach it to the Cluster subnet:**

1. Open the NAT Gateways page, and then click **Create NAT Gateway**:
   a. From the 'Subnet' drop-down list, select a subnet that belongs to the same Availability Zone where the Cluster subnet was created (and where Mediant CE will be deployed) and that has an Internet Gateway attached to it. For example, select **Main Subnet**.

Note: Do not select **Cluster Subnet** at this stage. The NAT Gateway itself will be configured as a default route in the Cluster Subnet and therefore, it won't be able to access the Internet from it.

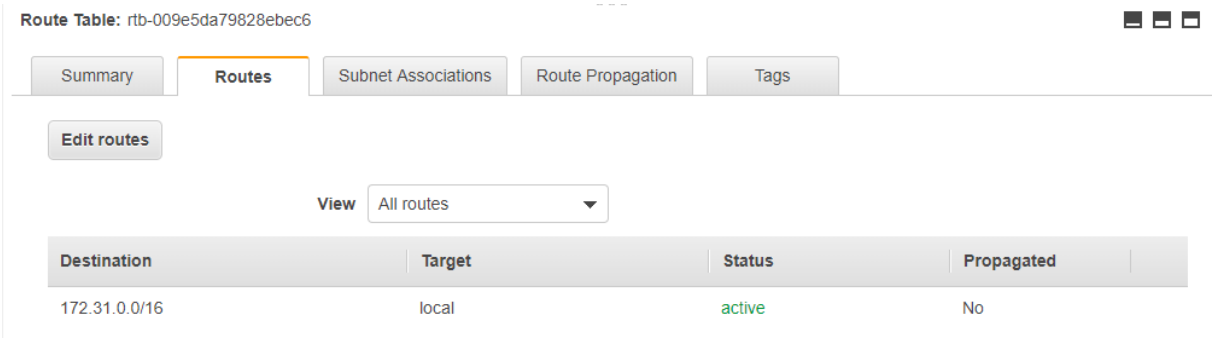
   b. From the 'Elastic IP Allocation ID' drop-down list, select an existing Elastic IP if you have pre-allocated Elastic IPs in your VPC, or click **Create New EIP** to create a new one.
   c. Click **Create a NAT Gateway** to create the NAT gateway.

Figure 2-9: Creating NAT Gateway



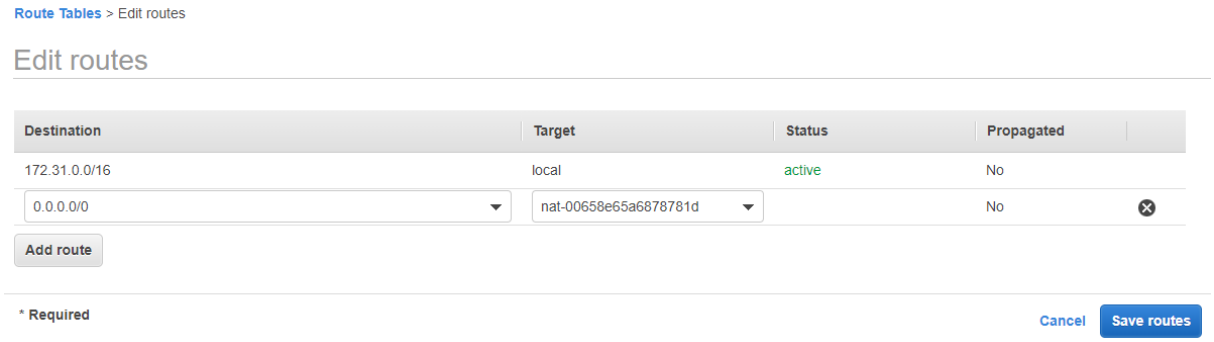
2. Open the Route Tables page, and then select the Cluster route table created in the previous steps.
3. Switch to the **Routes** tab, and then click **Edit routes** to edit the routes.

Figure 2-10: Editing Route Table



4. Create the default route entry (0.0.0.0/0) that points to the created NAT gateway, and then click **Save** to save your changes.

Figure 2-11: Creating Default Route



2.4 Instance Types

Default Mediant CE deployment uses the following instance types:

- **SC instances:** r4.2xlarge
- **Forwarding MC instances:** r4.large or r4.xlarge (depending on number of network interfaces)
- **Transcoding MC instances:** c4.4xlarge

You may customize instance types by specifying **sc_instance_type** and/or **mc_instance_type** advanced configuration parameters (via **Advanced Config** section) during stack creation.

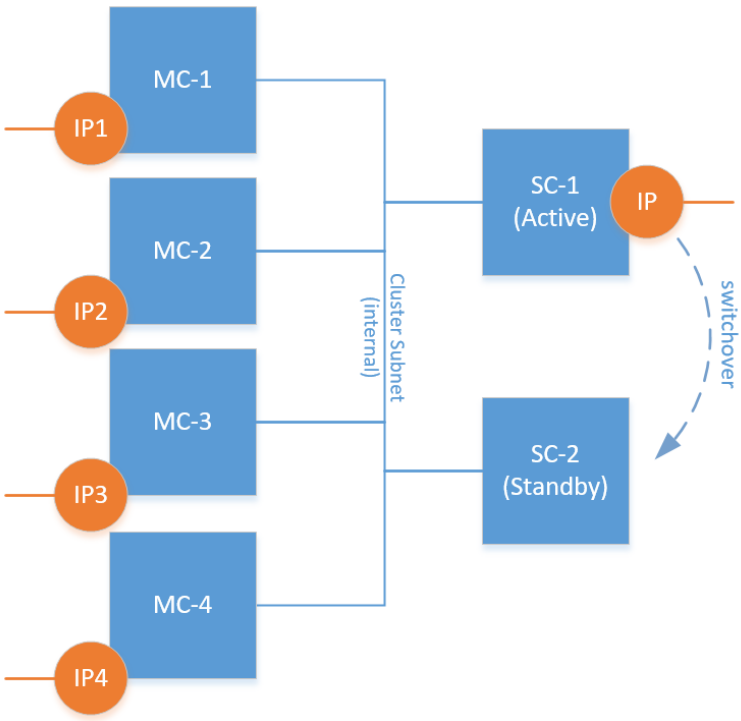
For example:

```
sc_instance_type = r4.xlarge
mc_instance_type = c4.2xlarge
```

2.5 Deployment Topology

All Mediant CE components are deployed in a single Availability Zone of an AWS Region.

Figure 2-12: Mediant CE Deployment Topology (AWS)



Communication with signaling and media components may be performed via either public or private IP addresses.

IP addresses of active SC instance are moved (using AWS APIs) to the standby SC instance in case of SC switchover.

2.6 Public IP Addresses

During Mediant CE stack creation, Stack Manager lets you specify which subnets (and corresponding network interfaces) will be assigned with public (Elastic) IP addresses via the **Public IPs** parameter in the **Networking** section.

For each assigned Elastic IP address, Stack Manager creates corresponding entries in the NAT Translation SBC configuration table, thus ensuring that when SIP application attached to the corresponding private IP addresses communicates with external SIP peers, it essentially does this via the Elastic IP address.

It is also possible to attach multiple Elastic IP addresses to the same network interface. This may be done by configuring the **sc_public_ips** / **mc_public_ips** advanced configuration parameter (via **Advanced Config** section).

Note: When the **sc_public_ips / mc_public_ips** advanced configuration parameter is specified (via **Advanced Config** section), it overrides any value configured via the **Public IPs** parameter in the **Networking** section.

- **sc_public_ips**

  Contains a comma-separated list of subnet names (main, additional1, and additional2), which will be assigned with Elastic IP addresses, and optionally, with the number of Elastic IP addresses on the corresponding network interface.

  For example:

  ```
  sc_public_ips = main:2,additional1
  ```

  attaches two Elastic IP addresses to the network interface connected to the Main subnet (eth1) and one Elastic IP address to the network interface connected to the Additional 1 subnet (eth2).

- **mc_public_ips**

  Same as above, but for MC network interfaces.

  For example:

  ```
  mc_public_ips = main,additional1:2
  ```

When the **sc_public_ips** / **mc_public_ips** advanced configuration parameter is specified, Stack Manager automatically creates secondary private IP addresses on the network interfaces that may be required for Elastic IP attachment. The exact behavior depends on the component type:

- For MCs: First Elastic IP address is attached to the primary private IP address; for each additional Elastic IP address, corresponding secondary IP addresses are implicitly created.
- For SCs: Elastic IP addresses are always attached to the secondary private IP addresses; for each Elastic IP address, corresponding secondary IP addresses are implicitly created.

2.7 Private IP Addresses

Stack Manager always creates one "operational" private IP address on each network interface. The exact behavior depends on the component type:

- **For MCs:** Primary IP address is used on each interface
- **For SCs:** Primary IP addresses on eth1, eth2 and eth3 interfaces (connected to Main, 1st and 2nd Additional subnets correspondingly) are not used, because they can't be moved between two SC instances during activity switchover; instead, secondary IP addresses are created and used

When an Elastic IP address is assigned to the specific subnet, a corresponding "operational" private IP address may not be used for SIP traffic, because of the NAT Translation table entry that implements SNAT translation at the application level (SIP and SDP).

If you wish to enable communication via both Elastic and private IP addresses on the same subnet, you need to create additional "operational" private IP addresses on the same network interface. This may be done by configuring the **sc_additional_ips** / **mc_additional_ips** advanced configuration parameters (via **Advanced Config** section).

- **sc_additional_ips**

  Contains a comma-separated list of subnet names (main, additional1, and additional2), which will be assigned with additional private IP addresses, and optionally, with the number of additional private IP addresses on the corresponding network interface.

  For example:

  ```
  sc_additional_ips = main,additional1:2
  ```

  attaches one additional private IP address to the network interface connected to the Main subnet (eth1) and two additional private IP addresses to the network interface connected to the Additional 1 subnet (eth2).

- **mc_additional_ips**

  Same as above, but for MC network interfaces.

  For example:

  ```
  mc_additional_ips = main,additional1:2
  ```

The number of additional private IP addresses specified via the **sc_additional_ips** / **mc_additional_ips** advanced configuration parameter is added *on top* of any private IP addresses created by Stack Manager by default and/or due to the public (Elastic) IP addresses assigned to the specific network interface.

For example, the following configuration:

```
Cluster Subnet: <cluster-subnet-id>
Main Subnet: <main-subnet-id>
1st Additional Subnet: <additional-subnet-id>
Public IPs: "Main subnet"
Advanced Config:
    sc_additional_ips = main,additional1
```

creates the following networking configuration on signaling components:

- **eth0** – one primary IP addresses (used for internal communication between SC instances) and one secondary IP address (used for internal communication with MC instances)
- **eth1** – one primary and two secondary IP addresses:
  - primary IP address is not used because it can't be moved between SC instances in case of switchover

- 1st secondary IP address – first "operational" private IP address, created implicitly and assigned with Elastic IP address (due to the **Public IPs** configuration parameter)
- 2nd secondary IP address – created due to the **sc_additional_ips** advanced configuration parameter

- **eth2** – one primary and two secondary IP addresses:
  - primary IP address is not used because it can't be moved between SC instances in case of switchover
  - 1st secondary IP address – first "operational" private IP address, created implicitly
  - 2nd secondary IP address – created due to the **sc_additional_ips** advanced configuration parameter

3 Installation Prerequisites for Microsoft Azure Environment

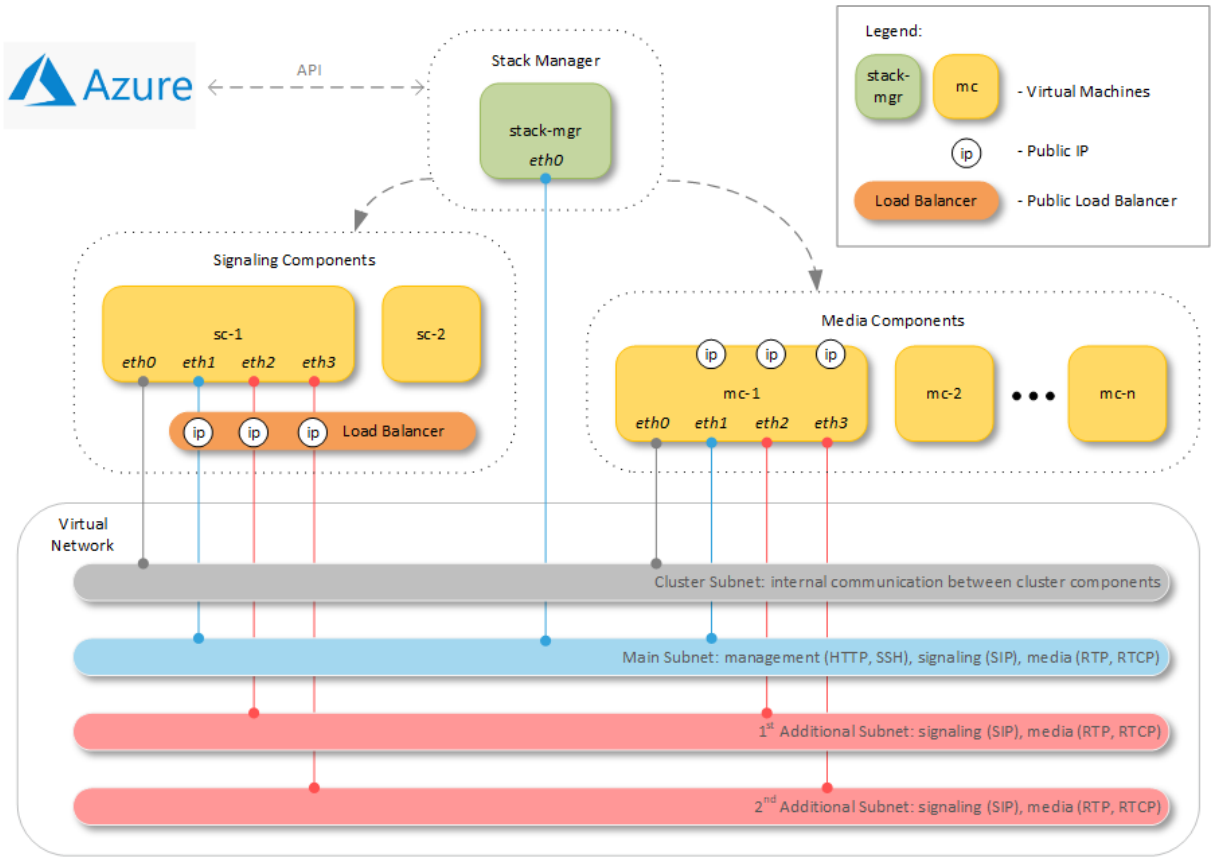
Prior to installing Mediant CE in a Microsoft Azure environment, make sure that you meet the following prerequisites:

- You have a Microsoft Azure account. If you don't have an Azure account, you can sign up for one on Microsoft's website at http://azure.microsoft.com.
- You have subscribed to AudioCodes Mediant VE offer in the Azure Marketplace. For more information, see Subscribing to Mediant VE Offer in Azure Marketplace on page 26.
- You have created all subnets needed for Mediant CE deployment, including the Cluster subnet. For more information, see Section Network Prerequisites on page 25.

3.1 Network Prerequisites

Mediant CE on Microsoft Azure uses the following network architecture:

Figure 3-1: Mediant CE Network Architecture – Azure



Up to four subnet may be used:

- **Cluster Subnet:** For internal communication between Mediant CE components; connected to both SC and MC instances as the first network interface (eth0).
- **Main Subnet:** Carries management (HTTP, SSH, etc.), signaling (SIP) and media (RTP, RTCP) traffic; connected to both SC and MC instances as the second network interface (eth1) and to the Stack Manager instance.

- **1st and 2nd Additional Subnets:** Carries signaling (SIP) and media (RTP, RTCP) traffic; connected to MC instances as the third and fourth network interfaces (eth2 and eth3) correspondingly. These subnets are optional, as the Main Subnet may carry all types of traffic.

All subnets must reside in the same Virtual Network.

All needed subnets must be created prior to the Mediant CE deployment.

During deployment, Stack Manager creates all relevant Mediant CE components, including SC and MC instances, load balancer, and public IP addresses.

3.2 Subscribing to Mediant VE Offer in Azure Marketplace

Mediant VE and CE products share the same software image. AudioCodes distributes Mediant VE/CE software images by publishing them in the Azure Marketplace.

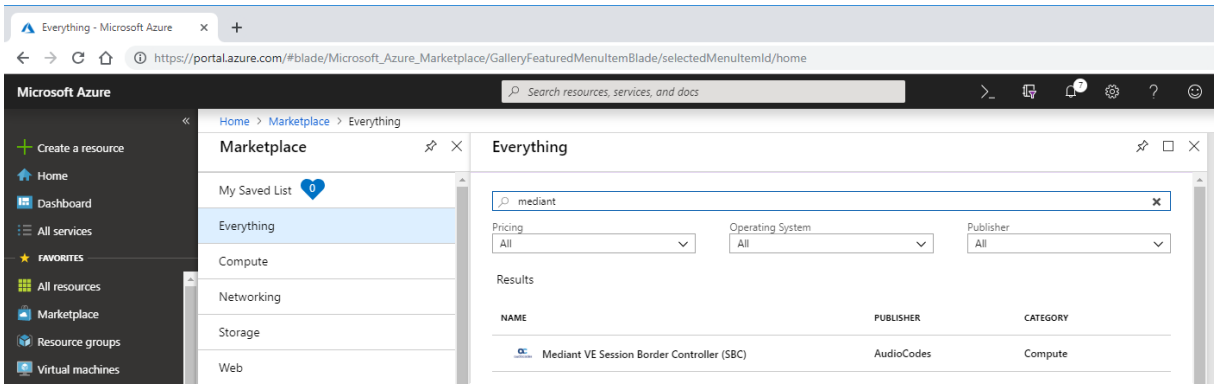
Note: As Mediant VE and CE products share the same software image, AudioCodes has published the image for these products on Azure Marketplace under the name "Mediant VE Session Border Controller (SBC)".

Prior to deploying the Mediant CE you must subscribe to the AudioCodes Mediant VE offer in Azure Marketplace. This is done by deploying a demo instance of Mediant VE product from Azure Marketplace in your subscription. The deployed instance may be deleted immediately after creation.

➢ **To deploy a demo instance of Mediant VE product from Azure Marketplace:**

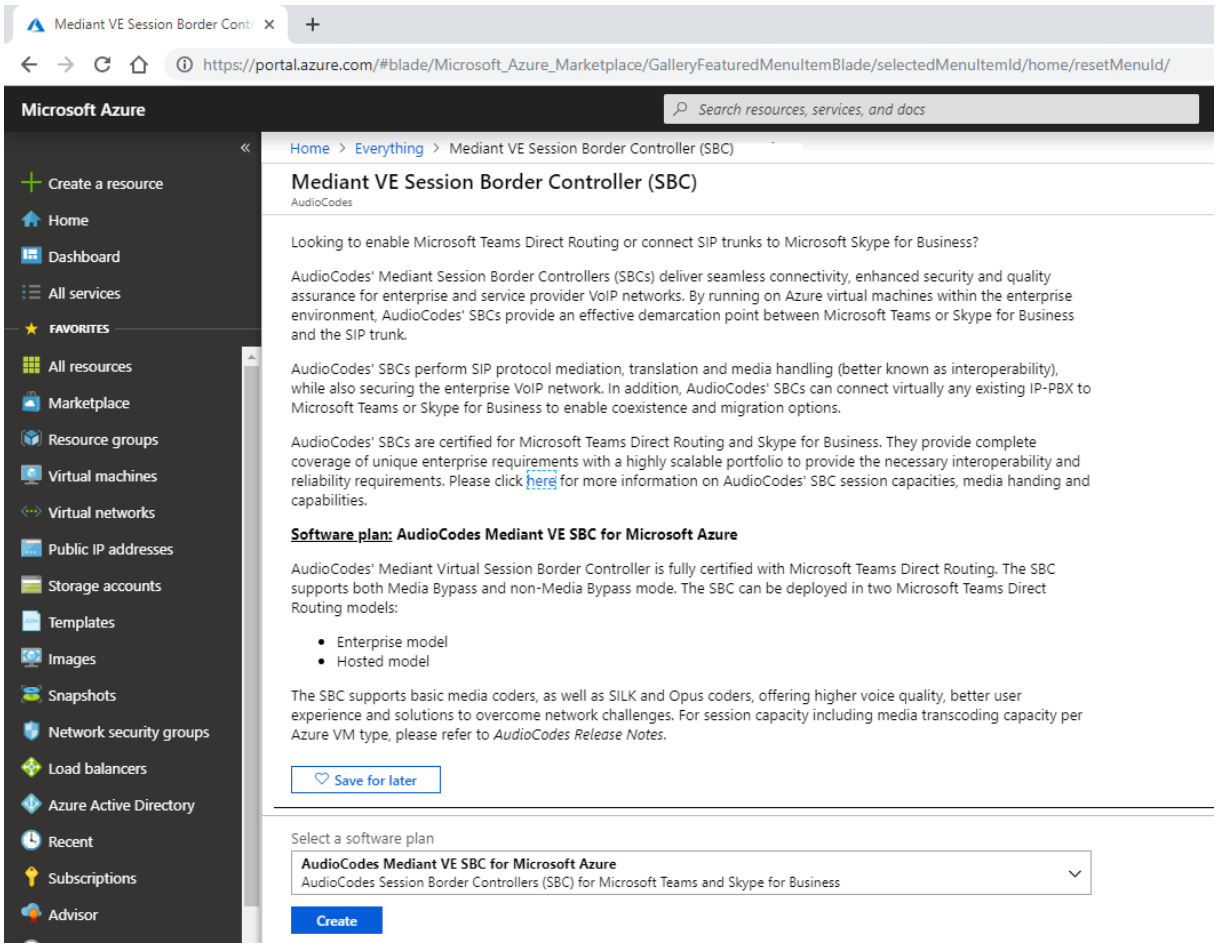
1. Open the Azure portal at https://portal.azure.com/.
2. Navigate to the Azure Marketplace (**All services** > **Marketplace**).
3. Search for the product "Mediant VE Session Border Controller (SBC)" published by AudioCodes.

Figure 3-2: Azure Marketplace



4. Click the **Mediant VE Session Border Controller (SBC)** product; the Mediant VE Product overview screen appears.

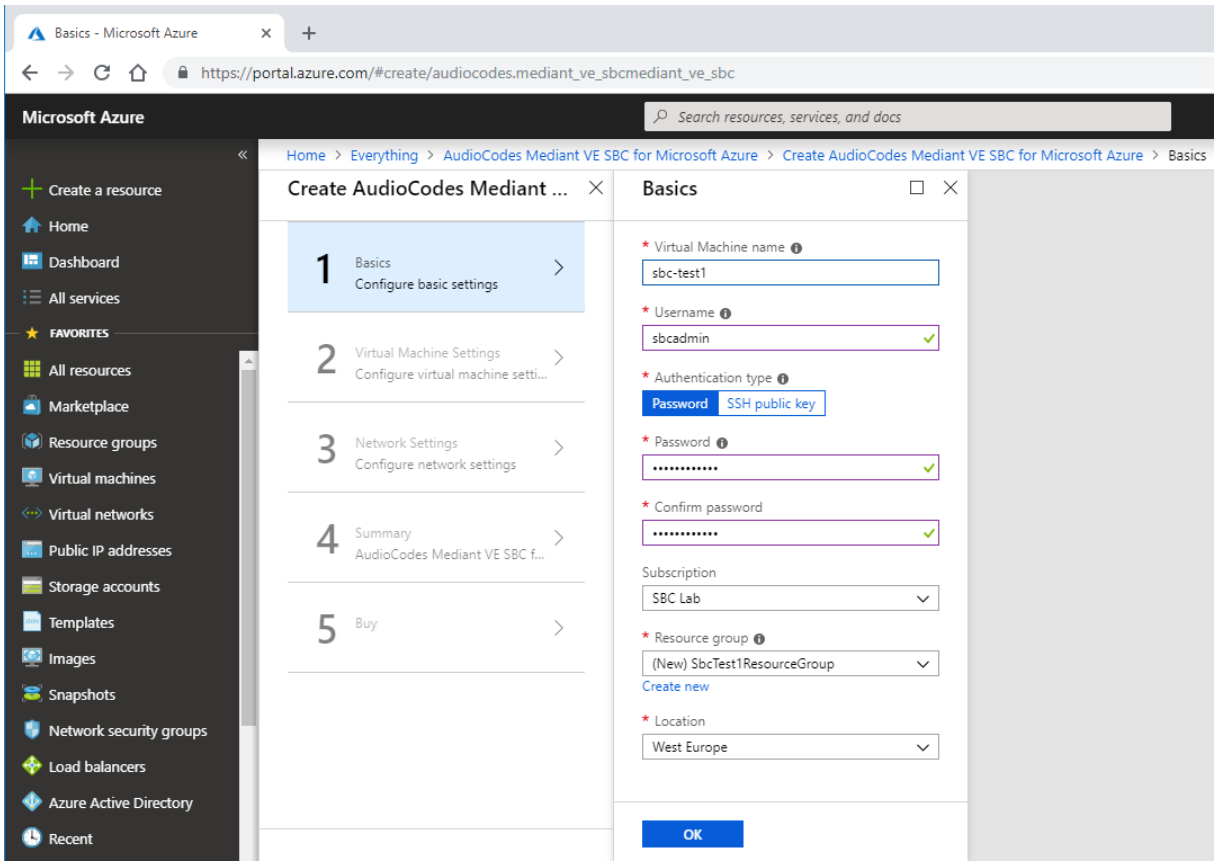
Figure 3-3: Mediant VE SBC Product Overview



5. Click **Create** to start a new Mediant VE deployment; the Create AudioCodes Mediant VE SBC for Microsoft Azure dialog box appears. The dialog box contains multiple steps. Complete each step according to the description below.

6. In the **Basics** step, do the following:

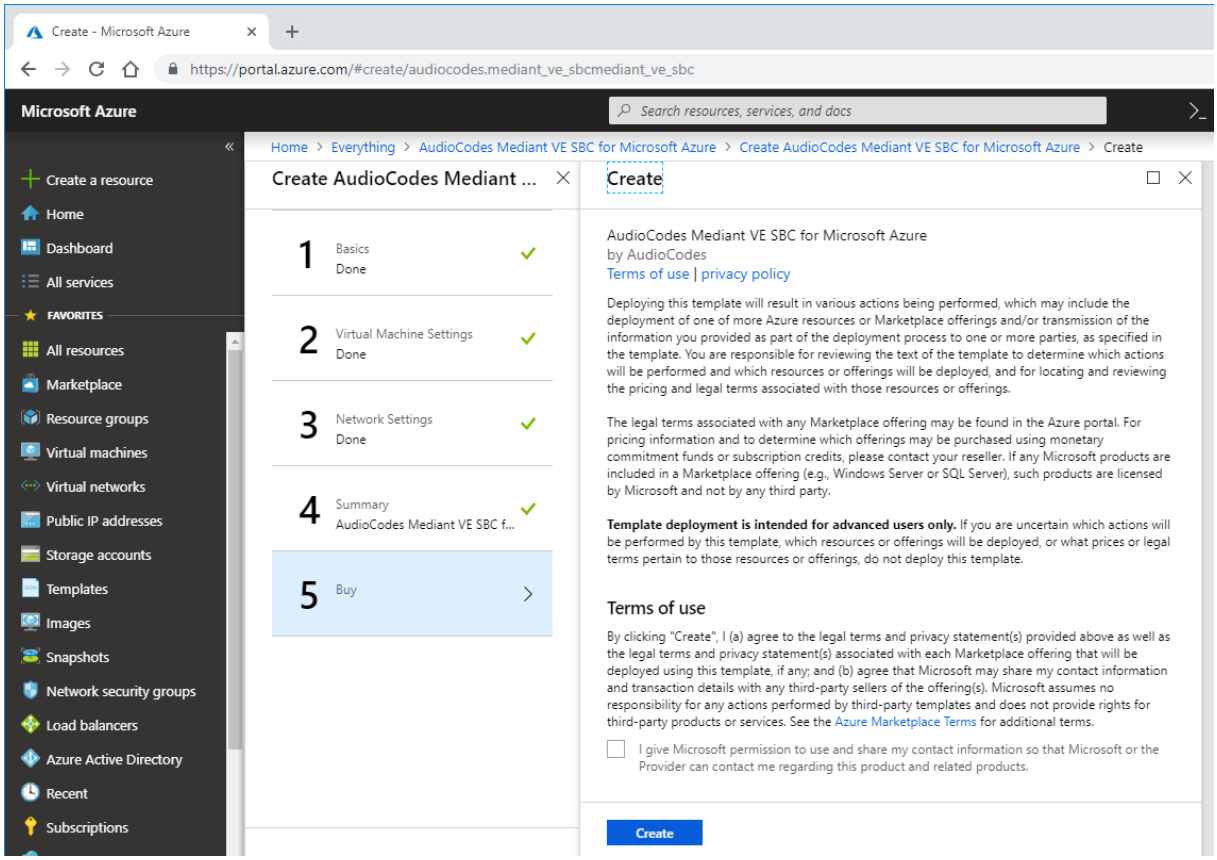
Figure 3-4: Basics Step



   a. In the 'Virtual Machine name' field, enter a unique name for the new VM.
   b. In the 'Username' field, enter a username – e.g. "sbcadmin".
   c. For 'Authentication type', select the **Password** option.
   d. In the 'Password' field, enter a password – e.g. "Admin#123456".
   e. From the 'Subscription' drop-down list, select a proper subscription for your deployment.
   f. Under 'Resource group', select the **Create new** option and then enter a new Resource Group name for your deployment.
   g. From the 'Location' drop-down list, select a proper location for your deployment.
   h. Click **OK**.

7. In the **Virtual Machine Settings** and **Network Settings** steps accept the defaults and click **OK**.
8. In the **Buy** step, review the Mediant VE SBC terms of use, and then click **OK** to start the virtual machine deployment.

Figure 3-5: Buy Step



9. Wait until the virtual machine deployment is complete
10. Delete deployed demo instance by deleting the corresponding Resource Group

➢ **To delete demo instance of Mediant VE product:**

- Delete the corresponding Resource Group specified during virtual machine creation

3.3 Virtual Machine Sizes

The following instance types are used by default Mediant CE deployment:

- **SC instances:** Standard_DS3_v2
- **Forwarding MC instances:** Standard_DS2_v2 or Standard_DS3_v2 (depending on number of network interfaces)
- **Transcoding MC instances:** Standard_DS3_v2

You may customize instance types by specifying the **sc_instance_type** and/or **mc_instance_type** advanced configuration parameters (via **Advanced Config** section) during stack creation.

For example:

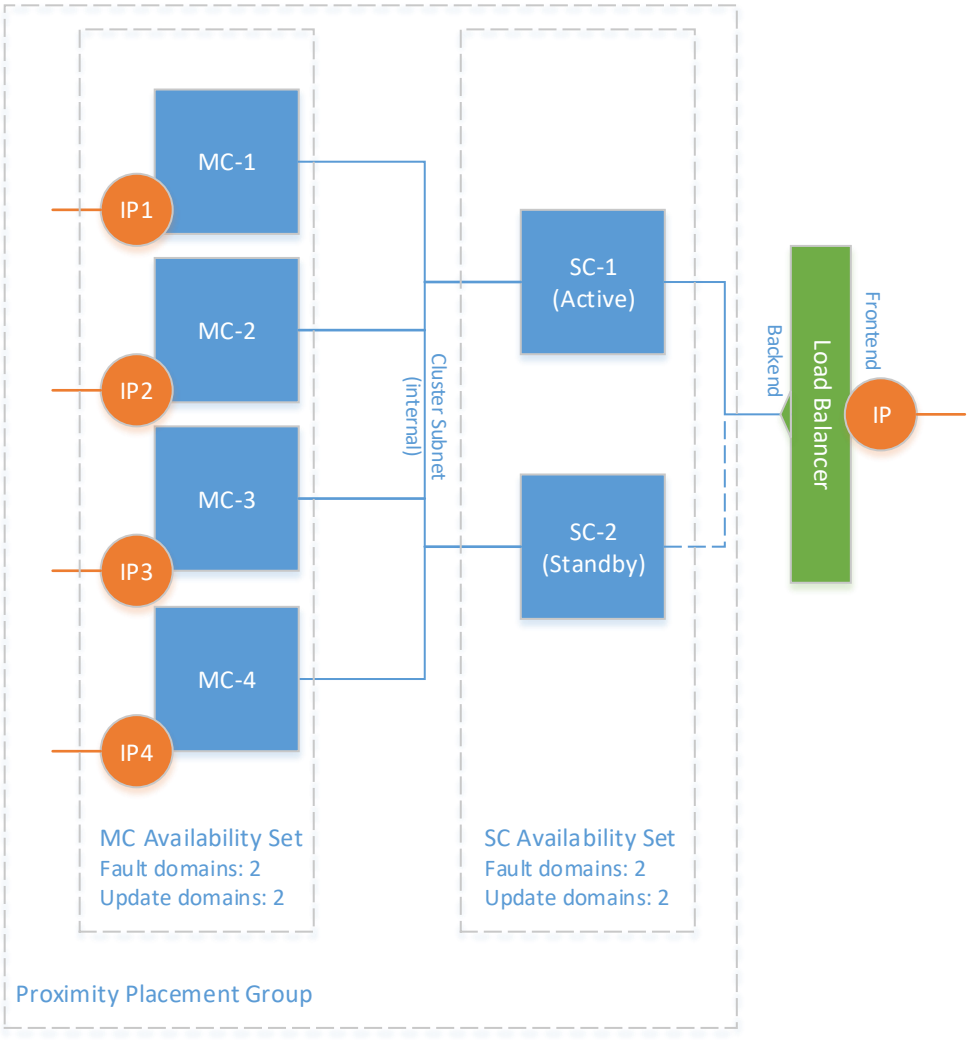
```
sc_instance_type = Standard_DS4_v2
mc_instance_type = Standard_DS4_v2
```

3.4 Deployment Topology

Mediant CE components are deployed into a single Proximity Placement Group with two Availability Sets (each containing two fault and update domains) for Signaling and Media Components, respectively. This deployment topology minimizes network latency between Mediant CE components while still providing adequate redundancy at the infrastructure level.

If you want to deploy Mediant CE into two Availability Zones instead, you may do so by specifying the **availability_zones** advanced configuration parameter. The parameter should contain a comma-separated list of two zone names (e.g., "1,2"). In this scenario, Mediant CE components will be evenly spread across these two zones. Note however, that such deployment topology may suffer from intermittent network latency between zones, which may affect internal communication between Mediant CE components and cause SC/MC switchovers.

Figure 3-6: Mediant CE Deployment Topology (Azure)

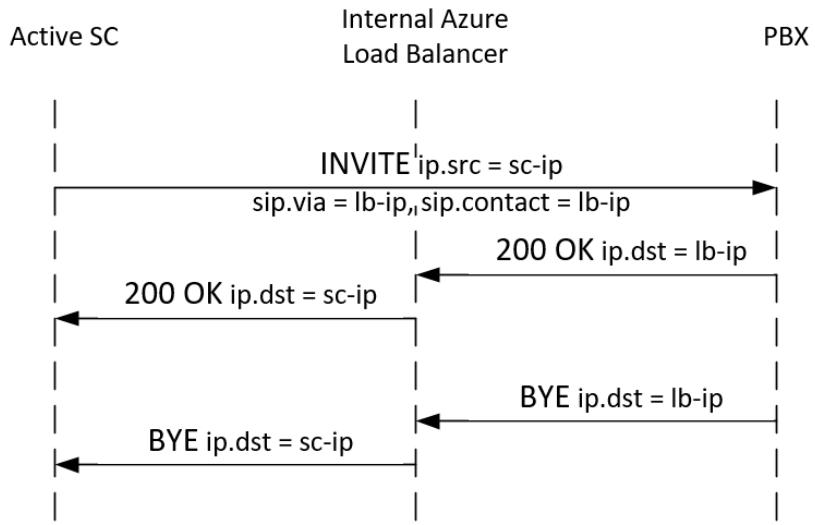


Azure Load Balancer is used to steer inbound (signaling and management) traffic towards active signaling components. Both public and internal Load Balancers are supported, enabling communication with signaling components via either public or private IP addresses respectively. The following limitations apply:

- When public IP addresses are used, Load Balancer also acts as a NAT gateway for outbound traffic. This ensures that all traffic arriving at the VoIP peer always has the public IP address of the Load Balancer as the source IP address at the IP layer. However, the source port is not preserved (e.g., SIP packets sent from port 5060 by

the active Signaling Component will arrive at the VoIP peer with a different port - for example 1024 – that is dynamically allocated by the Load Balancer).

- When private IP addresses are used, outbound traffic does not traverse through the Load Balancer. SIP headers (Via and Contact) contain the Internal Load Balancer's IP address and are used to route responses and subsequent dialogs via it; however, the source IP address at the IP layer contains the IP address of the Active SC instance.



Note: Outbound traffic may sometimes appear with the source IP address of the Internal Load Balancer at the IP layer as well. This may happen if the outbound flow occurs shortly after the inbound flow and is attributed to the existing DNAT translation in the Internal Load Balancer. Therefore, VoIP peers that communicate with Mediant CE via private IP addresses need to be configured to accept traffic from both the Internal Load Balancer IP and the private IP addresses of both SC instances.

- Communication with OVOC is performed via public or private IP addresses attached to the corresponding Azure Load Balancer. Refer to the *One Voice Operation Center User Manual* for detailed configuration instructions.
- Communication with media components is performed via either public or private IP addresses directly attached to them. Corresponding media traffic does not pass through the Load Balancer.

3.5 Public IP Addresses

During Mediant CE stack creation, Stack Manager lets you specify which subnets (and corresponding network interfaces) will be assigned with public IP addresses via the **Public IPs** parameter in the **Networking** section.

For each subnet that is configured to use a Public IP address, the following is created:

- Front-end rule with Public IP address on Azure Public Load Balancer
- Forwarding rules on Azure Public Load Balancer, which implement forwarding of incoming traffic towards the active SC instance
- Outbound rules on Azure Public Load Balancer, which implement SNAT translation for outbound traffic at the IP level
- Corresponding entry in the NAT Translation SBC configuration table, which implements SNAT translation for outbound traffic at the application level (SIP and SDP)

It is also possible to attach multiple public IP addresses to the same network interface. This may be done by configuring the **sc_public_ips** / **mc_public_ips** advanced configuration parameter (via **Advanced Config** section).

Note: When the **sc_public_ips / mc_public_ips** advanced configuration parameter is specified (via **Advanced Config** section), it overrides any value configured via the **Public IPs** parameter in the **Networking** section.

- **sc_public_ips**

  Contains a comma-separated list of subnet names (main, additional1, and additional2), which will be assigned with public IP addresses and optionally, with the number of public IP addresses on the corresponding network interface.

  For example:

  ```
  sc_public_ips = main:2,additional1
  ```

  attaches two public IP addresses to the network interface connected to the Main subnet (eth1) and one public IP address to the network interface connected to the Additional 1 subnet (eth2).

- **mc_public_ips**

  Same as above, but for MC network interfaces.

  For example:

  ```
  mc_public_ips = main,additional1:2
  ```

When the **sc_public_ips** / **mc_public_ips** advanced configuration parameter is specified, Stack Manager automatically creates secondary private IP addresses on the network interfaces that may be required for public IP assignment. The exact behavior depends on the component type:

- For MCs: First public IP address is attached to the primary private IP address; for each additional public IP address, corresponding secondary IP addresses are implicitly created.
- For SCs: Public IP addresses are always attached to the Public Azure Load Balancer and "mapped" to the corresponding private IP addresses; first public IP address is "mapped" to the primary private IP address; for each additional public IP address, corresponding secondary IP addresses are implicitly created.

3.6 Private IP Addresses

For each subnet that is configured not to use a Public IP address, the following is created:

- Front-end rule on Azure Internal Load Balancer
- Forwarding rule on Azure Internal Load Balancer, which implements forwarding of incoming traffic towards the active SC instance
- Corresponding entry in the NAT Translation SBC configuration table, which implements SNAT translation for outbound traffic at application levels (SIP and SDP)

It is also possible to use both private and public IP addresses on the same network interface (connected to a specific subnet) and/or use multiple private IP addresses on the same network interface. This may be done by configuring the **sc_additional_ips** / **mc_additional_ips** advanced configuration parameters (via **Advanced Config** section).

- **sc_additional_ips**

  Contains a comma-separated list of subnet names (main, additional1, and additional2), which will be assigned with additional private IP addresses and optionally, with the number of additional private IP addresses on the corresponding network interface.

  For example:

  ```
  sc_additional_ips = main,additional1:2
  ```

  creates one additional private IP address on the network interface connected to the Main subnet (eth1) and two additional private IP addresses on the network interface connected to the Additional 1 subnet (eth2).

- **mc_additional_ips**

  Same as above, but for MC network interfaces.

  For example:

  ```
  mc_additional_ips = main,additional1:2
  ```

The number of additional private IP addresses specified via the **sc_additional_ips** / **mc_additional_ips** advanced configuration parameter is added *on top* of any private IP addresses created by Stack Manager by default and/or due to the public IP addresses assigned to the specific network interface.

For example, the following configuration:

```
Cluster Subnet: <cluster-subnet-id>
Main Subnet: <main-subnet-id>
1st Additional Subnet: <additional-subnet-id>
Public IPs: "Main subnet"
Advanced Config:
    sc_additional_ips = main,additional1
```

creates the following networking configuration on signaling components:

- **eth0** – one primary IP address (used for internal communication between SC instances) and one secondary IP address (used for internal communication with MC instances)
- **eth1** – one primary and one secondary IP address
  - primary IP address – created implicitly and assigned with a public IP address (due to the **Public IPs** configuration parameter), placed behind Public Azure Load Balancer
  - 1st secondary IP address – created due to the **sc_additional_ips** advanced configuration parameter, placed behind Internal Azure Load Balancer

- **eth2** – one primary and one secondary IP address
  - primary IP address – created implicitly, placed behind Internal Azure Load Balancer
  - 1st secondary IP address – created due to the **sc_additional_ips** advanced configuration parameter, placed behind Internal Azure Load Balancer

3.7 Management Traffic

By default, the primary IP address of the "eth1" network interface, connected to the main subnet, is used for management traffic (Web, SSH, and SNMP).

If the main subnet is configured to use the Public IP address, this IP address is placed behind the Public Load Balancer and correspondingly, Mediant CE management should be performed via the corresponding Load Balancer's public IP address.

If the main subnet is configured not to use a Public IP address, this IP address is placed behind the Internal Load Balancer and correspondingly, Mediant CE management should be performed via the corresponding Load Balancer's internal IP address.

If you have both private and public IP addresses on the main subnet (placed behind the Internal and Public Load Balancers respectively) and want to manage Mediant CE via the private IP address, use the **oam_ip** parameter as follows:

```
Public IPs: "Main subnet"
Advanced Config:
    sc_additional_ips = main
    oam_ip = internal
```

The above configuration creates two IP addresses on the "eth1" network interface, connected to the main subnet:

- **eth1** – primary IP address, placed behind the Public Load Balancer and used for SIP traffic
- **eth1:1** – secondary IP address, placed behind the Internal Load Balancer and used for management traffic (Web, SSH, and SNMP)

4 Installation Prerequisites for Google Cloud Environment

Prior to installing Mediant CE in the Google Cloud environment, make sure that you meet the following prerequisites:

- You have a Google Cloud account. If you don't have a Google Cloud account, you can sign up for one on Google's website at https://cloud.google.com.
- You have uploaded AudioCodes Mediant VE/CE Image to the image repository. For more information, see AudioCodes Mediant CE Image on page 35.
- You have created all subnets needed for Mediant CE deployment, including the Cluster subnet and corresponding Firewall Rules. For more information, see Section Network Prerequisites on page 36.

4.1 AudioCodes Mediant CE Image

To deploy Mediant CE on Google Cloud, you must use the *Mediant VE/CE Image for Google Cloud*. For more information, go to https://www.audiocodes.com/library/firmware.

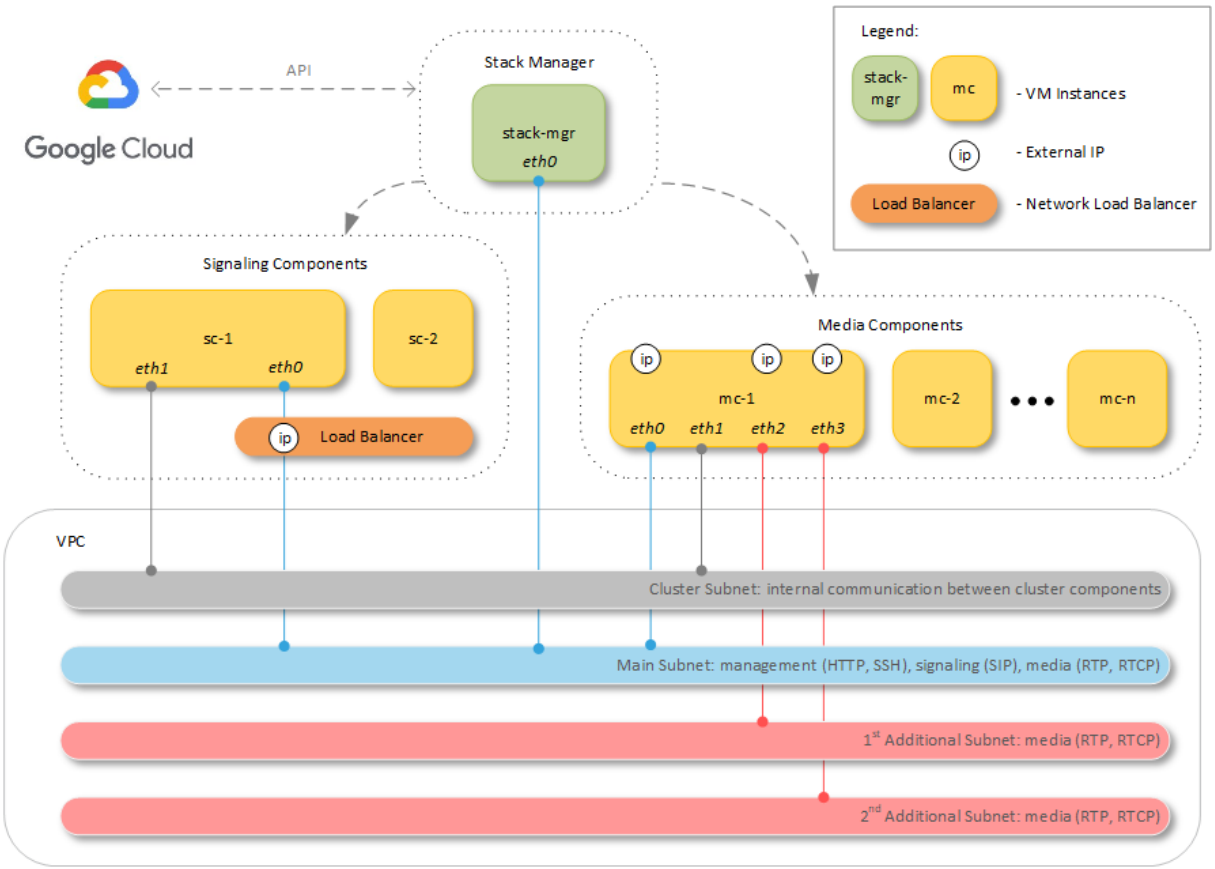
➢ **To upload Mediant CE image to Google Cloud image repository:**

1. Extract the .tar.gz file from the Mediant VE/CE Image for the Google Cloud .zip file.
2. In the Google Cloud Platform Console, go to the Storage > Browser page https://console.cloud.google.com/storage/browser.
3. Choose an existing bucket or create a new one.
4. Choose an existing folder(s) inside the bucket or create a new one if needed.
5. Click **Upload files**, and then select the Mediant VE/CE image for the Google Cloud .tar.gz file.
6. Wait until the upload completes.
7. Go to the Compute Engine > Images page https://console.cloud.google.com/compute/images.
8. Click **Create Image**.
9. Enter an image name.
10. Specify the source as the Cloud Storage file, and then choose the .tar.gz file that you uploaded in previous steps.
11. Specify the additional properties for your image (e.g. family or description).
12. Click **Create** to create the image.

4.2 Network Prerequisites

Mediant CE on Google Cloud uses the following network architecture:

Figure 4-1: Mediant CE Network Architecture – Google Cloud



Up to four subnet may be used:

- **Cluster Subnet:** For internal communication between Mediant CE components; connected to both SC and MC instances as the second network interface (eth1).
- **Main Subnet:** Carries management (HTTP, SSH, etc.), signaling (SIP) and media (RTP, RTCP) traffic; connected to both SC and MC instances as the first network interface (eth0) and to the Stack Manager instance.
- **1st and 2nd Additional Subnets:** Carries media (RTP, RTCP) traffic; connected to MC instances as the third and fourth network interfaces (eth2 and eth3) correspondingly. These subnets are optional, as the Main Subnet may carry all types of traffic.

Each subnet must reside in a different Virtual Network.

All needed subnets must be created prior to the Mediant CE deployment.

During deployment, Stack Manager creates all relevant Mediant CE components, including SC and MC instances, load balancer and external IP addresses.

4.2.1 Firewall Rules

On the Google Cloud platform, firewall rules are configured at network level rather than at the instance / network interface level. Therefore, you must manually configure them prior to the first Mediant CE deployment, as described below.

To simplify firewall rules configuration, Stack Manager assigns network tags to all created Mediant CE components. The following tags are created by default and may be customized by changing the **sc_tags** and **mc_tags** parameters in the stack configuration file.

- Signaling Components: sbc, sc
- Media Components: sbc, mc

The following firewall rules must be created for successful Mediant CE deployment:

Subnet	Name	Protocol	Port	Description	Target Tag
Cluster	udp-669	UDP	669	Internal communication between SC and MC instances	sbc
	udp-680	UDP	680	Internal communication between SC and MC instances	sbc
	http	TCP	80	Internal communication between SC and MC instances	sbc
	tcp-2424	TCP	2424	Internal communication between SC and MC instances	sbc
	tcp-2442	TCP	2442	Internal communication between SC and MC instances	sbc
	udp-925	UDP	925	Internal communication between SC and MC instances	sbc
	udp-3900	UDP	3900	Internal communication between SC and MC instances	sbc
Main	ssh	TCP	22	CLI management interface on active SC instance	sc
	http	TCP	80	Web management interface on active SC instance	sc
	https	TCP	443	Secure Web management interface on active SC instance	sc
	sip-udp	UDP	5060-5090	SIP signaling traffic on active SC instance	sc
	sip-tcp	TCP	5060-5090	SIP signaling traffic on active SC instance	sc
	media	UDP	6000-65535	RTP media traffic on MC instances	mc

Subnet	Name	Protocol	Port	Description	Target Tag
Additional Subnets	sip-udp	UDP	5060-5090	SIP signaling traffic on active SC instance	sc
	sip-tcp	TCP	5060-5090	SIP signaling traffic on active SC instance	sc
	media	UDP	6000-65535	RTP media traffic on MC instances	mc

➢ **To create Firewall Rules:**

1. In the Google Cloud Platform Console, go to the VPC Network > Firewall Rules page https://console.cloud.google.com/networking/firewalls.
2. Click **Create Firewall Rule** to create a new firewall rule.
3. Create firewall rules as per the table above:
   - Direction of traffic: Ingress
   - Action on match: Allow
   - Targets: Specified target tags
   - Tag name: <tag>
   - Source filter: IP ranges
   - Source IP ranges: 0.0.0.0/0
   - Protocols and ports: Specified protocol and ports
     - <protocol>: <ports>

4.3 Machine Types

The following instance types are used by default Mediant CE deployment:

- **SC instances:** n1-standard-8
- **Forwarding MC instances:** n1-standard-2 or custom-4-8192 (depending on number of network interfaces)
- **Transcoding MC instances:** custom-16-16384

You may customize instance types by specifying the **sc_instance_type** and/or **mc_instance_type** advanced configuration parameters (via **Advanced Config** section) during stack creation.

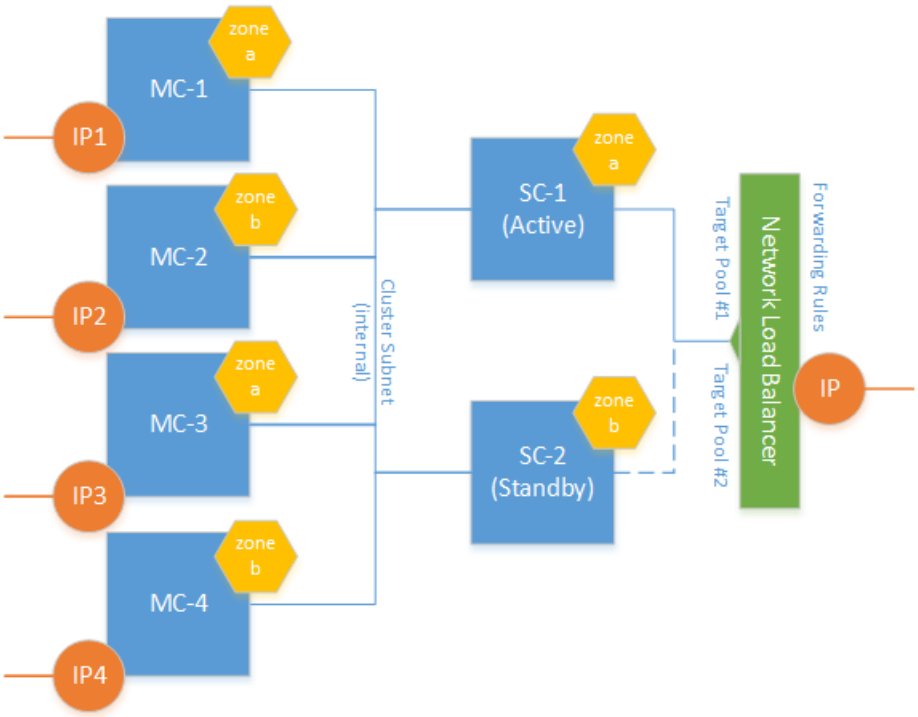
For example:

```
sc_instance_type = n1-standard-4
mc_instance_type = custom-8-8192
```

4.4 Deployment Topology

Mediant CE components are deployed across two availability zones of the Google Cloud region.

Figure 4-2: Mediant CE Deployment Topology (Google)



Communication with SCs is performed via the IP addresses attached to Google Load Balancer that steers inbound (signaling and management) traffic towards the active SC. The following load balancer types are used:

- Network Load Balancer for external IP addresses
- Internal Load Balancer for internal IP addresses

Google Load Balancer doesn't perform NAT translation and forwards traffic without modifying the IP packet's destination address. Therefore, IP addresses (external and internal) attached to the Load Balancer are configured as secondary IP addresses in both SC instances and used for all applications instead of primary IP addresses. For example, SIP Interfaces should be connected to secondary IP address eth0:1, and not to the primary IP address eth0.

Since Network Load Balancer supports only primary VM network interface, external IP addresses may be used to communicate with signaling components (SCs) only via the Main subnet (connected to eth0 network interface). Multiple public IP addresses are supported.

Internal IP addresses may be used to communicate with signaling components (SCs) via Internal Load Balancer that may be connected to all available subnets (Main, Additional 1, and Additional 2).

Communication with media components (MCs) is performed via internal and external IP addresses directly attached to them and doesn't require any Load Balancer configuration.

If an external IP address is assigned to the Main subnet, it's also used for management traffic (Web, SSH, SNMP).

4.5 External IP Addresses

During Mediant CE stack creation, Stack Manager lets you specify which subnets (and corresponding network interfaces) will be assigned with public (external) IP addresses via the **Public IPs** parameter in the **Networking** section.

For each subnet that is configured to use an External IP address, the following is created:

- External IP address
- Target Pools that implement keep-alive and traffic steering towards the active SC instance
- Forwarding Rules (of the Network Load Balancer) that implement forwarding of incoming traffic
- Secondary IP address entries in the network Interfaces SBC configuration table of both SC instances. Applications (e.g. SIP Interfaces) should be bound to these secondary IP addresses, and not to primary IP addresses.

Due to the Google's Network Load Balancer limitations, external IP addresses may be assigned only to the Main subnet (connected to primary network interface eth0).

It is also possible to assign multiple external IP addresses to the same network interface. This may be done by configuring the **sc_public_ips** / **mc_public_ips** advanced configuration parameter (via **Advanced Config** section).

Note: When the **sc_public_ips / mc_public_ips** advanced configuration parameter is specified (via **Advanced Config** section), it overrides any value configured via **Public IPs** parameter in the **Networking** section.

- **sc_public_ips**

  Contains a comma-separated list of subnet names (main, additional1, and additional2), which will be assigned with external IP addresses, and optionally, with the number of external IP addresses on the corresponding network interface.

  For example:

  ```
  sc_public_ips = main:2,additional1
  ```

  attaches two external IP addresses to the network interface connected to the Main subnet (eth0) and one external IP address to the network interface connected to the Additional 1 subnet (eth2).

- **mc_public_ips**

  Same as above, but for MC network interfaces.

  For example:

  ```
  mc_public_ips = main,additional1:2
  ```

When the **sc_public_ips** / **mc_public_ips** advanced configuration parameter is specified, Stack Manager automatically creates secondary private IP addresses on the network interfaces that may be required for external IP assignment. The exact behavior depends on the component type:

- **For MCs:** first external IP address is attached to the primary private IP address; for each additional external IP address corresponding secondary IP addresses are implicitly created.
- **For SCs:** external IP addresses are always attached to the Network Load Balancer and corresponding secondary IP address entries are implicitly created in the SBC Interface table.

4.6 Internal IP Addresses

For each subnet that is configured not to use an External IP address, the following is created:

- Two Regional Backend Services with Internal IP addresses – one for UDP traffic and one for TCP traffic
- Instance Groups that implement keep-alive and traffic steering towards the active SC instance
- Forwarding Rules (of the Internal Load Balancer) that implement forwarding of incoming traffic
- A pair of Secondary IP address entries (ethX.udp and ethX.tcp) in the network Interfaces SBC configuration table of both SC instances. Applications (e.g. SIP Interfaces) should be bound to these secondary IP addresses, and not to primary IP addresses.

It is also possible to use both internal and external IP addresses on the same network interface (connected to a specific subnet) and/or use multiple internal IP addresses on the same network interface. This may be done by configuring the **sc_additional_ips** / **mc_additional_ips** advanced configuration parameters (via **Advanced Config** section).

- **sc_additional_ips**

  Contains a comma-separated list of subnet names (main, additional1, and additional2), which will be assigned with additional private IP addresses and optionally, with the number of additional private IP addresses on the corresponding network interface.

  For example:

  ```
  sc_additional_ips = main,additional1:2
  ```

  creates a pair of additional private IP addresses (for UDP and TCP traffic) on the network interface connected to the Main subnet (eth0) and two additional private IP address pairs on the network interface connected to the Additional 1 subnet (eth2).

- **mc_additional_ips**

  Same as above, but for MC network interfaces.

  For example:

  ```
  mc_additional_ips = main,additional1:2
  ```

This page is intentionally left blank.

5 Installation Prerequisites for OpenStack Environment

Prior to installing Mediant CE in the OpenStack environment, make sure that you meet the following prerequisites:

- You have uploaded AudioCodes Mediant VE/CE Image to the image repository. For more information, see Section AudioCodes Mediant CE Image on page 43.
- You have created all subnets needed for Mediant CE deployment, including the Cluster subnet. For more information, see Section Network Prerequisites on page 43.

5.1 AudioCodes Mediant CE Image

To deploy Mediant CE on OpenStack, you must use the *Mediant VE/CE QCOW2 Image for KVM/OpenStack*. For more information, go to https://www.audiocodes.com/library/firmware.

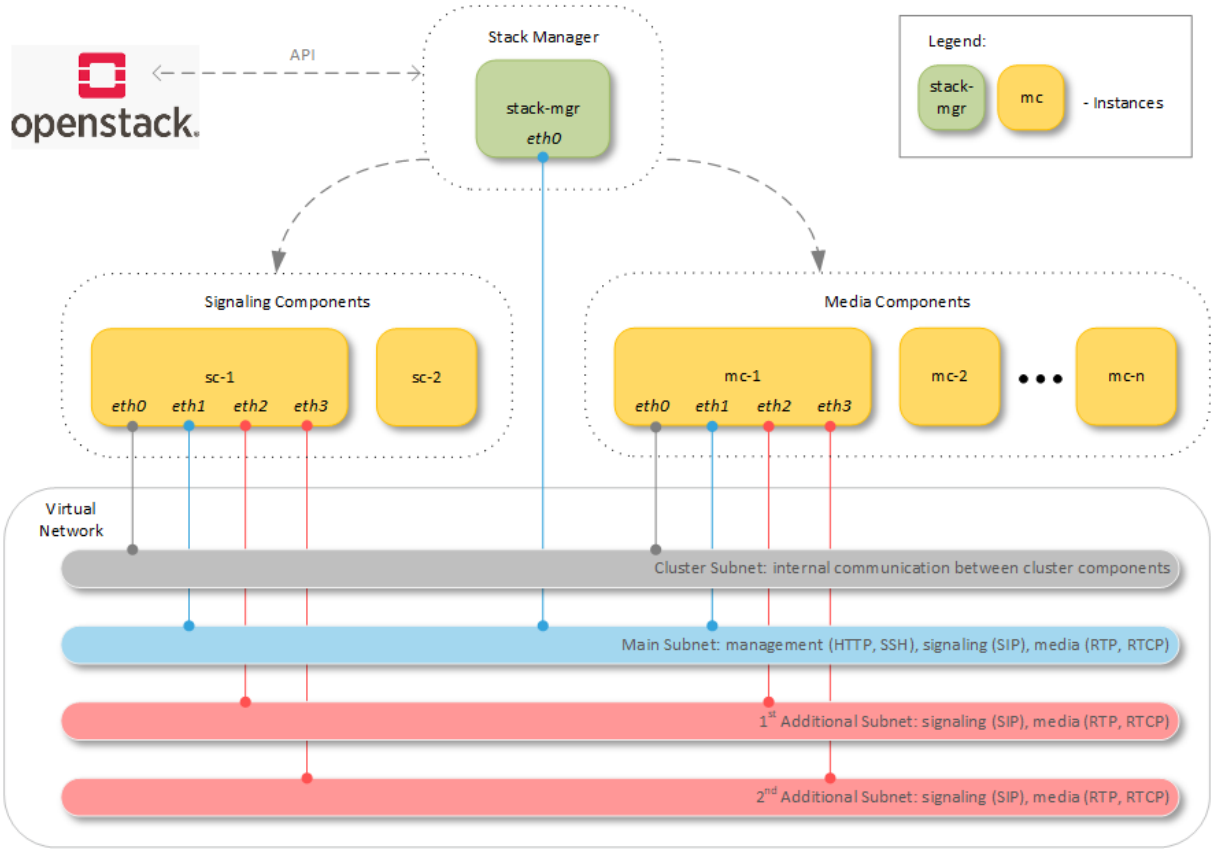
Upload the image to OpenStack image repository, using the following command:

```
# openstack image create --disk-format qcow2 \
  --container-format bare --public \
  --file ./sbc-F7.20A.202.204.qcow2 sbc-F7.20A.202.204
```

5.2 Network Prerequisites

Mediant CE on OpenStack uses the following network architecture:

Figure 5-1: Mediant CE Network Architecture – OpenStack



Up to four subnet may be used:

- **Cluster Subnet:** For internal communication between Mediant CE components; connected to both SC and MC instances as the first network interface (eth0).
- **Main Subnet:** Carries management (HTTP, SSH, etc.), signaling (SIP) and media (RTP, RTCP) traffic; connected to both SC and MC instances as the second network interface (eth1) and to the Stack Manager instance.
- **1st and 2nd Additional Subnets:** Carries signaling (SIP) and media (RTP, RTCP) traffic; connected to MC instances as the third and fourth network interfaces (eth2 and eth3) correspondingly. These subnets are optional, as the Main Subnet may carry all types of traffic.

All needed subnets must be created prior to Mediant CE deployment.

5.3 Instance Flavors

It is recommended to use the following instance flavors for Mediant CE components:

- SC instances: 4 vCPU (non-hyperthreaded), 32GB RAM
- Forwarding MC instances: 1 vCPU (non-hyperthreaded), 4GB RAM
- Transcoding MC instances: 8 vCPU (non-hyperthreaded), 8GB RAM

6 Installation for Non-Cloud Environments (e.g. VMware)

Prior to installing Mediant CE in a non-cloud environment (e.g. VMware), make sure that you meet the following prerequisites:

- You have AudioCodes Mediant VE/CE Image for your environment (e.g. OVF image for VMware). Images can be downloaded from AudioCodes website at https://www.audiocodes.com/library/firmware.
- All subnets needed for Mediant CE deployment are available, including the Cluster subnet. For more information, see the following section.

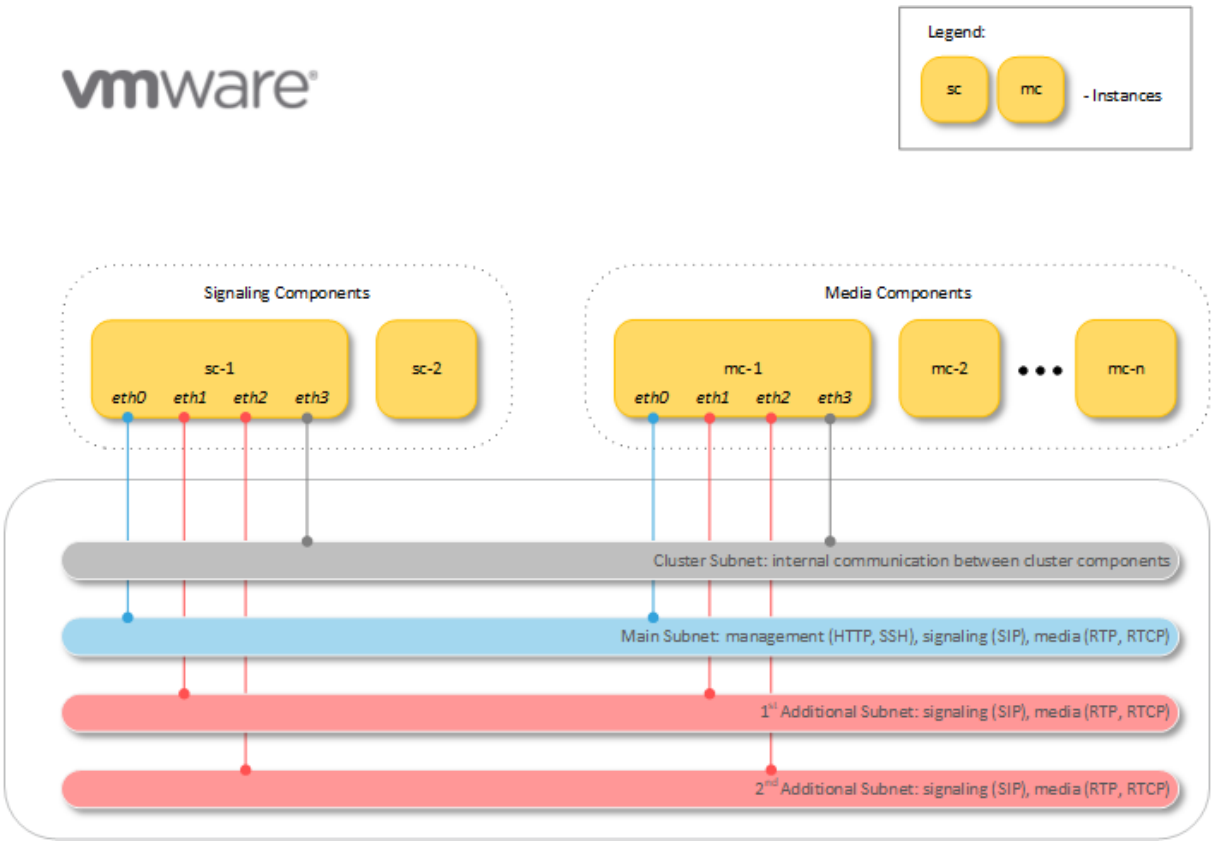
6.1 Prerequisites

This section describes the prerequisites.

6.1.1 Network Prerequisites

Mediant CE in non-cloud environments (e.g. VMware) uses the following network architecture:

Figure 6-1: Mediant CE Network Architecture – Non-Cloud Environments (e.g., VMware)



Up to four subnets may be used:

- **Cluster Subnet:** For internal communication between Mediant CE components.
- **Main Subnet:** Carries management (HTTP, SSH, etc.), signaling (SIP), and media (RTP, RTCP) traffic.

- **1st and 2nd Additional Subnets:** Carries signaling (SIP) and media (RTP, RTCP) traffic. These subnets are optional because the Main Subnet may carry all types of traffic.

The 1st network interface (eth0) is typically connected to the Main Subnet. The last network interface is typically connected to the Cluster Subnet.

6.1.2 Virtual Machine Types

The recommended virtual machine types for Mediant CE components depend on the host's CPU type:

- **Prior to Intel® Xeon® Scalable Processors:**
  - SC instances:
    - 4 vCPU (non-hyperthreaded, 4 physical cores)
    - 32GB RAM, 50GB Storage
  - Forwarding-only MC instances:
    - 1 vCPU (non-hyperthreaded, 1 physical core)
    - 4GB RAM, 10GB Storage
  - Transcoding MC instances:
    - 8 vCPU (non-hyperthreaded, 8 physical cores)
    - 8GB RAM, 10GB Storage

- **Intel® Xeon® Scalable Processors or later:** It is recommended to utilize Hyper-Threading capability, which provides improved performance while using lower CPU resources.
  - SC instances:
    - 4 vCPU (hyperthreaded, 2 physical cores)
    - 32GB RAM, 50GB Storage
  - Forwarding-only MC instances:
    - 2 vCPU (hyperthreaded, 1 physical core)
    - 4GB RAM, 10GB Storage
  - Transcoding MC instances:
    - 8 vCPU (hyperthreaded, 4 physical cores)
    - 8GB RAM, 10GB Storage

6.2 Redundancy Deployment Options

The following table describes redundancy options for deployment:

Component Module	Redundancy Protection		Number of Components Required	Servers (Hosts) Deployment
	Software Failure Only	Software and Server (Host) Failures		
SC (Signaling Components)	-	-	1	Single SC on a single server
	+	-	1+1	Both SCs can reside on the same server, or different servers
	+	+	1+1	Each SC on a different server
MC (Media Components)	-	-	N [1]	Each server can occupy as many MCs as possible
	+	-	At least N+1	Each server can occupy as many MCs as possible
	+	+	At least $\frac{N*S}{S-1}$	Each server should occupy at least $\frac{N}{S-1}$ MCs

Note: *N* is the number of MCs required to reach the required media capacity (forwarding and transcoding).

6.2.1 Protection from Hardware and Software Failure

When protection from both hardware and software failure is required on the SC, then the two host servers should occupy a single SC on each.

When protection from both hardware and software failure is required on the MC, then the minimum number of servers required for allocation of MCs can be calculated as follows:

$$S \geq \frac{N + Ns}{Ns}$$

Where:

- ***S*** – number of servers required (minimum 2).
- ***N*** –number of MCs required to reach the required media capacity (forwarding and transcoding).
- ***Ns*** –maximum number of MCs that can be installed on a single server.

6.3 Installation

Installing SCs and MCs is required as follows:

1. For each server (host) in the deployment, configure BIOS settings according to Section 3.1 "Configuring the Server's BIOS" in the *Mediant Virtual Edition SBC Installation Manual Ver. 7.2*.
2. Install the virtual machine for the SC, according to Section 3 in the *Mediant Virtual Edition SBC Installation Manual Ver. 7.2*. For example, installing the SBC on VMware vSphere ESXi Ver. 6.7 is according to the following sections:
   - Section 3.2.2 "Installing Mediant VE SBC on VMware vSphere ESXi Ver. 6.5 or later".
   - Sections 3.7 and 3.8 (Section 3.9 is not relevant for Mediant CE).
3. Repeat Step 2. for creating and initial installation of the virtual machine for each MC in the system, according to the number of MCs required per the "Redundancy Deployment Options" described in Section 6.2 Redundancy Deployment Options.
4. If SC redundancy is required, then:
   1. Repeat Step 2 for installation of another SC. The second SC should be located on the same server (host) or on a different server according to the "Redundancy Deployment Options" described in Section 6.2 Redundancy Deployment Options.
   2. Follow the instructions for installing an HA system according to Section 3.13 in the *Mediant Virtual Edition SBC Installation Manual Ver. 7.2*.
5. Follow the instructions in Section 7.2 Deployment via Manual Installation and Configuration.

7 Deploying Mediant CE

This chapter describes Mediant CE deployment.

7.1 Deployment via Stack Manager

Deployment of Mediant CE is performed using the Stack Manager tool. This deployment method features:

- Simplified Mediant CE deployment, ensuring all needed resources are properly created and configured
- Resizing and adjustment of Mediant CE resources to actual service needs – both manual and automatic
- Complete Mediant CE lifecycle, including update of Mediant CE network topology, software upgrade of all its components, north-bound API for integration with orchestration tools and others
- Simplified Mediant CE termination, ensuring all resources corresponding to the Mediant CE are properly removed

➢ **To deploy Mediant CE:**

1. Install the Stack Manager tool, as described in the *Stack Manager User's Manual*, which you can download from AudioCodes website at https://www.audiocodes.com/library/technical-documents.
2. Create a new Mediant CE stack via Stack Manager's **create** command, as described in the *Stack Manager User's Manual*.

During Mediant CE deployment in Azure and AWS environments, you will be prompted to choose the **OS Version** for the deployed Mediant CE instance:

- **6:** This version corresponds to the 7.20**A** stream, which is based on CentOS 6
- **8:** This version corresponds to the new 7.20**CO** stream, which is based on CentOS 8 and provides significantly better performance and capacity (refer to the *SBC-Gateway Series Release Notes* for details)

Note: The 7.20CO stream ('OS version': **8**) is currently available for Azure and AWS environments only.

7.1.1 Deployment Troubleshooting

Stack Manager uses cloud-native orchestration engines to perform deployment:

- AWS: Cloud Formation templates
- Azure: Azure Resource Manager (ARM) templates
- OpenStack: Heat templates
- Google: Deployment Manager templates

If Mediant CE deployment fails and the error description provided by Stack Manager is not detailed enough, refer to the corresponding orchestration engine's detailed logs for additional information.

7.2 Deployment via Manual Installation and Configuration

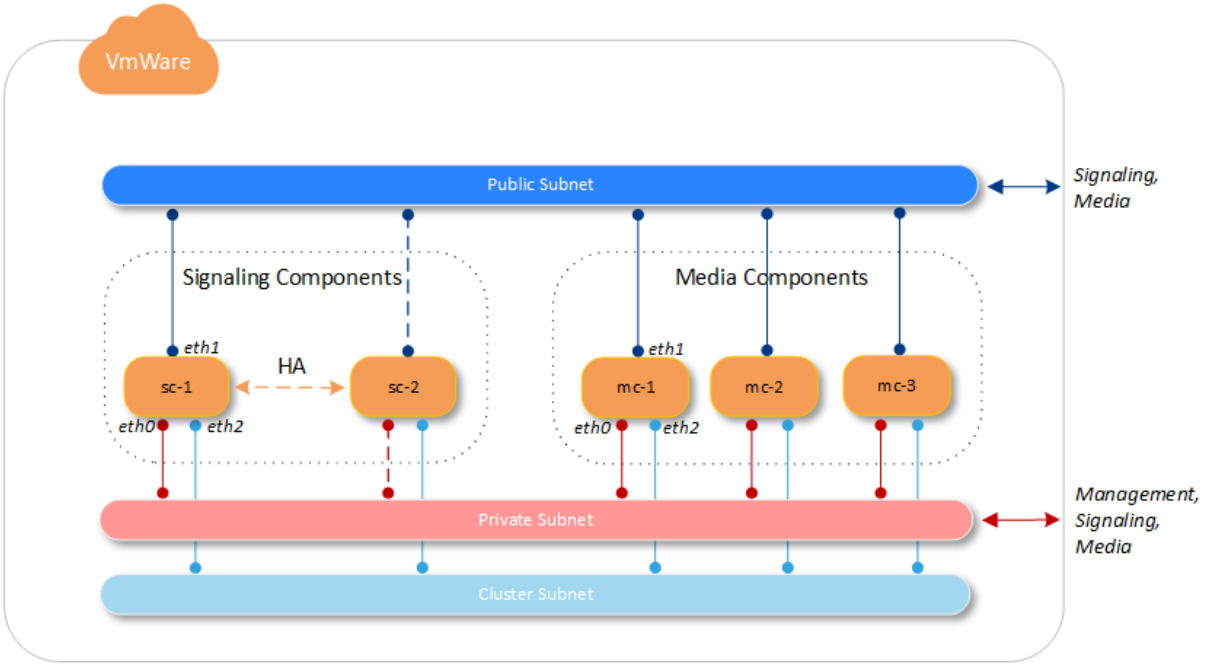
This deployment method enables Mediant CE deployment in non-cloud virtualized environments (e.g., VMware). All needed resources (e.g., subnets and virtual machines) must be manually created and properly configured by the operator, as described below.

As this deployment method doesn't include a "management component", automatic scaling is not supported. Manual scaling may be done by creating and configuring additional resources, but it is considerably more complicated than when using Stack Manager.

Note: For supported cloud environments, you should deploy Mediant CE using the Stack Manager tool, as described previously.

The following instructions describe the following Mediant CE deployment example:

Figure 7-1: Sample Mediant CE Deployment In VMware



The deployment consists of:

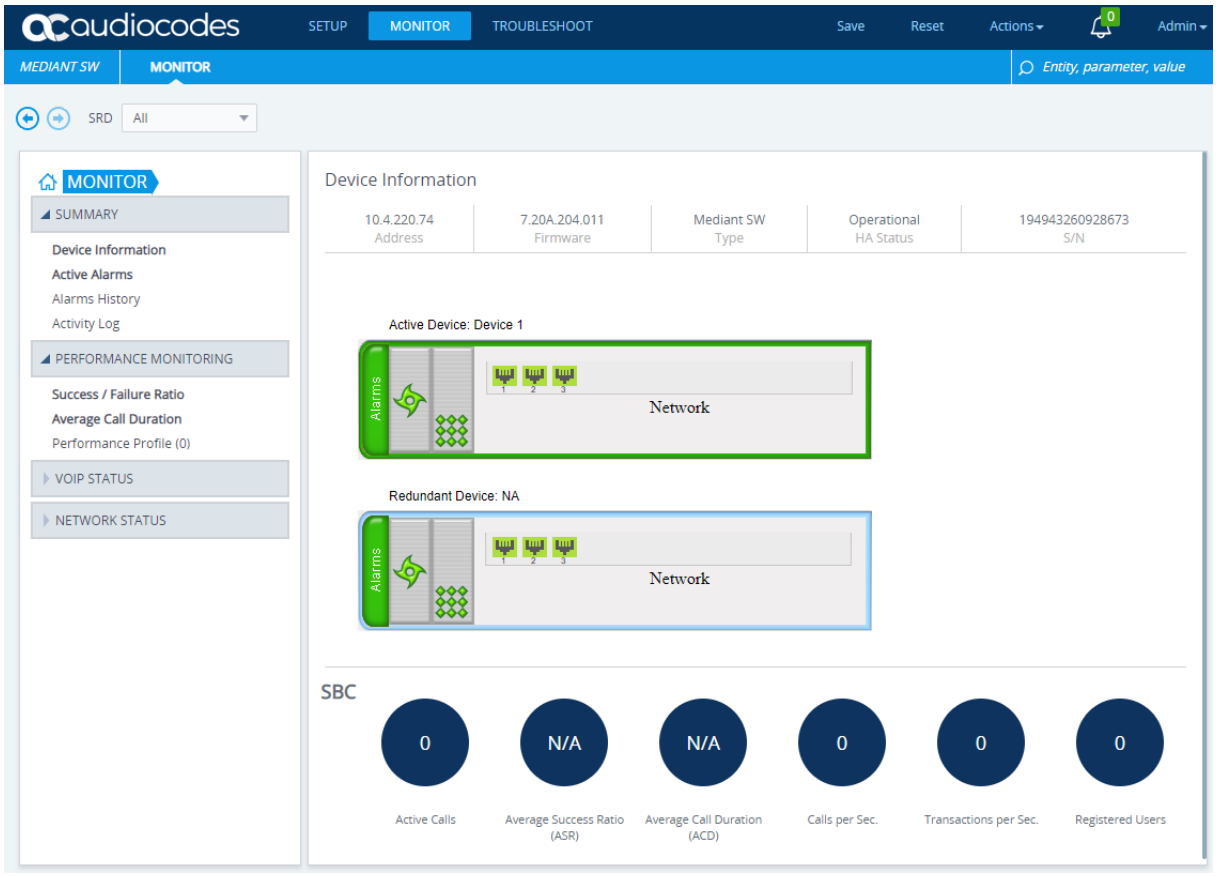
- Two signaling components: sc-1 and sc-2
- Three media components: mc-1, mc-2, and mc-3
- Private subnet, which is used for management (e.g., SSH and HTTP), signaling (SIP), and media (RTP) traffic
- Public subnet, which is used for signaling (SIP) and media (RTP) traffic
- Cluster subnet, which is used for internal communication between Mediant CE components

➢ **To deploy Mediant CE:**

1. Create virtual machines for all Mediant CE components.
2. Connect all virtual machines to the subnets:
   - eth0 (1st network port) – private subnet
   - eth1 (2nd network port) – public subnet
   - eth2 (3rd network port) – cluster subnet
3. Configure IP addresses on the 1st signaling component (sc-1):
   - eth0 – Application Type is **O+C+M**
   - eth1 – Application Type is **C+M**
   - eth2 – Application Type is **Maintenance (HA)**
4. Configure IP addresses on the 2nd signaling component (sc-2):
   - eth0 – Application Type is **O+C+M**
   - eth2 – Application Type is **Maintenance (HA)**
5. Configure IP addresses on the media component (mc-1, mc-2, and mc-3):
   - eth0 – Application Type is **O+C+M**
   - eth1 – Application Type is **C+M**
   - eth2 – Application Type is **Cluster**
6. Configure HA connection between signaling components:
   a. On the 1st signaling component (sc-1):
      a. Open the HA Settings page (**Setup** menu > **IP Network** tab > **Core Entities** folder > **HA Settings**).
      b. Configure the 'HA Remote Address' parameter to the Maintenance IP address (eth2) of the 2nd signaling component (sc-2).
      c. Save the configuration.
   b. On the 2nd signaling component (sc-2):
      a. Open HA Settings page (**Setup** menu > **IP Network** tab > **Core Entities** folder > **HA Settings**).
      b. Configure the 'HA Remote Address' parameter to the Maintenance IP address (eth2) of the 1st signaling component (sc-1).
      c. Save the configuration.
   c. Reset the 1st signaling component and wait until it boots up.
   d. Reset the 2nd signaling component. When the reset completes, the 2nd signaling component establishes HA connection with the 1st signaling component and loses all its networking configuration, except for the Maintenance IP address. Therefore, you will be unable to access its Web interface. Instead, you should check its status on the **Monitor** page on the Web interface of the 1st signaling component.

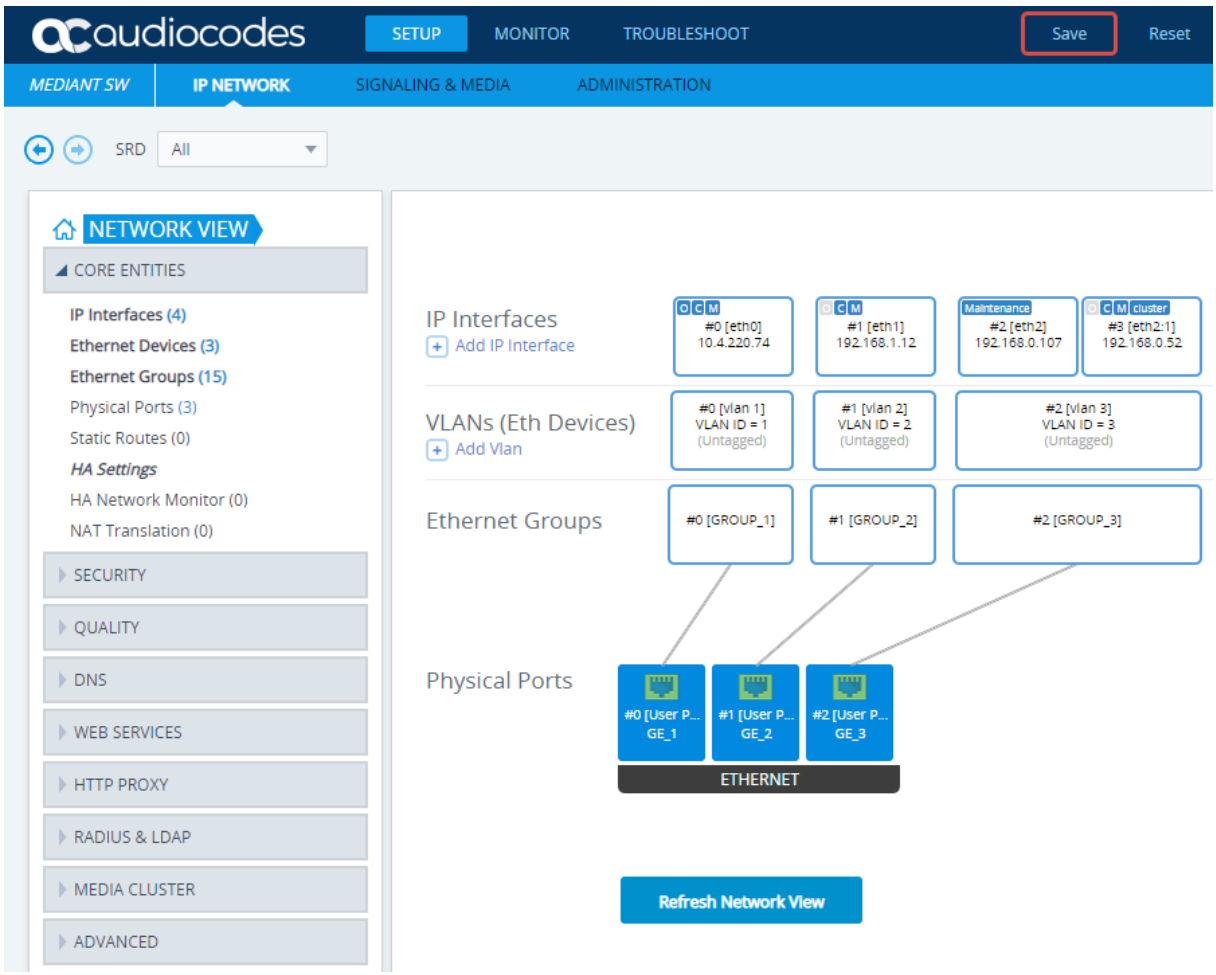
e. Wait until the HA connection between signaling components is fully established and **Monitor** page shows the 'HA Status' as "Operational" and both Active and Redundant devices are visible.

Figure 7-2: HA Connection Between Signaling Components



7. Add the cluster IP address to the signaling components:
   a. On the 1st signaling component (sc-1), open the Interfaces table (**Setup** menu > **IP Network** tab > **Core Entities** folder > **IP Interfaces**).
   b. Add an additional (secondary) IP address to the VLAN that is attached to the 3rd network interface (eth3).
   c. Configure the 'Application Type' parameter to **Cluster** for this additional IP address.

Figure 7-3: Network Configuration on Signaling Components

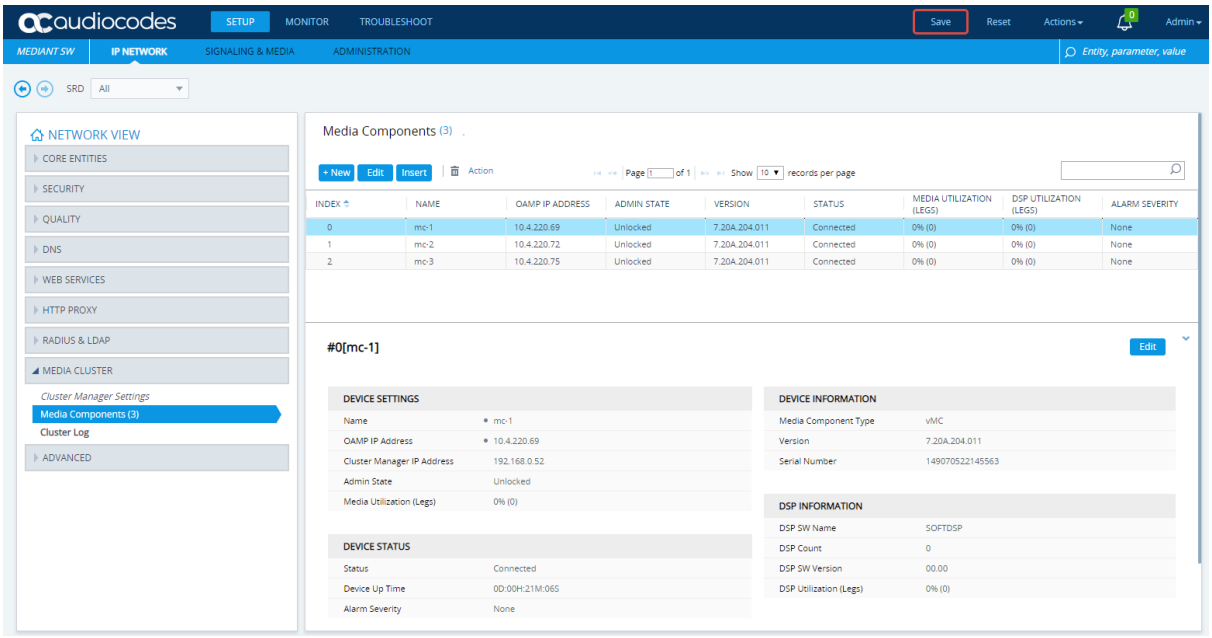


8. Configure signaling components to operate in Media Cluster mode:
   a. On the 1st signaling component (sc-1), open the Cluster Manager Settings page (**Setup** menu > **IP Network** tab > **Media Cluster** folder > **Cluster Manager Settings**).
      - Configure the 'Cluster Mode' parameter to **Media Cluster**.
      - Configure the 'Device Role' parameter to **Signaling Component**.
   b. Save the configuration.
   c. Reset the device to activate the new operation mode.
9. Configure media components (mc-1, mc-2, mc-3) to operate in Media Cluster mode:
   a. On each media component (mc-1, mc-2, mc-3), open the Cluster Manager Settings page (**Setup** menu > **IP Network** tab > **Media Cluster** folder > **Cluster Manager Settings**).
      - Configure the 'Cluster Mode' parameter to **Media Cluster**.
      - Configure the 'Device Role' parameter to **Media Component**.

b. Refresh the navigation menu, by clicking the browser's **Reload** button or using the Ctrl+R shortcut key.
c. Open the **MC Settings** page (**Setup** menu > **IP Network** tab > **Media Cluster** folder > **MC Settings**).
   - Configure the 'Cluster Manager IP Address' parameter to the Cluster IP address of the signaling component (added in Step 7).
   - Configure the 'Media Component Profile' parameter to match the intended operational mode of the media components.
d. Save the configuration.
e. Reset the device to activate the new configuration.

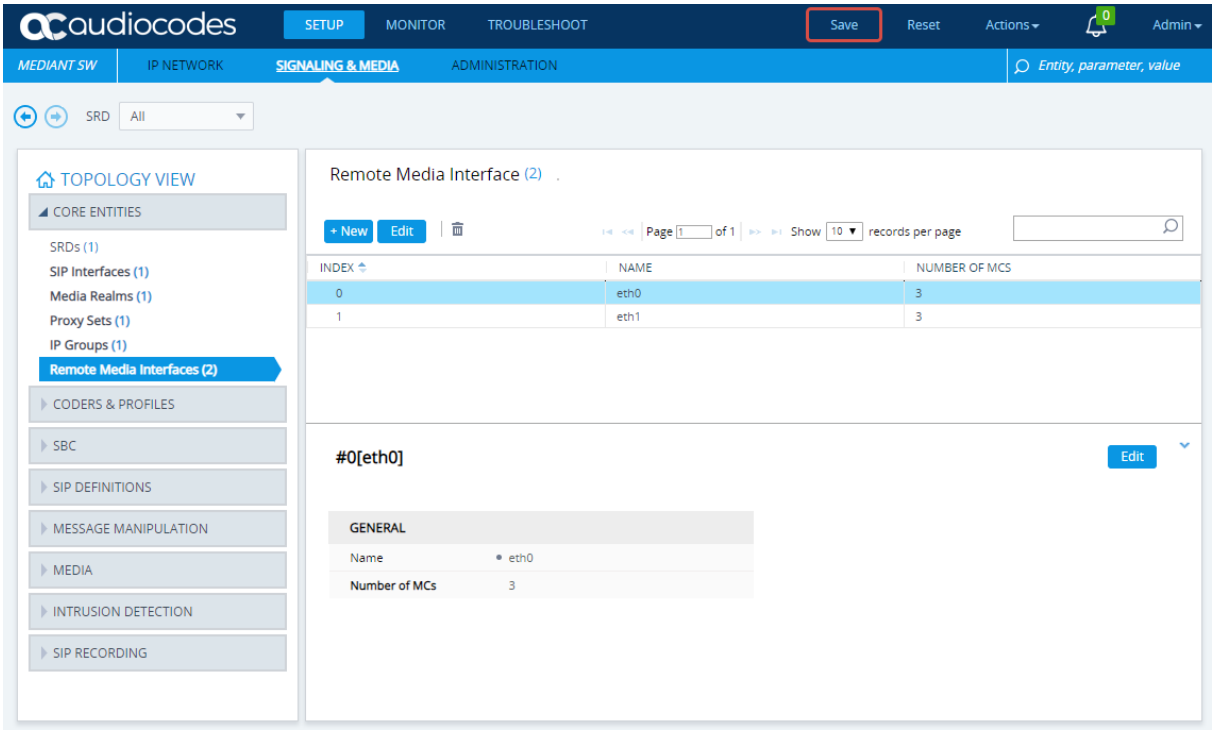
10. Configure signaling components to operate with media components:
    a. On the 1st signaling component (sc-1), open the Media Components page (**Setup** menu > **IP Network** tab > **Media Cluster** folder > **Media Components**).
    b. Click **New** to add new media component entry.
    c. Configure the media component name and corresponding OAM IP address (assigned to eth0 interface).
    d. Repeat the above steps for all media components.
    e. Save the configuration.
    f. Wait until the **Status** of all media components displays "Connected".

Figure 7-4: Media Components Configuration and Status Table



11. Configure Remote Media Interfaces on signaling components:
    a. On the 1st signaling component (sc-1), open the Remote Media Interfaces page (**Setup** menu > **Signaling & Media** tab > **Core Entities** folder > **Remote Media Interfaces**).
    b. Click **New** to add a new Remote Media Interface.
    c. Enter the name of the network interface on Media Components that is capable of handling media traffic (e.g., "eth0" or "eth1" in our example).
    d. Repeat the above steps for all network interfaces on the Media Components that are capable of handling media traffic.
    e. Verify that the 'Number of MCs' for each configured interface matches the actual number of Media Components (three in our example).

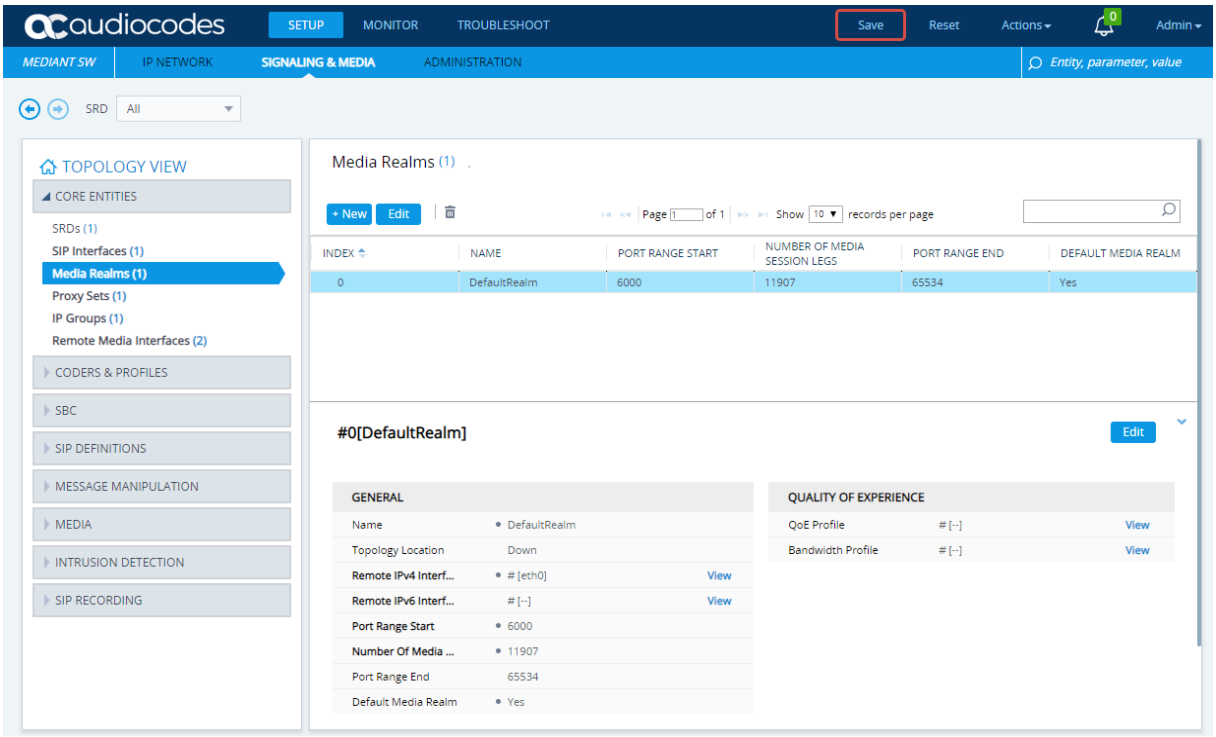
Figure 7-5: Remote Media Interfaces Configuration



12. Update Media Realms configuration on signaling components:
    a. On the 1st signaling component (sc-1), open the Media Realms page (**Setup** menu > **Signaling & Media** tab > **Core Entities** folder > **Media Realms**).
    b. Click **Edit** to edit the default Media Realm.
    c. Configure **Remote IPv4 Interface Name** to reference one of the Media Component's network interfaces, configured as Remote Media Interfaces in Step 12.

All traffic associated with this Media Realm will be sent/received via the corresponding network interface on one of the Media Components. If you need to define additional Media Realms, configure them in a similar manner. In other words, configure **Remote IPv4 Interface Name** or **Remote IPv6 Interface Name** to associate the Media Realm with the corresponding network interface on one of the Media Components. Mediant CE automatically distributes calls across available Media Components, choosing the proper network interface and port range as configured for the Media Realm.

Figure 7-6: Media Realms Configuration



13. If one of your subnets resides behind NAT device, configure NAT translation as follows:

    For each Media Component (mc-1, mc-2, and mc-3):

    a. Open the NAT Translation page (**Setup** menu > **IP Network** tab > **Core Entities** folder > **NAT Translation**).

    b. Click **New** to create a new NAT Translation rule, and then configure it as follows:

    - Configure the 'Source Interface' parameter to reference the corresponding network interface (e.g. eth1).
    - Configure the 'Source Start Port' parameter to **1**.
    - Configure the 'Source End Port' parameter to **65535**.
    - Configure the 'Target IP Address' parameter to match the public IP address of the NAT device (e.g., 10.6.2.101).
    - Configure the 'Target Start Port' parameter to **1**.
    - Configure the 'Target End Port' parameter to **65535**.

    c. Reset the Media Component to activate the new configuration.

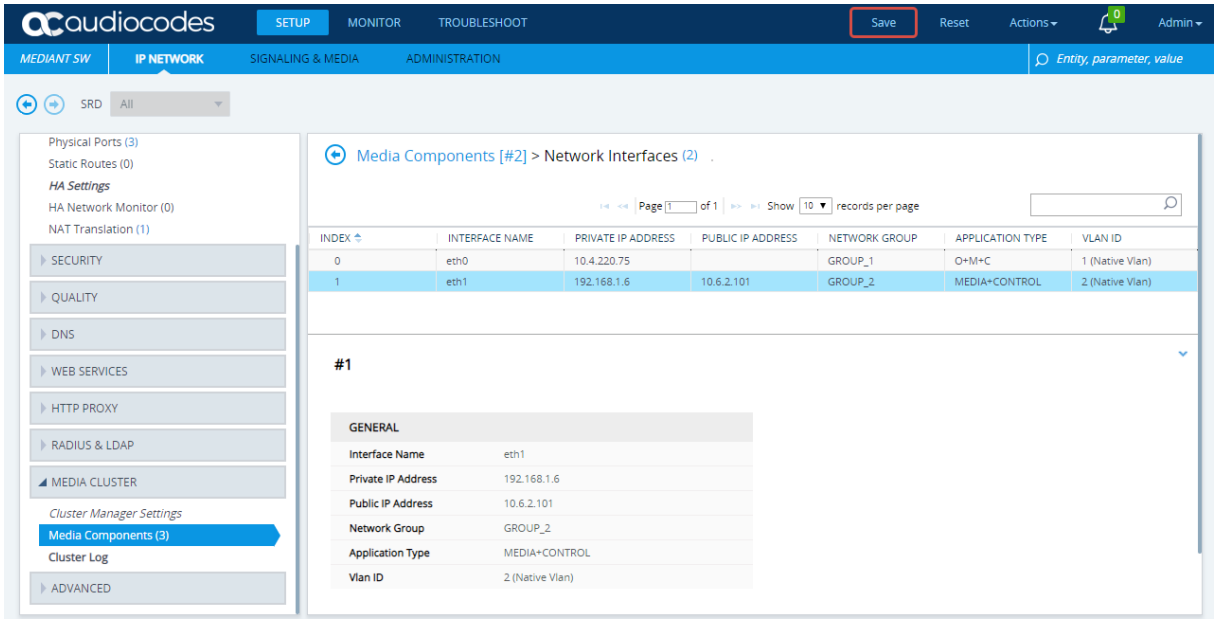
    d. Repeat the above steps for all Media Components.

On the 1st signaling component (sc-1):

a. Open the Media Components page (**Setup** menu > **IP Network** tab > **Media Cluster** folder > **Media Components**).

b. For each entry that corresponds to the specific Media Component, click the **Network Interfaces** link at the bottom of the page, and then verify that the Public IP Address is properly detected for relevant interfaces.

Figure 7-7: Verifying Public IP Address of the Media Component



14. Open the NAT Translation page (**Setup** menu > **IP Network** tab > **Core Entities** folder > **NAT Translation**).

15. Click **New** to create a new NAT Translation rule, and then configure it as follows:

- Leave the 'Source Interface' parameter empty.
- Configure the 'Remote Interface Name' parameter to reference the corresponding Media Component's network interface (e.g., eth1).
- Configure the 'Source Start Port' parameter to **1**.
- Configure the 'Source End Port' parameter to **65535**.
- Configure the 'Target IP Mode' parameter to **Automatic**.
- Configure the 'Target Start Port' parameter to **1**.
- Configure the 'Target End Port' parameter to **65535**.

Mediant CE will automatically perform NAT Translation, using the Public IP address of the Media Component that handles the specific call.

16. Your basic Mediant CE configuration is complete. You should now configure the SIP application, as described in the *Mediant VE/CE User's Manual* and perform some basic calls to verify correct system operation.

This page is intentionally left blank.

8 Managing Mediant CE

Mediant CE management is performed through the Web, CLI, and REST management interfaces provided by the active SC component. These management interfaces are accessible as follows:

- Azure: via "eth1" private or public IP address assigned to the Azure Load Balancer
- Google: via the primary External IP address assigned to the Network Load Balancer
- AWS, OpenStack and other environments: via "eth1" private or public IP addresses assigned to the active signaling component

All Mediant CE management operations are performed through the above described management interface. There is no need to access management interfaces on other components (e.g., on media components) and such access is blocked by default security rules.

8.1 Default Security Rules

Note: This section is not applicable to Google Cloud environment where Firewall Rules are defined at subnet level and are not managed by Stack Manager.

Mediant CE deployment creates security groups that enable only relevant traffic for each component and subnet. These security rules are assigned to network interfaces on both signaling and media components.

Table 8-1: Inbound Rules for Default Security Groups

Component	Traffic	Subnet	Protocol	Port
Signaling Component (SC)	SSH	Main	TCP	22
	HTTP	Main	TCP	80
	HTTPS	Main	TCP	443
	SIP over UDP	▪ Main ▪ Signaling1 ▪ Signaling2	UDP	5060
	SIP over TCP/TLS	▪ Main ▪ Signaling1 ▪ Signaling2	TCP	5060, 5061
Media Component (MC)	RTP, RTCP	▪ Main ▪ Media1 ▪ Media2	UDP	6000-65535
All	Internal	Cluster	UDP	669, 680, 925, 3900
		Cluster	TCP	80, 2442, 224

Inbound security rules in the Main and Additional subnets are configured by default to accept all traffic, including management traffic, from all sources, which constitutes a significant security risk. It is highly recommended to modify them after Mediant CE creation to allow inbound traffic only from specific IP addresses / subnets, especially for management traffic.

Inbound security rules in the Cluster subnet are configured by default to accept traffic from the VMs that belong to the same security group / virtual network only. Therefore, there is no need to further adjust them.

Outbound security rules in all subnets are configured by default to allow all traffic. You may adjust them as per your needs.

For AWS environment, if you adjust the outbound rules for the Cluster subnet, make sure that they include the following minimal required rules:

Table 8-2: Minimal Required Outbound Rules for Cluster Security Group in AWS Environment

Type	Protocol	Port Range	Destination	Description
All	All	All	clusterSecurityGroup	Internal traffic between Mediant CE instances
HTTP	TCP	80	169.254.169.254/32	Communication with EC2 instance metadata service
HTTPS	TCP	443	A.B.C.D/32	Communication with EC2 API endpoint. Replace A.B.C.D with the actual IP address of the private EC2 endpoint in the Cluster subnet. If you use a NAT Gateway to access the public EC2 endpoint, replace the destination with 0.0.0.0/0.

8.2 Adjusting Security Groups

Default Security Groups described above may be modified during Mediant CE stack creation, by configuring the **Management ports** and **Signaling ports** configuration parameters. These parameters contain a comma-separated list of ports and corresponding transport protocols, for example, "22/tcp,80/tcp,443/tcp,161/udp".

If you need to adjust this configuration after the stack is created, for example, to allow signaling traffic on additional ports, use the *Modify* operation to change these configuration parameters and then *Update* to apply the changes.

For Azure and Google Cloud environments, the provided configuration affects not only on security groups, but also the corresponding Load Balancers.

9 Upgrading Software Version

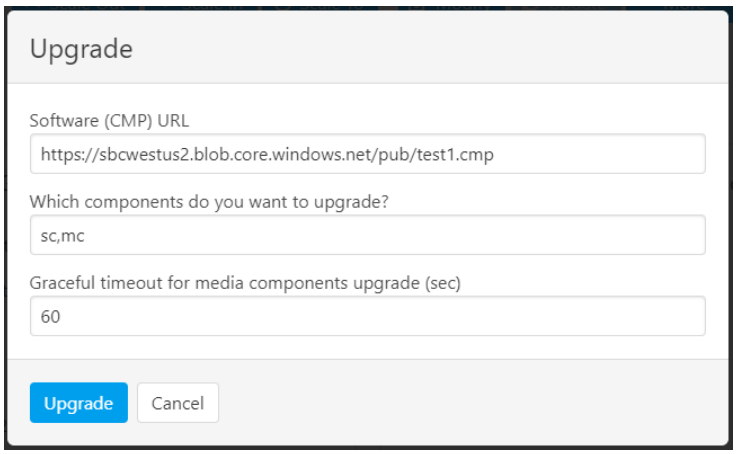
IMPORTANT NOTICE

For upgrading Mediant CE SBC to a version using a digitally signed .cmp file, you **must** follow the upgrade prerequisites and instructions in the document Mediant SW-90xx SBC Signed-CMP Upgrade Procedure Configuration Note.

You may upgrade the software version of the deployed Mediant CE using the Software Version file (.cmp) through one of the following means:

- Using Mediant CE Web interface:
  - Upgrade signaling components using the Software Upgrade Wizard (**Action > Software Upgrade**).
  - Upgrade "active" (currently running) media components using the Cluster Management page (**SETUP > IP NETWORK > MEDIA CLUSTER > Cluster Management**).
  - Upgrade "idle" (currently stopped) media components using Stack Manager (**Update Idle MCs**).
- Using Stack Manager's Web interface:
  - Upgrade all components at once using the **Upgrade** operation

Figure 9-1: Upgrading Mediant CE via Stack Manager



Upgrade using the Software Version file (.cmp) may be performed only within the same OS version stream. For example, if your Mediant CE is currently running software version 7.20A.256.396 (i.e., 7.20**A** stream / 'OS version': **6**), you may use 7.20A.258.010 .cmp file to upgrade it to a newer version (also 'OS version': **6**). However, you may not use 7.20CO.256.009 .cmp file to perform a similar upgrade to a version of the 7.20**CO** stream ('OS version': **8**).

Note: The 7.20CO stream ('OS version': **8**) is currently available for Azure and AWS environments only.

If you want to upgrade Mediant CE deployed with a version from 7.20A stream ('OS version': **6**) to a version from 7.20CO stream ('OS version': **8**), use one of the following methods:

- Method 1 – deploy a new Mediant CE instance ('OS version': **8**). Configure it and switch live traffic to the new instance. See Section 9.1 for detailed instructions.
- Method 2 – rebuild an existing Mediant CE instance from the new CentOS 8 image. See Section 9.2 for detailed instructions.

Advantages and disadvantages of each method are listed in the table below:

Method	Advantages	Disadvantages
Method 1	▪ In case of any problems with the new software version ('OS version': **8**), live traffic may be switched back to the old instance ('OS version': **6**). ▪ Traffic may be gradually moved to a new instance (assuming VoIP equipment that sent traffic towards the Mediant CE supports such functionality), thereby providing better control over the upgrade process and minimizing service downtime.	▪ Requires the use of additional resources for the duration of the upgrade. ▪ Implies a change of IP addresses (both public and private) and therefore, requires re-configuration of VoIP equipment that communicates with the Mediant CE. ▪ Requires a new License Key for the new Mediant CE instance.
Method 2	▪ Doesn't require additional resources. ▪ Preserves public and private IP addresses of the deployed CE instance.	▪ Requires a new License Key after the upgrade (because SC's serial number changes). ▪ Service is unavailable while instances are rebuilt (typically for 10-15 minutes).

9.1 Method 1 – Side-By-Side Deployment of New Version

This chapter describes the upgrade of a Mediant CE instance running a software version from the 7.20A stream ('OS version': **6**) to a version from the 7.20CO stream ('OS version': **8**), by side-by-side installation of a new Mediant CE instance and gradual migration of live traffic from old to the new instance.

➢ **To perform upgrade via "side-by-side deployment" method:**

1. Deploy a new Mediant CE instance using Stack Manager, as described in Section 7.1. Choose **OS Version = 8** during the deployment. Connect the new Mediant CE instance to the same Virtual Network and Subnets as the existing Mediant CE instance.
2. Download the configuration (INI) file from the existing Mediant CE instance: **Actions > Configuration File > Save INI File**.
3. Remove all networking configuration from the downloaded file, by doing one of the following:
   - Manually: Open the file in a text editor (e.g. Notepad++), and then delete the following elements:
     - Configuration tables: PhysicalPortsTable, EtherGroupTable, DeviceTable, InterfaceTable, MtcEntities
     - Configuration parameters: HARemoteAddress, HAUnitIdName, HARemoteUnitIdName, HAPriority, HARemotePriority, HALocalMAC, HARemoteMAC
   - Using the ini_cleanup.py script from the *Mediant VE Installation Kit* available on www.audiocodes.com portal:

     ```
     # python ini_cleanup.py old.ini new.ini
     ```

4. Load the "cleaned up" configuration file to the new Mediant CE instance as an incremental INI file: **SETUP > ADMINISTRATION > MAINTENANCE > Auxiliary Files > INI file (incremental)**.
5. Obtain, activate and apply the license to the new Mediant CE instance, as described in Section 10.
6. Switch live traffic from the old Mediant CE instance to the new one. This typically requires a change in the SBC IP address in the VoIP equipment that communicates with the Mediant CE. Consider performing gradual traffic migration if your VoIP equipment supports it. For example, first switch 10% of your live traffic to the new Mediant CE instance, verify that it is processed as expected, and only after that switch the rest of the traffic.
7. After all live traffic is switched to the new Mediant CE instance and service operates normally, delete the old Mediant CE instance.

9.2 Method 2 – Rebuild Existing Mediant CE Instance from New Image

This chapter describes the upgrade of a Mediant CE instance running a software version from the 7.20A stream ('OS version': **6**) to a version from the 7.20CO stream ('OS version': **8**), by rebuilding an existing Mediant CE instance from a new image.

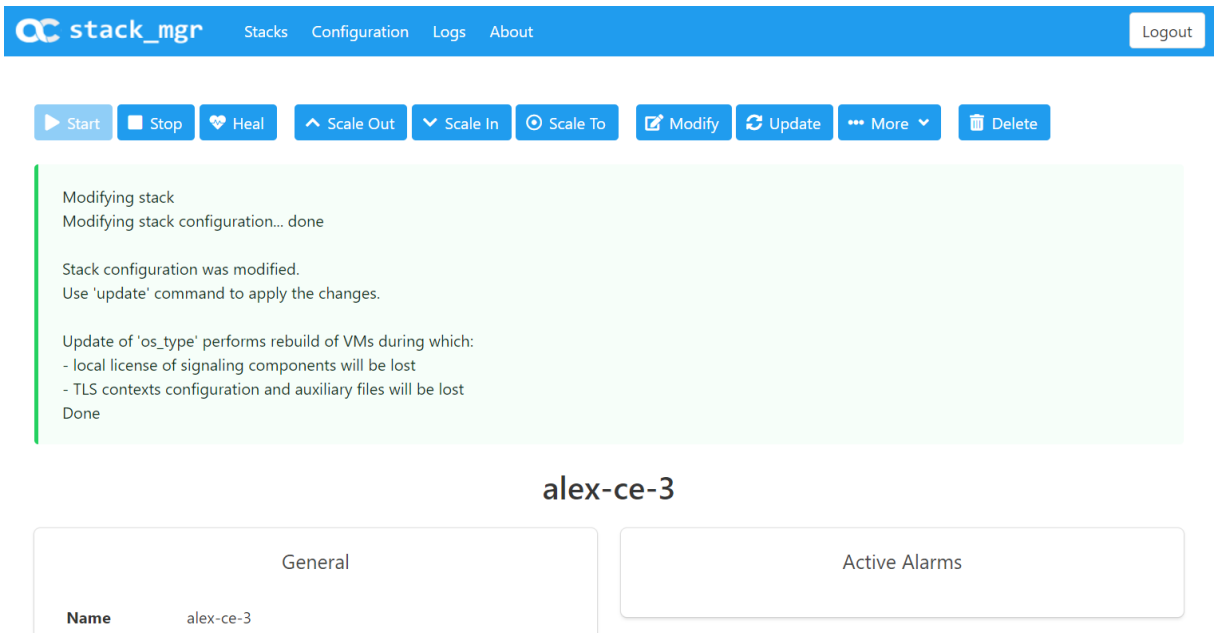
The described procedure preserves all IP addresses (private and public) assigned to the Mediant CE instance, as well as most of the SBC configuration. However, the following configuration elements will be lost and must be manually restored after the procedure:

- TLS Contexts configuration (certificates and private keys)
- Auxiliary files (e.g., Pre-recorded Tone files)
- License keys (as the serial number of rebuilt instances changes)

➢ **To perform upgrade via "rebuild from a new image" method:**

1. Connect to the Stack Manager Web interface.
2. Click the corresponding stack name.
3. Click **Modify,** and then change the **OS Version** to **8**.
4. Click **Update** to rebuild the stack.
5. Wait for the **Update** operation to complete. The operation typically takes 10-15 minutes, during which all VM instances are rebuilt and service is unavailable. Mediant CE configuration, including private and public IP addresses is preserved.
6. Restore parts of the SBC configuration that have been lost during the rebuild (i.e., TLS certificates, private keys and auxiliary files).
7. Obtain, activate and apply the license to the signaling components, as described in Section 10. Your Mediant CE is now running CentOS 8 based load and is fully operational.

Figure 9-2: Upgrading Mediant CE to New Image Based on CentOS 8



10 Licensing Mediant CE

Once you have successfully installed Mediant CE, you need to obtain, activate and then install the License Key.

Note: Licensing is applicable only to SCs; MCs do not require licensing.

10.1 Obtaining and Activating a Purchased License Key

For Mediant CE to provide you with all the required capacity and features, you need to obtain and activate a License Key which enables these capabilities.

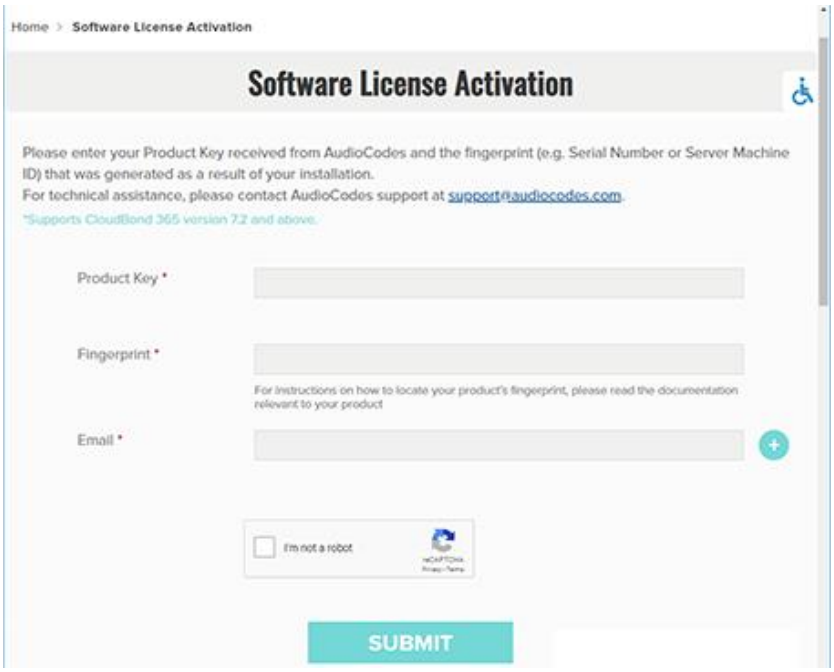
Note:

- License activation is intended **only** for first-time software activation upon product purchase (or if your License Key is "lost", due to whatever reason). For subsequent software feature upgrades, the License Key file is e-mailed to you after your Purchase Order has been processed.
- For Mediant CE with two SC instances, each SC instance has its own Serial Number, Product Key and License Key. Therefore, the instructions in this section must be done per SC instance.

➢ **To obtain and activate the License Key:**

1. Open AudioCodes Web-based Software License Activation tool at https://www.audiocodes.com/swactivation:

Figure 10-1: Software License Activation Tool



2. Enter the following information:
   - **Product Key:** The Product Key identifies your specific Mediant CE purchase for the purpose of subsequent communication with AudioCodes (for example, for support and software upgrades). The Product Key is provided in the Order Confirmation e-mail sent to you by AudioCodes upon your purchase, as shown in the example below:

Figure 10-2: Product Key in Order Confirmation E-mail

Dear Customer,

Customer PO# 12345
Order # 123456 , Line # 1
Ordered CPN: MSW/HIGH

Product Key Details:
Please note that products with same redundant pair should be installed together as redundant key.

Application	Product Key	Redundant Pair
Embedded (S/W SBC)	LC376CAD7FF01WR3	
Embedded (S/W SBC)	YDABF41BFFF01AY7	

Note: For Mediant CE orders with two SC instances, you are provided with two Product Keys, one for each SC instance. In such cases, you need to perform license activation twice to obtain License Keys for both SC instances.

   - **Fingerprint:** The fingerprint is the Mediant CE's Serial Number. The Serial Number uniquely identifies the software installation. The Serial Number is displayed in the 'Serial Number' field on the Device Information page (**Monitor** menu > **Monitor** menu > **Summary** tab > **Device Information**).
   - **Email:** Provide one or more e-mail addresses to where you want the License Key to be sent.
3. Click **Submit** to send your license activation request.
4. Once AudioCodes processes and completes your license activation, you will receive an e-mail notification with the License Key file attached. Open the file with any text-based program (such as Notepad) and make sure that the serial number ("**S/N**") in the License Key is correct and reflects the Serial Number of your SC instance.

Warning: Do not modify the contents of the License Key file.

10.2 Installing the License Key

For installing the License Key on Mediant CE, refer to the *Mediant Software SBC User's Manual*.

Note: The License Key file for Mediant CE with two SC instances must contain two License Keys - one for the active SC instance and one for the redundant SC instance. Each License Key has a different serial number ("S/N"), which reflects the serial number of each SC instance.

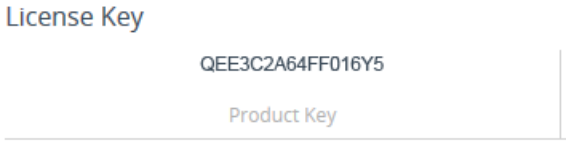
10.3 Product Key

The Product Key identifies a specific purchase of your Mediant CE deployment for the purpose of subsequent communication with AudioCodes (e.g., for support and software upgrades). The Product Key is provided in the order-confirmation email sent to you upon your product purchase and is used for activating your license through AudioCodes Software License Activation tool.

The Product Key is included in the License Key. Once the License Key is installed, you can view the Product Key in the following Web pages:

- License Key page (**Setup** menu > **Administration** tab > **Maintenance** folder > **License Key**). The Product Key is displayed in the read-only 'Product Key' field, as shown in the example below:

Figure 10-3: Viewing Product Key

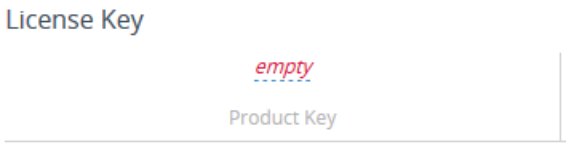


- Device Information page.

If your License Key was purchased in an earlier version (for example, 7.0), the 'Product Key' field may appear empty. In such a scenario, request the Product Key from your AudioCodes sales representative. Once received, do the following:

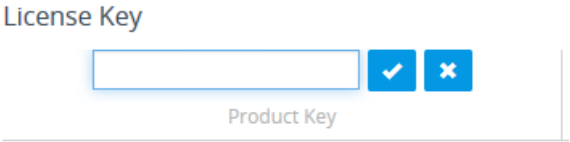
1. Open the License Key page.
2. Locate the Product Key group:

Figure 10-4: Empty Product Key Field



3. Click "empty"; the following appears:

Figure 10-5: Entering Product Key



4. In the field, enter the Product Key, and then click **Submit** (or **Cancel** to discard your entry).

International Headquarters
1 Hayarden Street,
Airport City
Lod 7019900, Israel
Tel: +972-3-976-4000
Fax: +972-3-976-4040

AudioCodes Inc.
200 Cottontail Lane,
Suite A101E,
Somerset, NJ 08873
Tel: +1-732-469-0880
Fax: +1-732-469-2298

Contact us: https://www.audiocodes.com/corporate/offices-worldwide
Website: https://www.audiocodes.com/

Document #: LTRT-10890