Installation Manual

AudioCodes Mediant™ Family of Session Border Controllers (SBC)

Mediant Cloud Edition (CE)

Deployment in Microsoft Azure

Version 7.6

SBC

QC audiocodes

Table of Contents

List of Figures

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.

Abbreviation	Description
MC	Media Component
SC	Signaling Component

Document Revision Record

LTRT	Description
11016	Initial document release for Version 7.6

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

Mediant Cloud Edition (CE) Session Border Controller (SBC), hereafter referred to as *Mediant CE*, is a software-based product that can be deployed in one of the following operational environments:

■   Amazon Web Services (AWS)

■   Microsoft Azure

■   Google Cloud

■   OpenStack

■   Non-cloud virtual environments (e.g. VMware)

This document describes deployment of Mediant CE in a Microsoft Azure environment.

For detailed instructions on Mediant CE installation in other operational environments (for example, VMware), refer to the dedicated installation manual.

Note:

- The scope of this document does not fully cover security aspects for deploying the product in the Azure cloud. Security measures should be done in accordance with Azure security policies and recommendations.

- For configuring Mediant CE SBC, refer to the *Mediant Software SBC User's Manual*.

- Mediant VE and CE products share the same software image published by AudioCodes in Azure Marketplace. Therefore, in some places in this document the Mediant VE product name is referenced even though the document concerns Mediant CE.

1.1 Architecture Overview

Figure 1-1: Mediant CE Architecture
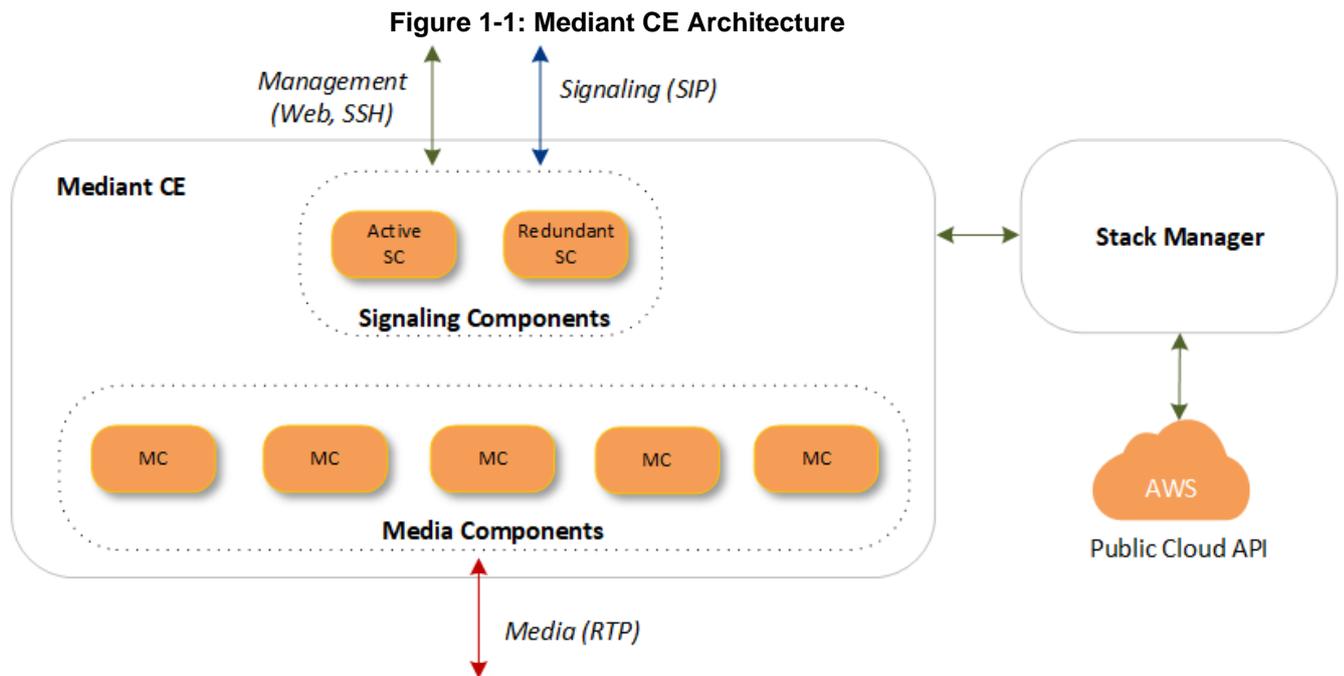
The Mediant CE cluster is comprised of multiple components (virtual machines) that perform distinct functions:

■ **Signaling Components:** Handle signaling (SIP) and management (Web, SSH, etc) 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:** Handle media (RTP, RTCP) traffic, including transcoding functionality. Up to 21 Media Components can be used in the deployed Mediant CE.

Incoming calls are initially processed (at signaling level) by Signaling Components, which choose Media Component based on the current cluster utilization and pass the media streams to it.

Signaling components also serve as a "single point of contact" for all management tasks. They provide Web and CLI interfaces through which customers have complete control over all cluster components.

1.2   Deployment Topology

In a typical Mediant CE deployment, two Signaling Components are created and operate in 1+1 Active / Standby mode. They are placed behind Azure Load Balancer to enable preservation of management and signaling IP addresses, as well as established calls, during switchover.

Mediant CE cluster may contain up to 21 Media Components that operate in N+1 Load Sharing mode. In case of a specific Media Component failure, calls handled by it are re-distributed across remaining Media Components, with no visible effect on established calls.

Two deployment topologies are supported:

■   **Availability Sets:**

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.

Figure 1-2: Mediant CE Deployment via Availability Sets
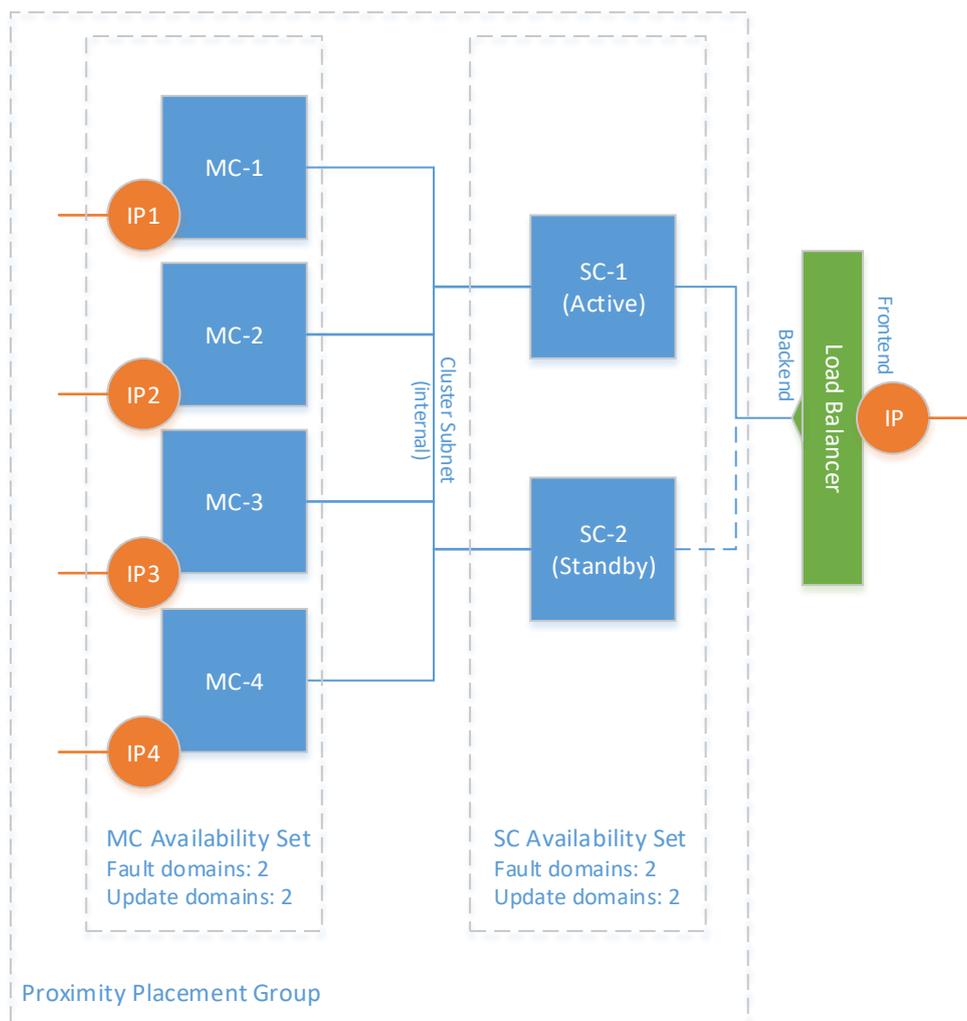
■ **Availability Zones:**

Mediant CE components are distributed across two Availability Zones. This topology provides higher SLA. However, this may suffer from intermittent network latency between zones, which may affect internal communication between Mediant CE components and cause Signaling Component/Media Component switchovers.

Figure 1-3: Mediant CE Deployment via Availability Zones
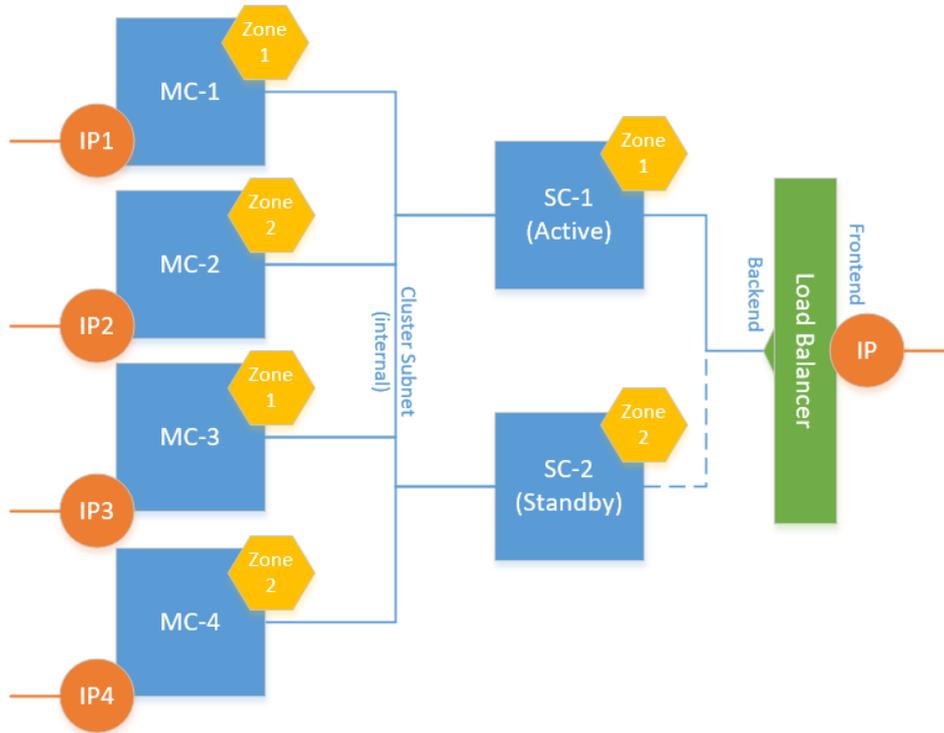
You can adjust cluster size by scaling Media Components "in" or "out" based on cluster utilization and/or explicit customer request. "Scaled down" media components are kept in "deallocated" state, which ensures that they can be quickly started during "scale out" operation.

1.3   Azure Load Balancer

Azure Load Balancer is used to steer inbound (signaling and management) traffic towards the active signaling component. 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). "Rport" extension, as per RFC3581, is used to make request/response flows symmetric.

■ When private IP addresses are used, outbound traffic does not traverse 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 Signaling Component instance.

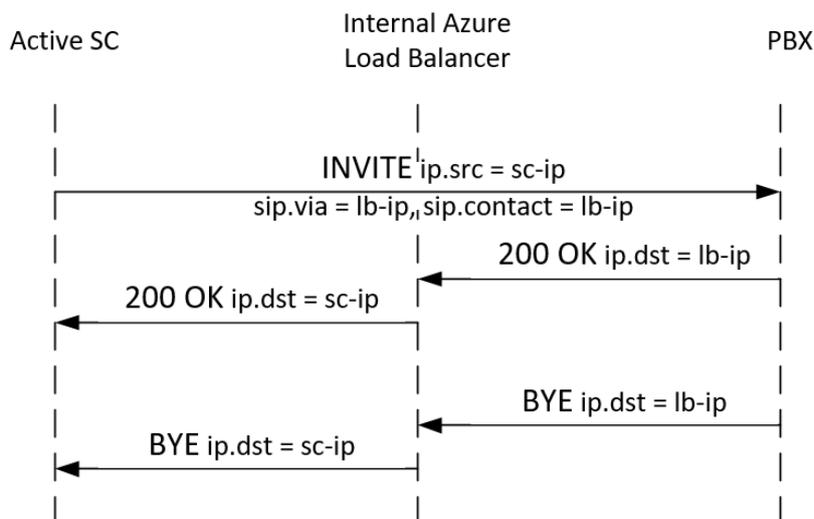> **Note:** Outbound traffic may sometimes appear with the source IP address of the Internal Load Balancer at the IP layer as well. This may happen if the outbound flow occurs shortly after the inbound flow and is attributed to the existing DNAT translation in the Internal Load Balancer. Therefore, VoIP 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 Signaling Component 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.

1.4 Stack Manager

The Stack Manager tool is provided as part of the solution. It is used for initial Mediant CE cluster deployment and complete lifecycle management; for example update of network topology, rebuild of cluster components in case of underlying cloud resources corruption or accidental removal etc.

Stack Manager also supports automatic scaling of Media Components based on cluster utilization, thus significantly reducing associated infrastructure costs.

1.5 Deployment Methods

Starting with Version 7.6, Mediant VE software is available in the following deployment modes:

- **Virtual Appliance:** A fully integrated virtual machine image that includes the Mediant VE software along with a fully secured and highly optimized Linux kernel. This mode does not provide access to the underlying Linux OS and does not support the installation of additional software or third-party tools on the same virtual machine.

- **Container Image:** Deployed on a standard Linux OS host and run by a Docker engine. This mode allows for the installation of additional software and third-party tools on the same virtual machine. However, it may offer slightly lower performance compared to the Virtual Appliance due to the absence of kernel-level optimizations

The rest of this document by default refers to deployment of Mediant CE software as a **virtual appliance**. If you want to use **container images** instead, refer to the dedicated chapters and notes.

For generic information about the **container image** deployment mode, refer to *Mediant VE Container Installation Manual*.

2   Installation Prerequisites

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 Azure Marketplace. For more information, see Subscribing to Mediant VE Offer in Azure Marketplace.

■ You have created all subnets needed for Mediant CE deployment, including the Cluster subnet. For more information, see Network Prerequisites.

2.1   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.

> **Notes:**
>
> • 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)".
>
> • Skip this section if you want to deploy Mediant CE software as a **container image**.

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 2-1: Azure Marketplace
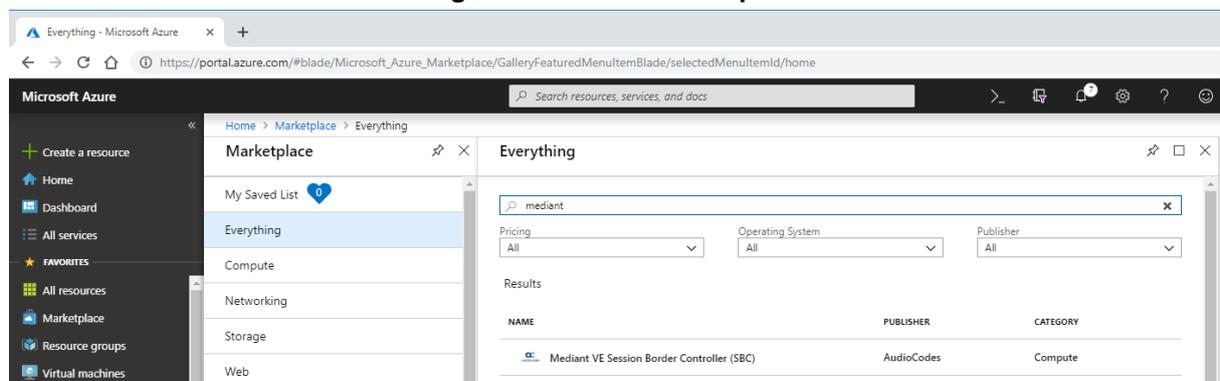
4. Click the **Mediant VE Session Border Controller (SBC)** product; the Mediant VE Product overview screen appears.

Figure 2-2: Mediant VE SBC Product Overview
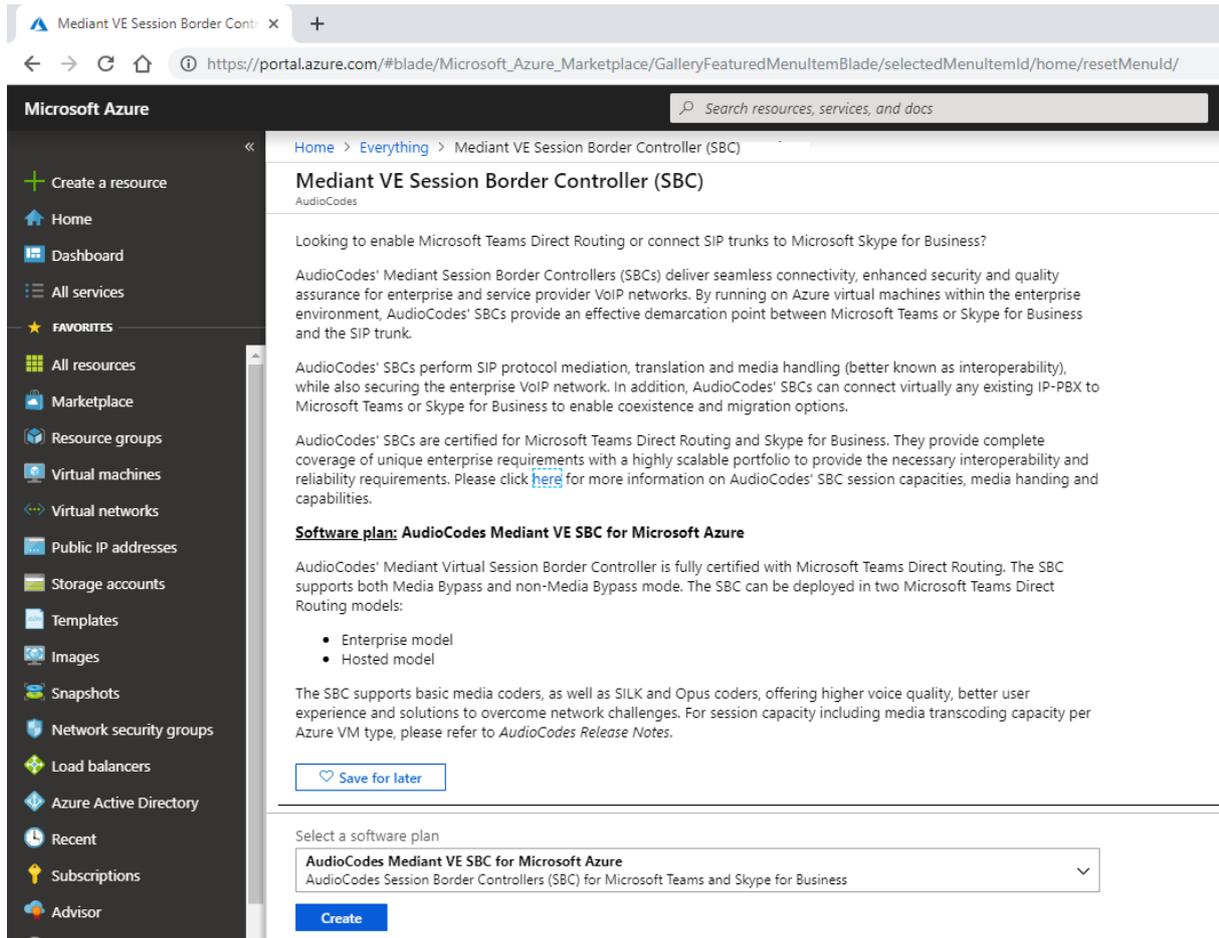
5. Click **Create** to start a new Mediant VE deployment; the Create AudioCodes Mediant VE SBC for Microsoft Azure dialog box appears. The dialog box contains multiple steps. Complete each step according to the description below.

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

Figure 2-3: 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', click **Create new**, 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 2-4: Buy Step
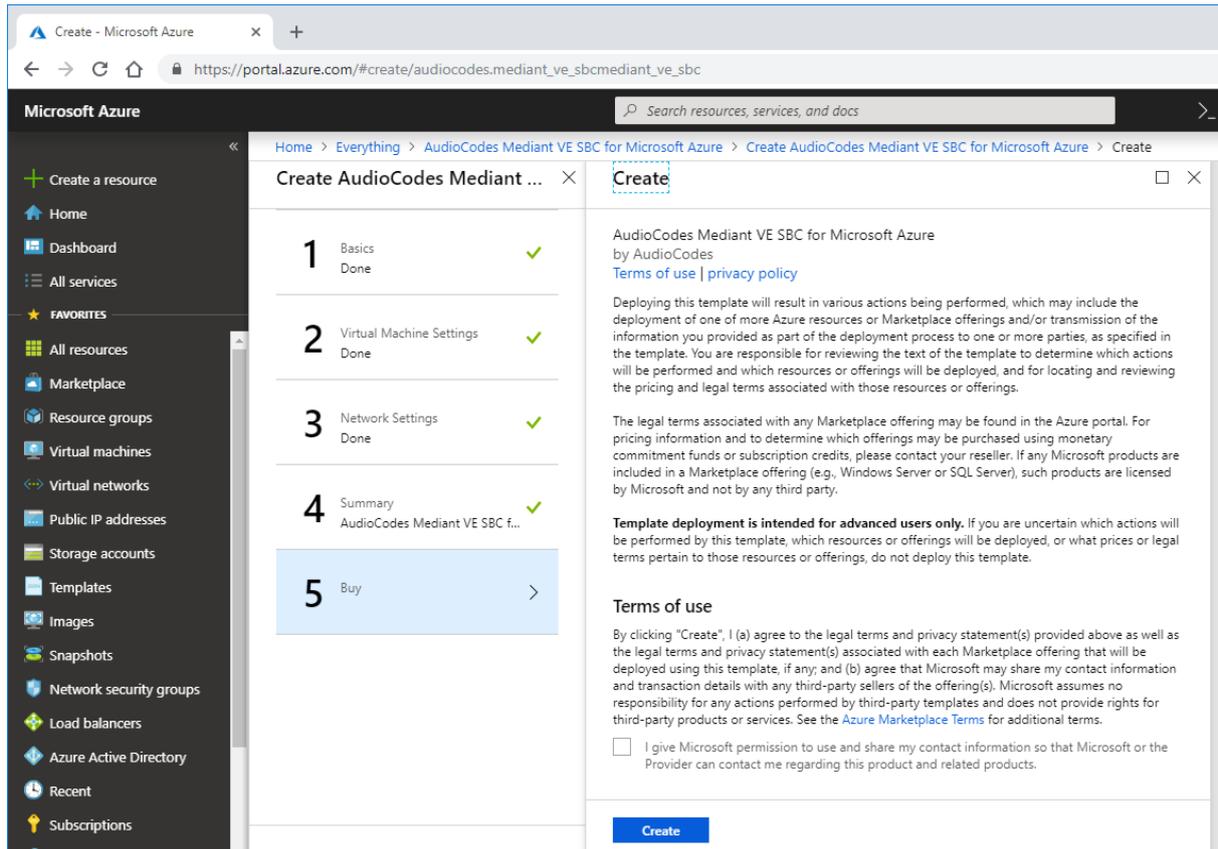
9. Wait until the virtual machine deployment is complete

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.

2.2   Network Prerequisites

Mediant CE on Microsoft Azure uses the following network architecture:

Figure 2-5: Mediant CE Network Architecture – Azure
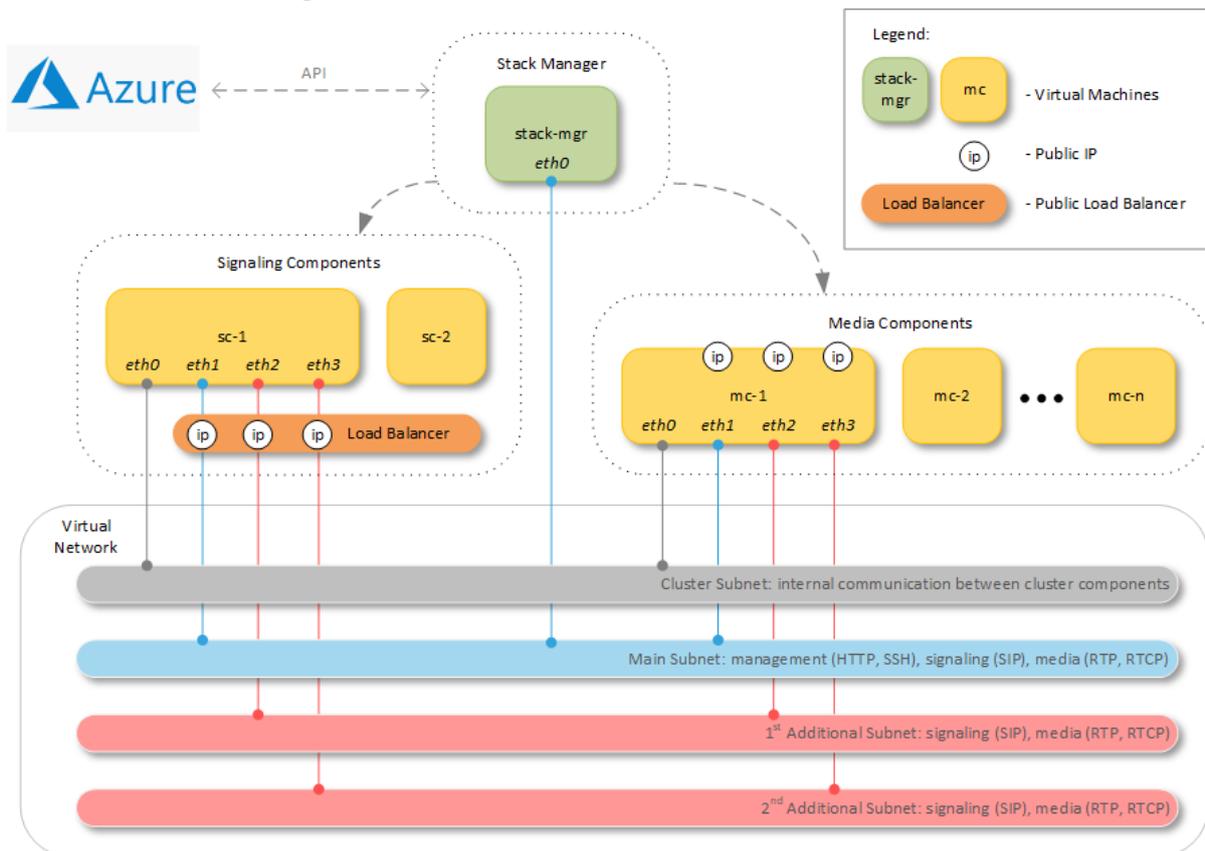
Up to eight subnets may be used:

■ **Cluster Subnet:** Carries internal communication between Mediant CE components. It's connected to both Signaling Component and Media Component instances as the first network interface (eth0).

■ **Main Subnet:** Carries management (HTTP, SSH, etc.), signaling (SIP) and media (RTP, RTCP) traffic. It's connected to both Signaling Component and Media Component instances as the second network interface (eth1) and to the Stack Manager instance.

■ **1st, 2nd, etc. Additional Subnets:** Carry signaling (SIP) and media (RTP, RTCP) traffic. It's connected to Media Component instances as the third (eth2), fourth (eth3), etc. network interfaces, correspondingly. These subnets are optional because the Main Subnet may carry all types of traffic.

All subnets must reside in the same Virtual Network and be created prior to Mediant CE deployment.

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

2.3 Virtual Machine Sizes

The default Mediant CE deployment uses the following instance types:

- **Signaling Component instances:** Standard_D8ds_v5
- **Media Component instances:** Standard_D2ds_v5 (for two network interfaces) or Standard_D8ds_v5 (for three or four network interfaces)

You may customize instance types during stack creation.

Refer to the *Release Notes* for a complete list of instance types supported by Mediant CE, their capacities and capabilities

3   Deploying Mediant CE

Deployment of Mediant CE on Azure platform is performed via Stack Manager.

Stack Manager is a management tool developed by AudioCodes that enables simple and intuitive deployment and complete lifecycle management of Mediant VE and Mediant CE products on public clouds. The tool provides the following features for Mediant CE:

■ Initial product deployment.

■ Update of deployed stack's network topology

■ Automatic and on-demand scaling of media components, to adjust stack footprint and minimize infrastructure costs.

■ Monitoring of deployed Azure resources and recovery in case of their corruption / accidental removal

■ Upgrade of software on all Mediant CE components

■ Removal of all deployed resources in case of stack deletion

Stack Manager uses dynamically generated Azure Resource Management (ARM) templates for stack deployment on Azure platform and is not involved in call processing or any other services provided by Mediant CE.

> **Note:** The following instructions describe the installation of Mediant CE as a **virtual appliance**. If you want to install Mediant CE as a **container image**, see Section 3.1, Deploying Mediant CE as Container Image.

➢ **To deploy Mediant CE:**

1. Install the Stack Manager tool, as described in the *Stack Manager User's Manual.*

2. Log into the Stack Manager tool after the deployment; the following screen appears:

Figure 3-1: Stack Manager Main Screen

3. Click **Create** to create a new stack; the following dialog box appears:

Figure 3-2: Create Stack Dialog – Step 1

4. In the 'Name' field, enter a name for the stack (e.g., "mediant-ce").

5. From the 'Stack type' drop-down list, select **Mediant CE**.

6. From the 'Region' drop-down list, select a region where the stack will be deployed; additional fields appear:

Figure 3-3: Create Stack Dialog – Step 2

7. From the 'Deployment topology' drop-down list, select the topology **availability set** or **availability zones.** If you select **availability zones**, you are prompted to specify the zones, whose value is a comma-separated list of two zone numbers (e.g., "1,2").

8. From the 'Virtual network' drop-down list, select the virtual network where the stack will be deployed; additional fields appear:

Figure 3-4: Create Stack Dialog – Step 3

9. Select the subnets that Mediant CE will be connected to.

10. From the 'Public Ips' drop-down list, select which subnets need to communicate with external equipment via public IP addresses. Based on the selected value, Stack Manager places corresponding signaling component's network interfaces behind Public or Internal Load Balancer and assigns public IP addresses to the media components.

11. If you assign a Public IP address to the Main subnet, Stack Manager by default configures the corresponding Public Load Balancer's frontend IP address as Mediant CE's management IP address and uses it for communicating with the deployed stack. You may override this behaviour, by checking the 'Use private IP address for management' checkbox. In this case, Stack Manager creates additional (secondary) IP address on the signaling component's second network interface (eth1), attached to the Main subnet, places it behind the Internal Load Balancer, configures this Internal Load Balancer's frontend IP address as Mediant CE's management IP address, and uses it to communicate with the deployed stack.

Figure 3-5: Create Stack Dialog – Step 4

12. The virtual machine type for both signaling and media components is pre-selected and automatically updated based on other parameters that you define in **Create Stack** dialog box. If you want to modify it, check the 'Customize' checkbox next to it and then select a value from the 'VM type' drop-down list.

13. From the 'Profile' drop-down list, select whether you need media components to perform simple media stream **forwarding** (includes RTP-to-SRTP translation and vice versa) or **transcoding** capabilities (for coder conversion or DTMF detection).

14. From the 'Min number' and 'Max number' drop-down lists, select the minimum and maximum number of media components in the stack. Stack Manager creates the maximum number of media components that you selected, but initially starts only with creating the minimum number that you selected. You may later adjust the number of running media components via **scale out** and **scale in** actions.

15. In the 'Username' and 'Password' fields, enter the admin user credentials that will be configured on the deployed stack. You use these credentials when connecting to the stack via Web or CLI management interfaces. Note that Stack Manager uses different credentials to communicate with the stack – **StackMgr** user and randomly generated password. Therefore, even if you later change admin user credentials (e.g., via Mediant CE's Web or CLI interface) communication between Stack Manager and the deployed Mediant CE stack is not affected.

Figure 3-6: Create Stack Dialog – Step 5

16. From the 'SBC version' drop-down list, select the Mediant CE version that you want to deploy.

17. In the 'Management ports' and 'Signaling ports' field, enter a list of management and signaling ports respectively that should be open on Mediant CE. Specified ports are configured in the corresponding network security groups assigned to Mediant CE network interfaces, and corresponding rules are created in Azure Load Balancer. The value is a comma-separated list of the following elements:

    • <port>/udp: Opens a specific UDP port for all sources (e.g., 161/udp)

    • <port>/udp/<cidr>: Opens aspecific UDP port for traffic originating from aspecific CIDR (e.g., 161/udp/172.16.0.0/16 opens UDP port 161 for traffic from 172.16.0.0/16 subnet)

    • <port>/tcp: Opens aspecific TCP port (e.g., 22/tcp)

    • <port>/tcp/<cidr>: Opens aspecific TCP port for traffic originating from aspecific CIDR (e.g., 22/tcp/172.16.1.0/24)

18. From the 'Use main subnet for' drop-down list, select whether the Main subnet should be used for all traffic (management, signaling and media) or for management traffic only. The selection effects network security groups assigned to the Mediant CE network interface connected to the Main subnet and default SIP Interface and Media Realm created on Mediant CE.

19. In the 'Advanced config' text box, enter advanced configuration parameters, if needed. See the next sections for a partial list of supported advanced configuration parameters. Refer to *Stack Manager User's Manual* for a complete list.

20. Click **Create** to start stack creation.

21. Wait until stack is created.

Figure 3-7: Successful Stack Creation

3.1   Deploying Mediant CE as Container Image

Starting from Version 7.6, you have an option to deploy Mediant CE software in Azure public cloud as a **container image**. In this mode, Mediant CE software is run on a standard Linux host using a Docker engine.

Mediant VE and CE share the same container image, which is delivered as a **docker_debian_sip_F7.60A.xxx.yyy.tar.gz** archive that can be downloaded from AudioCodes Services Portal - Mediant Virtual Edition (VE), Mediant Cloud Edition (CE) and Mediant Server Edition (SE) Session Border Controllers (SBC).

You need Stack Manager Version 3.8.5 or later to deploy Mediant CE as a container image.

You also need to enable the **docker deployment mode**:

1.   Log in to Stack Manager.

2.   On the menu bar, click **Configuration**; the Configuration screen appears.

3.   Under the **Advanced** group, set 'Docker Deployment Mode' to **Enabled**.

4.   Click **Update** to apply the new configuration.

➢ **To deploy a Mediant CE as container image:**

1.   Log in to Stack Manager.

Figure 3-8: Stack Manager Tool
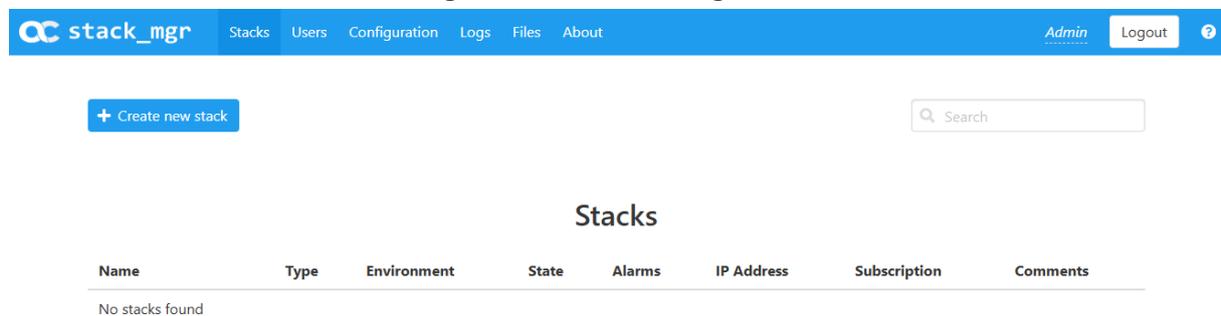
2.   On the menu bar, click **Files**; the Files screen appears.

Figure 3-9: Stack Manager: Files Screen
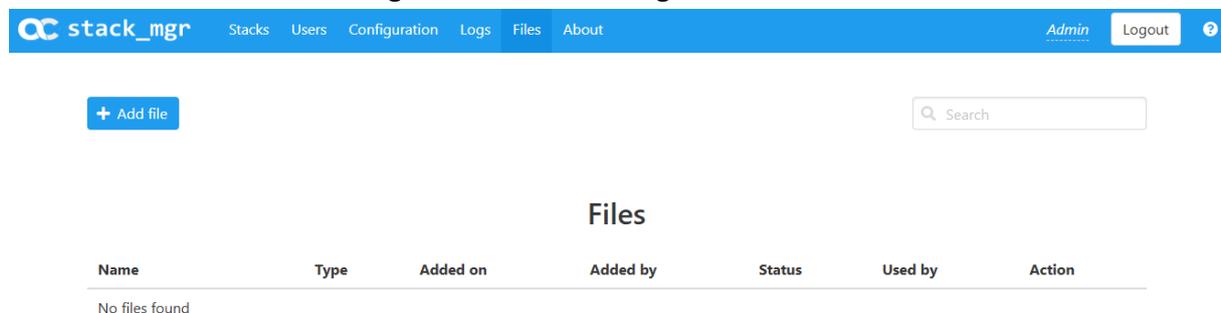
3. Click **Add file**; the Add file dialog box appears.

Figure 3-10: Stack Manager: Add File Dialog

Add file

Add firmware (.cmp) or docker archive (.tar.gz) file

[   ] [Choose file...]

Note that added firmware (.cmp) files will be publicly accessible via files/<filename> URL.

[Add] [Cancel]
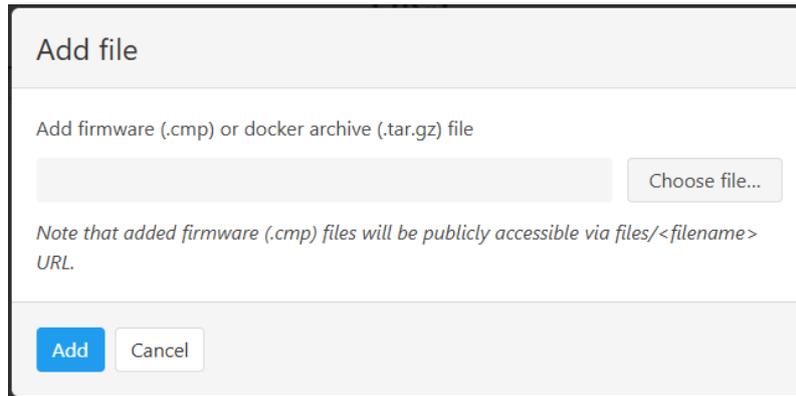
4. Choose the Mediant VE/CE container image file, and then click **Add**; the file is uploaded to Stack Manager and appears in the **Files** list.

Figure 3-11: Stack Manager: File Added

![stack_mgr interface]

| Stacks | Users | Configuration | Logs | Files | About | Admin | Logout |

+ Add file   [Search]

Adding file 'docker_debian_sip_F7.60A.092.734.tar.gz'
Uploading file... done
Done

Files

Name	Type	Added on	Added by	Status	Used by	Action
docker_debian_sip_F7.60A.092.734.tar.gz	docker	Mar 30, 2025 10:06:48	Admin	ok		Delete
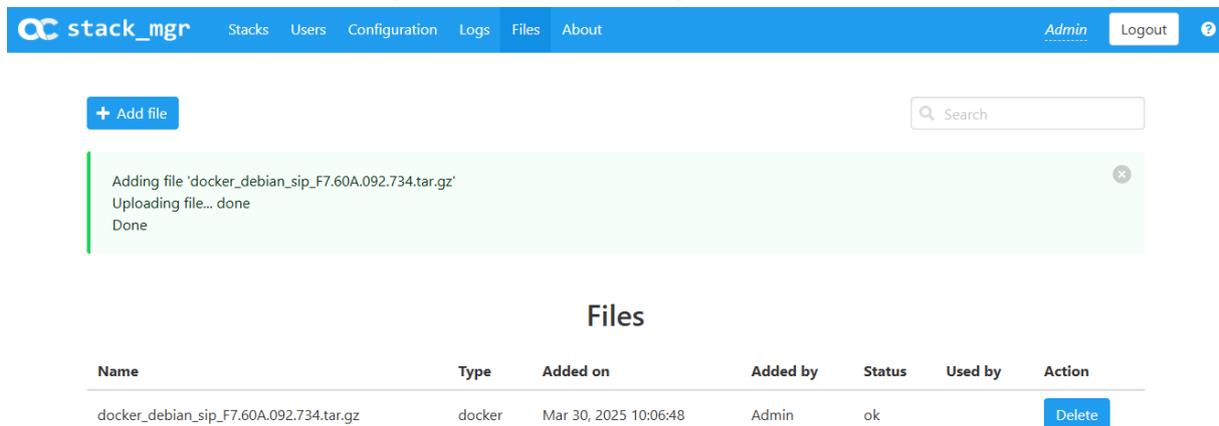
5. On the menu bar, click **Stacks**; the Stacks screen appears.

6. Proceed with Mediant VE installation, as described in Section **Error! Reference source not found.**3, Deploying Mediant CE with the following changes:

- In stack's **Advanced** group, from the 'Deployment mode' drop-down list, select **Docker**.

- Select the **container image** file that you uploaded in Step 4.

3.2   Public IP Addresses

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

By default, Stack Manager applies the same configuration for both signaling and media components. For each subnet that is configured to use a Public IP address, the following is created:

- **For signaling components:**
  - 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 Signaling Component 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)

- **For media components:**
  - Public IP address on the corresponding Media Component network interface
  - Corresponding entry in the NAT Translation SBC configuration table, which implements SNAT translation for outbound traffic at the application level (SIP and SDP)

You can specify different configuration for signaling and media components. You can also attach multiple public IP addresses to the same network interface. This may be done by configuring the **sc_public_ips** and **mc_public_ips** advanced configuration parameter in the **Advanced Config** section.

> **Note:** When the **sc_public_ips** or **mc_public_ips** advanced configuration parameter is specified in the **Advanced Config** section, it overrides any value configured via the **Public IPs** parameter in the **Networking** section for the corresponding components (Signaling Component or Media Component).

- **sc_public_ips**

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

  For example, below configuration 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):
  ```
  sc_public_ips = main:2,additional1
  ```

- **mc_public_ips**

  Same as above, but for Media Component network interfaces.
  For example:
  ```
  mc_public_ips = main,additional1:2
  ```

When the **sc_public_ips** or **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 attachment. The exact behavior depends on the component type:

■ **For Signaling Components:** Public IP addresses are always attached to the Public Azure Load Balancer and "mapped" to the corresponding private IP addresses. The 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.

■ **For Media Components:** 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.

3.3   Private IP Addresses

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

- **For signaling components:**
    - Front-end rule on Azure Internal Load Balancer
    - Forwarding rule on Azure Internal Load Balancer, which implements forwarding of incoming traffic towards the active Signaling Component instance
    - Corresponding entry in the NAT Translation SBC configuration table, which implements SNAT translation for outbound traffic at application levels (SIP and SDP)
- **For media components:**
    - Private IP addresses on corresponding network interfaces are used

If you want to enable communication via both public 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** or **mc_additional_ips** advanced configuration parameters in the **Advanced Config** section.

- **sc_additional_ips**

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

    For example, below configuration 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):

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

- **mc_additional_ips**

    Same as above, but for Media Component network interfaces.

    For example:

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

The number of additional private IP addresses specified via the **sc_additional_ips** or **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 addresses (for internal communication between Signaling Component instances) and one secondary IP address (for internal communication with Media Component instances)

- **eth1** – one primary and two secondary IP addresses:
  - primary IP address - created implicitly, placed behind Public Azure Load Balancer (due to the **Public IPs** configuration parameter)
  - 1st secondary IP address – created due to the **sc_additional_ips** advanced configuration parameter containing "main" element, placed behind Internal Azure Load Balancer
- **eth2** – one primary and two secondary IP addresses:
  - 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

3.4   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. Therefore, 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. Therefore, Mediant CE management should be performed via the corresponding Load Balancer's internal IP address.

You may use the 'Use private IP address for management' parameter during Mediant CE creation to create both private and public IP addresses on the main subnet (placed behind the Internal and Public Load Balancers respectively) and use private IP address for management. The same may be achieved by configuring the **oam_ip** advanced configuration parameter after stack creation:

```
oam_ip = internal
```

The above configuration creates two IP addresses on the signaling component's "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)

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.

3.5   Security Groups

3.5.1   Default Security Groups

Stack Manager creates the following security groups during Mediant CE deployment:

- **Main** – security group for the "Main" subnet on Signaling Components
- **Signaling** – security group for the "Additional 1", "Additional 2", etc. subnets on Signaling Components
- **Media** – security group for the "Main", "Additional 1", "Additional 2", etc. subnets on Media Components
- **Cluster** – security group for internal traffic between Mediant CE instances

These default security groups are assigned to the following components and network interfaces:

Table 3-12: Assignment of Default Security Groups

Security Group	Components	Interface Name
Cluster	Signaling Components, Media Components	eth0
Main	Signaling Components	eth1
Signaling	Signaling Components	eth2, eth3, ...
Media	Media Components	eth1, eth2, eth3, ...

The following inbound rules are created in the default security groups:

Table 3-13: Inbound Rules for Default Security Groups

Security Group	Traffic	Protocol	Port	Source
Main	SSH	TCP	22	0.0.0.0/0
	HTTP	TCP	80	0.0.0.0/0
	HTTPS	TCP	443	0.0.0.0/0
	SIP over UDP	UDP	5060	0.0.0.0/0
	SIP over TCP	TCP	5060	0.0.0.0/0
	SIP over TLS	TLS	5061	0.0.0.0/0
	Media	UDP	6000-65535	0.0.0.0/0
Signaling	SIP over UDP	UDP	5060	0.0.0.0/0
	SIP over TCP	TCP	5060	0.0.0.0/0
	SIP over TLS	TCP	5061	0.0.0.0/0
Media	RTP, RTCP	UDP	6000-65535	0.0.0.0/0
Cluster	Internal	UDP	669	VirtualNetwork
	Internal	UDP	680	VirtualNetwork

Security Group	Traffic	Protocol	Port	Source
	Internal	TCP	80	VirtualNetwork
	Internal	TCP	2442	VirtualNetwork
	Internal	TCP	2424	VirtualNetwork
	Internal	UDP	3900	VirtualNetwork
	Internal	UDP	925	VirtualNetwork

Outbound rules are configured by default to allow all traffic.

3.5.2 Adjusting Default Security Groups

The default **OAM**, **Signaling** and **Media** security groups are configured by default to accept traffic from all sources, which constitutes a significant security risk. It is highly recommended to modify them after Mediant VE creation to allow inbound traffic only from specific IP addresses and/or subnets, especially for management traffic.

Note that inbound rules of the **Cluster** security group allow only traffic that originates from instances that reside in the same Virtual Network. Therefore, there is typically no need to modify them.

Such modification can be done via the following stack configuration parameters:

■ **Management Ports**

Defines a list of inbound management ports and corresponding transport protocols as provided by the **Main** security group.

The value is a comma-separated list of the following elements:

```
<port>/<protocol>/[<cidr>]
```

Where:

* <port> is an individual port number (e.g., 22) or a port range (e.g., 22-23)
* <protocol> is tcp or udp
* <cidr> is optional and can be an IP address (e.g., 10.1.2.3) or a CIDR (e.g., 10.1.0.0/16)

For example:

```
22/tcp/10.11.2.0/24,80/tcp/10.11.2.34,443/tcp
```

■ **Signaling Ports**

Defines a list of inbound signaling ports and corresponding transport protocols as provided by the **Main** and **Signaling** security group.

■ **Media Ports**

Defines a list of inbound media ports and corresponding transport protocols as provided by the **Media** security group.

You can also update inbound and/or outbound rules of default security groups via Azure Portal or CLI interfaces. If you do, consider using the "keep-" prefix for inbound rules that you create. This ensures that these rules are preserved during "update" and "rebuild" operations via Stack Manager.

3.5.3   Using Custom Security Groups

Instead of modifying rules of default security groups created by Stack Manager, you can use custom security groups, for example, created by your IT department.

Such configuration can be done via the following stack advanced configuration parameters:

■ **cluster_nsg_id**

Defines a custom security group to be used instead of the default **Cluster** security group.

Syntax: <ResourceGroupName>/<NsgName>

For example:

```
cluster_nsg_id = rg1/cluster-nsg
```

■ **main_nsg_id**

Defines a custom security group to be used instead of the default **Main** security group.

■ **signaling_nsg_id**

Defines a custom security group to be used instead of the default **Signaling** security group.

■ **media_nsg_id**

Defines a custom security group to be used instead of the default **Media** security group.

Alternatively, you can assign custom network security groups to a specific interface of specific components via the following stack advanced configuration parameters:

■ **nsg_id_sc_ethX**

Defines a custom security group for a specific network interface on Signaling Components instead of default security groups.

For example:

```
nsg_id_sc_eth0 = rg1/cluster-nsg
nsg_id_sc_eth1 = rg1/main-nsg
```

■ **nsg_id_mc_ethX**

Defines a custom security group for a specific network interface on Media Components instead of default security groups.

For example:

```
nsg_id_mc_eth0 = rg1/cluster-nsg
nsg_id_mc_eth1 = rg1/media-nsg
```

3.6 Deployment Troubleshooting

Stack Manager uses dynamically generated Azure Resource Manager (ARM) templates to perform deployment on Azure platform.

If Mediant CE deployment fails and the error description provided by Stack Manager is not detailed enough, refer to the **Deployments** logs in corresponding Resource Group (same as deployed stack name) for additional information.

4   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.

> ⚠️ **Note:** This section is applicable to Mediant VE deployed as a **virtual appliance**. For deployments using **container images**, see Section 4.1, Upgrading Software Version for Container Deployments.

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 4-1: Upgrading Mediant CE via Stack Manager

> ⚠️ **Note:** Make sure that the Signaling Components have the same or later version than the Media Components.

Upgrade to 7.6 stream (7.60A.xxx.yyy versions) may be performed only from a 7.4 stream (7.40A.xxx.yyy versions).

> ⚠️ **Note:** If you have an earlier version installed (e.g., from 7.2 stream), first upgrade to the 7.4 stream; only afterwards can you upgrade to a 7.6 stream. Refer to 7.4 version documentation for detailed upgrade instructions.

4.1 Upgrading Software Version for Container Deployments

Mediant CE deployments that use **container images** don't support software upgrade using the firmware (.cmp) file. Instead, the Mediant VE/CE Container installation package (**docker_debian_sip_F7.60A.xxx.yyy.tar.gz**) is used.

For simplicity, the Stack Manager's Web interface is described in this section. However, you can perform the same task through the CLI and REST management interfaces. Refer to *Stack Manager User Manual* for details.

1. Log into the Stack Manager's Web interface.
2. Upload the new Mediant VE Container image in the **Files** screen.
3. In the **Stacks** screen, click your stack.
4. Click **Upgrade**; the upgrade dialog is displayed.

Figure 4-2: Stack Manager: Upgrade Dialog

5. Select the new Mediant VE/CE Container image (**Docker file**).
6. Leave the 'AudioCodes repository (ECR)' text box empty.
7. Select (check) the **Hitless upgrade** check box if you want to perform the upgrade without service interruption (HA deployments only). Otherwise, clear the check box if you want the upgrade process to be faster.
8. Click **Upgrade** to start the upgrade.

5 Downgrading Software Version

The procedure for downgrading Mediant CE software version is similar to the upgrading procedure, as described in the previous section, but in the reverse order:

■ You first need to downgrade the Media Components.

■ Afterwards, you need to downgrade the Signaling Components

This sequence ensures that the Signaling Components always have the same or later version than the Media Components.

When downgrading from version 7.40A.100.* or later to version 7.40A.005.*, the following additional configuration steps must be performed prior to the downgrade:

1. Connect to the Mediant CE's CLI interface (provided by Signaling Components) through an SSH client or a serial console.

2. Log in as an administrative user.

3. Run the following commands:

```
enable
    <password> (e.g. "Admin")
configure system
    voice-config
    TpncpEncryptionEnable = 0
    exit
exit
```

4. Reboot the Signaling Components using the `reload now` CLI command or the Web interface's **Reset** button**.**

5. Wait until the Media Components are connected. Verify that their displayed status is "Connected" and not "Connected (TLS)".

> **Note:** The above procedure is required because the communication protocol between the Signaling Components and Media Components was changed in version 7.40A.100.*. Failure to perform this procedure will prevent the Media Components from connecting to the Signaling Components after the latter are downgraded to the 7.40A.005.* version.

This page is intentionally left blank.

6 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 Signaling Components; Media Components do not require licensing.

6.1 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 Signaling Component instances, each Signaling Component instance has its own Serial Number, Product Key and License Key. Therefore, the instructions in this section must be done per Signaling Component instance.

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

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

Figure 6-1: 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 6-2: 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 Signaling Component instances, you are provided with two Product Keys, one for each Signaling Component instance. In such cases, you need to perform license activation twice to obtain License Keys for both Signaling Component 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 Signaling Component instance.

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

6.2   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 Signaling Component instances must contain two License Keys - one for the active Signaling Component instance and one for the redundant Signaling Component instance. Each License Key has a different serial number ("S/N"), which reflects the serial number of each Signaling Component instance.

6.3   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:
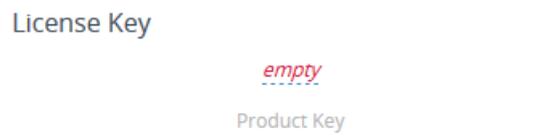
Figure 6-3: Viewing Product Key

License Key

QEE3C2A64FF016Y5

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:
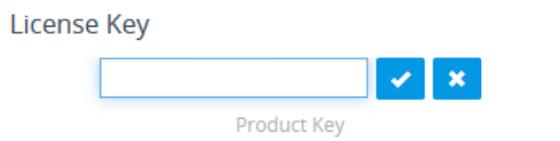
1.   Open the License Key page.

2.   Locate the Product Key group:

Figure 6-4: Empty Product Key Field

License Key

empty

Product Key

3.   Click "empty"; the following appears:

Figure 6-5: Entering Product Key

License Key

Product Key

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

International Headquarters
Naimi Park
6 Ofra Haza Street
Or Yehuda, 6032303, Israel
Tel: +972-3-976-4000
Fax: +972-3-976-4040

AudioCodes Inc.
80 Kingsbridge Rd
Piscataway, NJ 08854, USA
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-110016