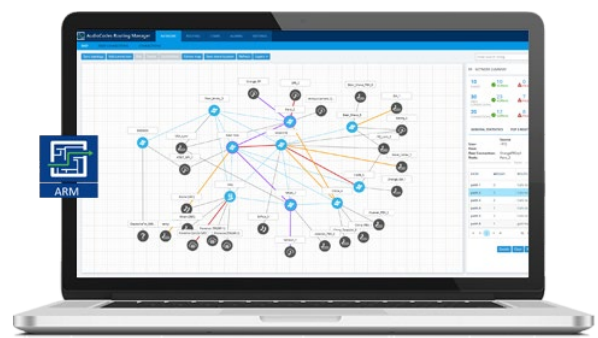


# AudioCodes Routing Manager (ARM)

Version 9.4





## Table of Contents

<b>1</b>	<b>Overview .....</b>	<b>9</b>
1.1	Managed AudioCodes Devices.....	9
<b>2</b>	<b>What's New in Version 9.4.....</b>	<b>11</b>
2.1	<b>ARM for Teams Hosters .....</b>	<b>11</b>
2.1.1	'Customer' Entity: Hosted Teams Multi-Tenant Direct Routing .....	11
2.1.1.1	Defining a 'Customer' Entity (Teams Tenant) .....	12
2.1.1.1.1	Adding a New 'Customer' Entity .....	13
2.1.1.1.2	Editing a 'Customer' Entity .....	14
2.1.1.1.3	Deleting a 'Customer' Entity .....	14
2.1.1.1.4	Locking/Unlocking a 'Customer' Entity .....	15
2.1.1.1.5	Viewing the 'Customers' Page.....	15
2.1.1.2	Defining 'Customer' Entities using ARM Users & Extended Policy Studio....	16
2.1.1.3	'Customer' Entities Supported in Routing Rules .....	18
2.1.1.4	'Customer' Entity Supported in Test Route .....	20
2.1.1.5	'Customer' Entity in ARM Calls .....	21
2.1.1.6	'Customer' Entity Related Statistics .....	22
2.1.2	Quota (Calls Time Limit) per Peer Connection / Set of Peer Connections.....	22
2.1.2.1	Defining a Calls Quota .....	23
2.1.2.2	Defining a Calls Quota Threshold .....	25
2.1.2.3	Attaching a Calls Quota to a Peer Connection or a Resource Group .....	25
2.1.2.4	Map Representation of the Quota Status .....	29
2.1.2.5	Quota Threshold Alarms .....	30
2.1.2.6	Statistics .....	30
2.1.2.6.1	Quota over Time .....	31
2.1.2.6.2	Peer Connection over Time .....	32
2.1.2.6.3	Resource Group over Time .....	33
2.1.2.6.4	Resource Group by Peer Connection .....	33
2.1.3	CAC Profiles.....	34
2.1.3.1	Defining a CAC Profile .....	34
2.1.3.2	Defining a CAC Profile Threshold .....	35
2.1.3.3	Disabling CAC and Session Counting.....	35
2.1.3.4	Attaching a CAC Profile to a Peer Connection.....	35
2.1.3.4.1	Map Representation of CAC Status .....	37
2.1.3.4.2	Peer Connection CAC Threshold Alarms .....	38
2.1.3.4.3	Peer Connection Session Statistics .....	38
2.1.3.5	Attaching a CAC Profile to a VoIP Peer .....	39
2.1.3.5.1	VoIP Peer CAC Threshold Alarms .....	40
2.1.3.5.2	VoIP Peer Session Statistics .....	40
2.1.4	Prefix Group Usage Visibility .....	41
<b>2.2</b>	<b>New Engine for Validation of Prefix/DID Uniqueness .....</b>	<b>43</b>
<b>2.3</b>	<b>ARM Integration with Azure AD .....</b>	<b>45</b>
2.3.1	Configuring the ARM in the Azure Portal.....	45
2.3.2	Azure AD as a Source for Users in the ARM .....	49
2.3.3	Azure AD for Operators Authentication .....	51
2.3.4	Azure AD for REST Requests Authentication.....	52
2.3.5	Revoking Azure User Tokens .....	54
<b>2.4</b>	<b>Appending   Deleting Prefixes in a Prefix Group via the REST API .....</b>	<b>55</b>
<b>2.5</b>	<b>VoIP Peers Page .....</b>	<b>55</b>
<b>2.6</b>	<b>Customized ARM Connection (IP Group Name, User-Defined IP Profile &amp; Media Realm) .....</b>	<b>56</b>
<b>2.7</b>	<b>Authentication Order.....</b>	<b>57</b>

<b>3</b>	<b>Supported Platforms.....</b>	<b>59</b>
<b>4</b>	<b>Earliest SBC/GW Software Versions Supported by ARM Features .....</b>	<b>61</b>
<b>5</b>	<b>Resolved Issues in ARM 9.4.....</b>	<b>63</b>
<b>6</b>	<b>Tested ARM Capacities.....</b>	<b>65</b>
<b>7</b>	<b>Known Limitations and Workarounds.....</b>	<b>67</b>

---

## List of Tables

---

Table 1-1: AudioCodes Devices Supported by ARM Version 9.4 .....	9
Table 3-1: ARM 9.4 Supported Platforms .....	59
Table 4-1: ARM Features Supported by the Earliest Node Software .....	61
Table 5-1: Resolved Issues in ARM 9.4.....	63
Table 6-1: Tested ARM Capacities.....	65
Table 7-1: Known Limitations and Workarounds.....	67



## 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: May-04-2021

## 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
ARM Installation Manual
ARM User's Manual
ARM REST API Developer's Guide
Mediant 9000 SBC User's Manual
Mediant 4000 SBC User's Manual
Mediant SE SBC User's Manual
Mediant SE-H SBC User's Manual
Mediant VE SBC User's Manual
Mediant VE-H SBC User's Manual
Mediant 1000B Gateway and E-SBC User's Manual
Mediant 800B Gateway and E-SBC User's Manual
Mediant 500 Gateway and E-SBC User's Manual
Mediant 500 MSBR User's Manual
Mediant 500L Gateway and E-SBC User's Manual
Mediant 500L MSBR User's Manual
MP-1288 High-Density Analog Media Gateway User's Manual
One Voice Operations Center Server Installation, Operation and Maintenance Manual
One Voice Operations Center Integration with Northbound Interfaces
One Voice Operations Center User's Manual
One Voice Operations Center Product Description
One Voice Operations Center Alarms Guide
One Voice Operations Center Security Guidelines

## 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 Overview

These *Release Notes* describe the new features and known issues in version 9.4 of the AudioCodes Routing Manager (ARM).

## 1.1 Managed AudioCodes Devices

ARM 9.4 supports the following AudioCodes devices (Gateways and SBCs) referred to in the ARM GUI as *nodes*:

**Table 1-1: AudioCodes Devices Supported by ARM Version 9.4**

Device	Major Versions
Mediant 9000 SBC	7.20A.258 and later
Mediant 4000 SBC	7.20A.258 and later
Mediant 2600 SBC	7.20A.258 and later
Mediant SE/VE SBC	7.20A.258 and later
Mediant 1000B Gateway and E-SBC	7.20A.258 and later
Mediant 800B Gateway and E-SBC	7.20A.258 and later
Mediant 800C	7.20A.258 and later
Mediant 500 E-SBC	7.20A.258 and later
Mediant 500L - SBC	7.20A.258 and later
Mediant SBC CE (Cloud Edition)	7.20A.258 and later
Mediant 3000 Gateway only	7.00A.142.001 and later
Mediant 3100 SBC, Gateway or Hybrid	7.40M3.002.084 and later



**Note:** See also Section 4 for the earliest device version supported by the ARM *per ARM feature*.

This page is intentionally left blank.

## 2 What's New in Version 9.4

This section covers the new features and capabilities introduced in ARM 9.4.

### 2.1 ARM for Teams Hosters

ARM 9.4 provides a holistic solution for Microsoft Teams Hosters. In addition to support for Teams Local Media Optimization (which was part of ARM 9.2), ARM 9.4 adds support for Teams Multitenancy (Teams Super Trunk), Call Admission Control (CAC), calls quota and other features described in this section.

#### 2.1.1 'Customer' Entity: Hosted Teams Multi-Tenant Direct Routing

ARM 9.4 adds support for a hosted **Teams multi-tenant** Direct Routing solution (ARM 'customer' entity feature). Microsoft Teams Hosters that implement the Microsoft recommended **Super Trunk** deployment model for multi-tenancy can use this feature and have each tenant represented by an ARM 'customer' entity. All 'customer' entities can traverse the same Peer Connection/VoIP Peer (SBC IP Group) on the AudioCodes Direct Routing SBC.

The ARM has added a new logical entity named 'customer' (**Teams tenant**). The 'customer' entity can be defined uniquely by either Prefix Groups or by a special tag assigned to a call, in the Policy Studio (Policy Studio Tag) if the operator wants to manage 'customer' entity DID's in the Users page and use the Policy Studio and other ARM users' capabilities. In this way, the 'customer' entity's DID's can be managed in both the Prefix Group or the ARM Users page (a combination of the two is also allowed).

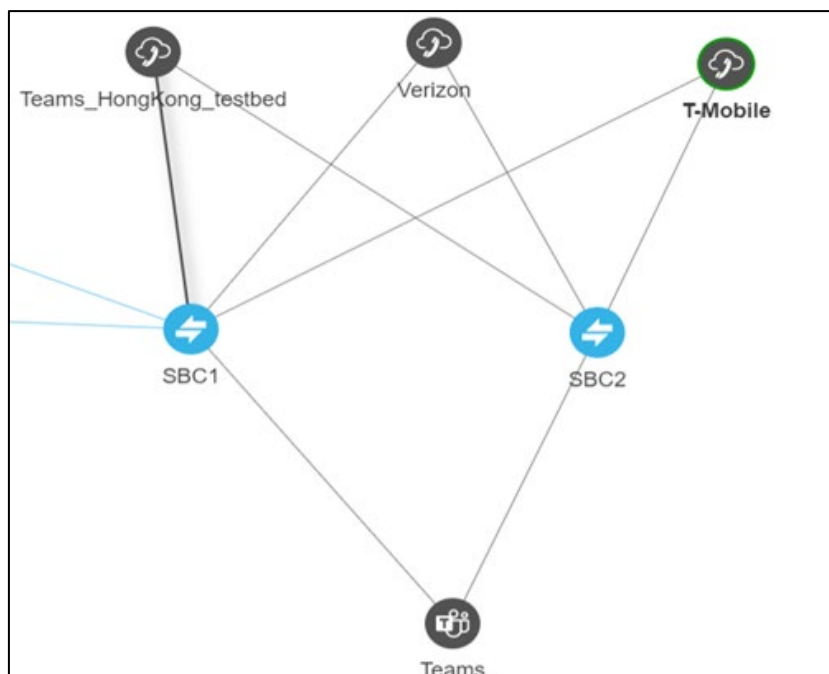
New statistics and alarms, related to the 'customer' entity, were added.

ARM Routing capabilities were extended to support the 'customer' entity as a routing condition (specific 'customer' entities or all 'customer' entities). This also includes SIP header manipulations, required by Teams multi-tenancy, that can now easily be performed by the ARM (SIP 'Contact' header).

In addition, the ARM provides CAC capabilities for each Teams tenant. Since a Super Trunk (single IP Group for Teams) is provisioned on the SBC, individual CAC Profiles can't be applied in the SBC for each individual tenant that shares the Super Trunk. The ARM adds this capability by applying and performing CAC for each tenant that shares the same Super Trunk (Peer Connection/VoIP Peer/IP Group). This includes the following CAC capabilities:

- Capability to define ingress CAC for logical customers under a VoIP Peer
- Capability to define egress CAC for logical customers under a VoIP Peer
- Capability to define CAC applicable for both directions

The following network diagram demonstrates this feature's most common use case:



- Multiple 'customer' entities (Teams tenants) can share the same ARM Peer Connection for Teams access (northbound).
- Operators can share the same Service Providers/PSTN SIP trunks for multiple 'customer' entities (southbound).



**Note:** Note that for redundancy purposes, multiple SBCs can be used leading to the same VoIP Peers, with local Peer Connections (IP Groups) on each SBC.

- Connectivity to Microsoft using a 'derived trunk' setup.
  - A derived trunk can be considered a Super Trunk using only one (1) IP Group on each SBC (ARM Peer Connection).
  - Each unique 'customer' entity / Teams tenant making outbound calls can be identified by the FQDN in the 'Contact' or 'From' header, or by its DID.
  - Inbound calls from SIP trunks to the Teams 'customer' entity can be identified and associated with a specific 'customer' entity / Teams tenant by the destination DID (which can be managed either in the ARM Users page or by the Prefix Group).
  - This type of trunk eliminates the need for each 'customer' entity to have its own IP Group/Peer Connection with Sip:options requesting health checks.
  - Using the ARM for routing allows a very high number of 'customer' entities to be supported (as it becomes a logical entity).

### 2.1.1.1 Defining a 'Customer' Entity (Teams Tenant)

The new logical entity named 'customer' (Teams tenant) has been added to the ARM GUI under **Network > Customers**:

AudioCodes Routing Manager								
NETWORK								
CUSTOMERS								
NAME	QAC STATE	ADMIN STATE	PREFIX GROUP	POLICY STUDIO TAG	SIP HEADER NAME	SIP HEADER VALUE	QAC PROFILE	
customer1	●	●	HLGJapan	Verizon	CONTACT_HOST	stun1.audiocodes.com	qac_global	
customer2	●	●			CONTACT_HOST	cust2.com	qac_morningfromteams	
customer3	●	●		T-Mobile	CONTACT_HOST	cust3.com	qac_morningfromteams	
customer4	●	●	PT		CONTACT_HOST	cust4.com	qac_morningfromteams	
customer5	●	●	PT		CONTACT_HOST	cust5.com	qac_morningfromteams	
customer6	●	●	PT		CONTACT_HOST	cust6.com	qac_morningfromteams	

The page allows **Add**, **Edit**, **Delete**, **Lock/Unlock** and **Refresh** actions for a specific single 'customer' entity.

Before implementing the feature, best practice is for operators to decide *how to identify* a 'customer' entity: using either Prefix Groups, or ARM Users.

Note that a combination of the two is also supported, but may be less convenient.

A more detailed explanation and use-case for each 'customer' definition method (either with Prefix Group or with Users) is provided as part of 'customer' entity definition and parameters and in Section 2.1.1.2.

### 2.1.1.1.1 Adding a New 'Customer' Entity

When adding a new 'customer' entity, operators must provide the following information:

**Name** - Mandatory. Unique name of the 'customer'.

**Prefix Group** - Used if the operator chooses to identify a 'customer' entity with Prefix Groups. The operator can select a Prefix Group or several Prefix Groups previously defined (**Settings > Call Flow > Prefix Group**). Multiple Prefix Groups are treated as 'or' in terms of 'customer' entity definition (DIDs and ranges from all the selected Prefix Groups are considered to belong to the 'customer' entity). A Prefix Group can include not only full DIDs but also ranges. Note that the same Prefix Group cannot be used for several 'customer' entities as it uniquely identifies 'customer' entity DIDs. However, the ARM does not prevent a collision between the ranges of Prefix Groups; it's the operator's responsibility to prevent a collision of ranges between 'customer' entities.

**Policy Studio Tag** - Used if the operator chooses to manage 'customer' DIDs in the ARM Users page and thereby benefit from ARM Users capabilities (such as Policy Studio with pre-routing manipulations or Users Groups). The Policy Studio Tag should be provided in the Policy Studio (for incoming and outgoing calls) and is used by the ARM mainly for CAC counting and enforcement for specific 'customer' entities / Teams tenants. The extension for this Tag in a Policy Studio action is described under Section 2.1.1.2.

**SIP header** - Each unique 'customer'/Teams tenant making outbound calls is identified/marked by Teams with the FQDN in the 'Contact' or 'From' header. A call in the direction 'to Teams' should have this 'Contact' header identification as well. From Teams' perspective, this is the way to identify and distinguish between 'customer' entities / tenants. The ARM provides an easy way to put the predefined string (the one used by Teams to identify a tenant) in the 'Contact' header for calls toward Teams (this option is described in Section 2.1.1.3). The SIP header attribute allows the operator to provide a string to be used for the 'Contact' header. Note that it should be coordinated with the Teams settings for the ARM 'customer' entity / Teams tenant.

**CAC profile** – can optionally be attached per 'customer' entity. For a description of a CAC profile and its capabilities, see under Section 2.1.3). The operator can attach a CAC profile to a 'customer' entity with both directions or a one-direction sessions limitation (defined under **Settings > Routing > CAC profiles**):

NAME	TOTAL LIMIT	INCOMING LIMIT	OUTGOING LIMIT
cac_global	10		
cac_incoming(from team's customer)		10	
cac_outgoing(to team's customer)			10

Operators can reuse the same CAC Profile for multiple 'customer' entities.

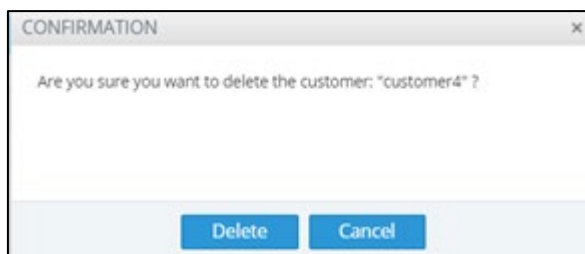
#### 2.1.1.1.2 Editing a 'Customer' Entity

The option to **Edit** a 'customer' entity allows the operator to change all the attributes provided in the **Add customer** action (including 'Name').

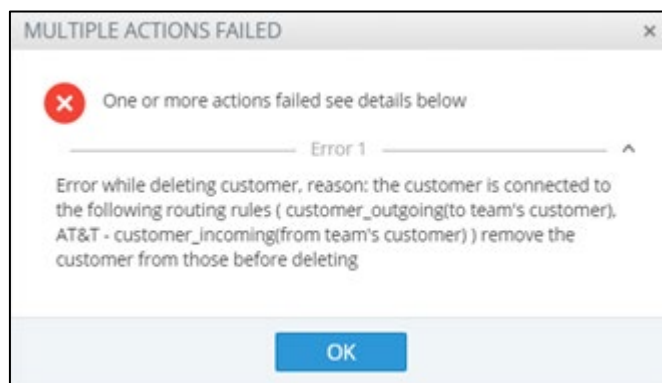
If during **Edit** the operator updates the 'customer' entity's CAC profile (or adds a CAC profile), the ARM verifies if the 'customer' entity should be blocked / unblocked due to the change (from the CAC's perspective).

#### 2.1.1.1.3 Deleting a 'Customer' Entity

The action **Delete** a 'customer' entity should be used to delete an 'existing' 'customer' entity. The operator is asked for confirmation before the delete action:



Note that if a 'customer' entity explicitly appears in a Routing Rules condition, the ARM does not allow deleting it until it is removed:

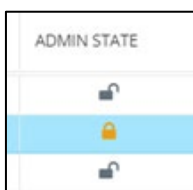


#### 2.1.1.1.4 Locking/Unlocking a 'Customer' Entity

This action allows operators to manually Lock or Unlock a specific 'customer' entity for maintenance due to administrative reasons. It blocks incoming and outgoing calls associated with the locked 'customer' entity.

Note that when a Lock action is applied to the 'customer' entity, the ARM does not allow any calls to/from the 'customer' entity (tenant).

The Lock/Unlock action is reflected in the 'customer' entity's Admin state:



#### 2.1.1.1.5 Viewing the 'Customers' Page

The Customers page (**Network > Customer**) provides operators the capability to view all provisioned 'customer' entities (Teams tenants) in a table (one row per 'customer' entity). In addition to the information configured per 'customer' entity (provided by the operator in the **Add/Edit** action), the following two columns are shown in the table for each 'customer' entity:

- **Admin State** - can be either **Locked** or **Unlocked**; reflects an operator's Lock/Unlock action applied to the 'customer' entity. The ARM rejects a calls routing request for a Locked 'customer' entity.
- **CAC State** - shown only for 'customer' entities with an attached CAC Profile. Reflects the CAC status of the 'customer' entity based on the current number of concurrent sessions of the 'customer' entity, related to the attached CAC profile. It can have one of the following values:
  - **Unblock** - the 'customer' entity didn't reach the allowed number of simultaneous sessions and calls to/from it.

- **Block** - the 'customer' entity reached the maximum number of allowed simultaneous sessions defined in the attached CAC Profile and calls are currently blocked.
- **Block Incoming** – the 'customer' entity reached the maximum number of incoming calls and only incoming calls are blocked.
- **Block Outgoing** – the 'customer' entity reached the maximum number of outgoing calls defined in the attached CAC Profile and outgoing calls are currently blocked.

NAME	CAC STATE	ADMIN STATE	PREFIX GROUPS	POLICY STUDIO TAG	SIP HEADER NAME	SIP HEADER VALUE	CAC PROFILE
customer1	✓	⬆	mgjapan		CONTACT_HOST	ibc18.audiocodes.com	cac_global
customer2	✓	⬆		Webcam	CONTACT_HOST	cust2.com	cac_incomingfrom team...
customer3	✓	⬆		T-Mobile	CONTACT_HOST	cust3.com	cac_outgoingto team's cu...
customer4	✓	⬆	pf1		CONTACT_HOST	cust4.com	cac_global
customer5	✓	⬆	pf2		CONTACT_HOST	cust5.com	cac_incomingfrom team...
customer6	✓	⬆	pf3		CONTACT_HOST	cust6.com	cac_outgoingto team's cu...
cust_temporary		⬆		Cust_temp	CONTACT_HOST	cust_temp	

The 'Customers' page can tabulate thousands of entries; a smart search and filter engine in the uppermost right corner facilitates management.

In addition to a string search, the following filters are supported:

Name:

CAC State:

Prefix Group:

Policy Studio Tag:

Administrative State:

SIP Header Name:

SIP Header Value:

CAC Profile:

For example, the operator can select one of the CAC Profiles and filter all 'customer' entities listed in the page using this specific profile. The operator can alternatively select a 'customer' entity in the Customers page filtered by Prefix Groups, etc.

### 2.1.1.2 Defining 'Customer' Entities using ARM Users & Extended Policy Studio

In a simple scenario, it's quite convenient to define a 'customer' entity (Teams tenant) in the ARM using a Prefix Group (or multiple Prefix Groups). However, deployment sometimes requires functionality which requires use of ARM Users capability (such as smart DID manipulation or replacement) or use of a Users Group. In this case, DIDs of 'customer' entities should be defined in the ARM's Users page. An example of a deployment like this is routing based on groups of users as destination. The operator can have cross-tenant (cross-'customer' entities) users who're allowed to dial to specific destinations (specific countries), or long distance. These users can have a property in the ARM's Users page which will allow composing a Users Group of 'Allowed for long distance'.



Another use case for defining a 'customer' entity DID in the ARM's Users page is use of short dial within the same 'customer' entity. Microsoft Teams does not support short dial yet this functionality can be achieved in the ARM. In this case, the ARM's Users Dictionary should include 'Full number' and 'short number' properties, which can be manipulated/substituted using the existing ARM Policy Studio engine.

Operators using ARM Users to define a 'customer' entity DID must have a Users property identifying the 'customer' entity in the Users Property Dictionary. AudioCodes recommends using Policy Studio for 'customer' entity tagging.

NAME	ORIGIN	COUNTRY	OFFICE PHONE	DISPLAY NAME	DEPARTMENT	MS LYNC LINE URI	TENANT	DLE
hongkonglyong	ARM		34697577					
ilyong	ARM	HongKong	+85234697577			+85297282142	hongkong.audiocodes...	+85
japanilyong	ARM	Japan	+815034697577			+81279998800		
User01	users	Canada	1101	User1	AT&T	110	AT&T	
User02	users	Canada	1102	User2	AT&T	120	AT&T	
User03	users	Canada	1201	User3	AT&T	130	AT&T	
User04	users	Canada	1202	User4	AT&T	140	AT&T	
User05	users	Canada	1301	User5	Verizon	150	Verizon	
User06	users	USA	1302	User6	Verizon	160	Verizon	
User07	users	USA	1401	User7	Verizon	170	Verizon	
User08	users	USA	1402	User8	Verizon	180	Verizon	
User09	users	USA	1501	User9	T-Mobile	190	T-Mobile	
User10	users	USA	1502	User10	T-Mobile	200	T-Mobile	
User11	users	USA	1601	User11	T-Mobile	210	T-Mobile	

Note that if 'customer' entity DIDs are defined in the ARM's Users page, operators cannot define a range of DIDs to be associated with these 'customer' entities.

Policy Studio was extended to support INCOMING CUSTOMER TAG and OUTGOING CUSTOMER TAG under ACTION. Note that the value assigned to these tags for a specific 'customer' entity must match the 'Policy Studio Tag' provisioned when defining the 'customer' entity.

The ARM needs these Incoming and Outgoing 'customer' entity tags assignments for classification of calls; they're used to categorize calls as belonging to a specific 'customer' entity and correctly handle / count the session in terms of the associated CAC Profile (hence direction is also necessary for the ARM).

The following two examples of Policy Studio rules match the DID (either from the SOURCE or the DESTINATION URI) and assign the 'customer' entity tag (INCOMING or OUTGOING) with the tenant value displayed in the Users page (**Users > Users**):

Note that rules with a 'customer' entity tag assignment can come in combination with other legacy Policy Studio rules.

Flow action should be used to define multiple rules to be applied to a call.

### 2.1.1.3 'Customer' Entities Supported in Routing Rules

ARM Routing Rules capabilities and Routing Engine were extended to support 'customer' entity based routing. Operators can compose a Routing Rule with either a specific 'customer' entity / set of selected 'customer' entities or all 'customer' entities in the SOURCE and DESTINATION definitions in a Routing Rule condition.

Following is an example of using specific 'customer' entities (which can be selected from the predefined Customers table) in a Routing Rule SOURCE matching condition:

Operators can also specify that the Routing Rule must be applied for all 'customer' entities (without selecting a specific 'customer' entity or list of 'customer' entities). This is a very powerful functionality especially in the case of a very high number of 'customer' entities. In this way, with a single rule the operator can define Calls Routing towards all the 'customer' entities with Teams Peer VoIP Peer destination (action). This single rule will cover calls toward Teams for all 'customer' entities coming from several SBCs.

Following is an example of a rule using **Use All Customers** in the Destination condition of a Routing Rule leading toward Teams.

**ADD ROUTING RULE**

Name \*  Live Test

Group: UI Test do not delete 1

SOURCE	DESTINATION	ADVANCED CONDITIONS	ROUTING ACTIONS
Prefixes / Prefix Groups	<input type="text"/>		
Hosts	<input type="text"/>		
User Groups	<input type="text"/>		
Customers	<input type="text"/> <div> <input checked="" type="checkbox"/> Use All Customers </div> <p>If a Customers is matched, the 'Destination' will be matched.</p>		

OK Cancel

As mentioned, each 'customer' entity is identified/indicated by Teams with the FQDN in the 'Contact' or 'From' header. The call in the direction 'to Teams' should have this 'Contact' header identification as well. ARM provides an easy way to put the predefined string (the one used by Teams to identify the tenant) in the Contact header for calls towards Teams.

In a Routing Rule's 'Routing Action', the operator can check the new **Use Contact host from destination customer** checkbox under the 'Advanced' section of a specific action. In this case, the ARM will automatically install the value (string) provisioned in the **SIP header** field of the customer definition table into **SIP Contact header** of the invite designated to reach Teams.

**EDIT ROUTING RULE**

Name \*  Live Test

Group: customersRG

**SOURCE** **DESTINATION** **ADVANCED CONDITIONS** **ROUTING ACTIONS**

Routing method:

[Online VoIP Peer] Teams

Advanced

Normalization After Routing

Source URI User  ☒ From ☒ PAI ☒ PPI

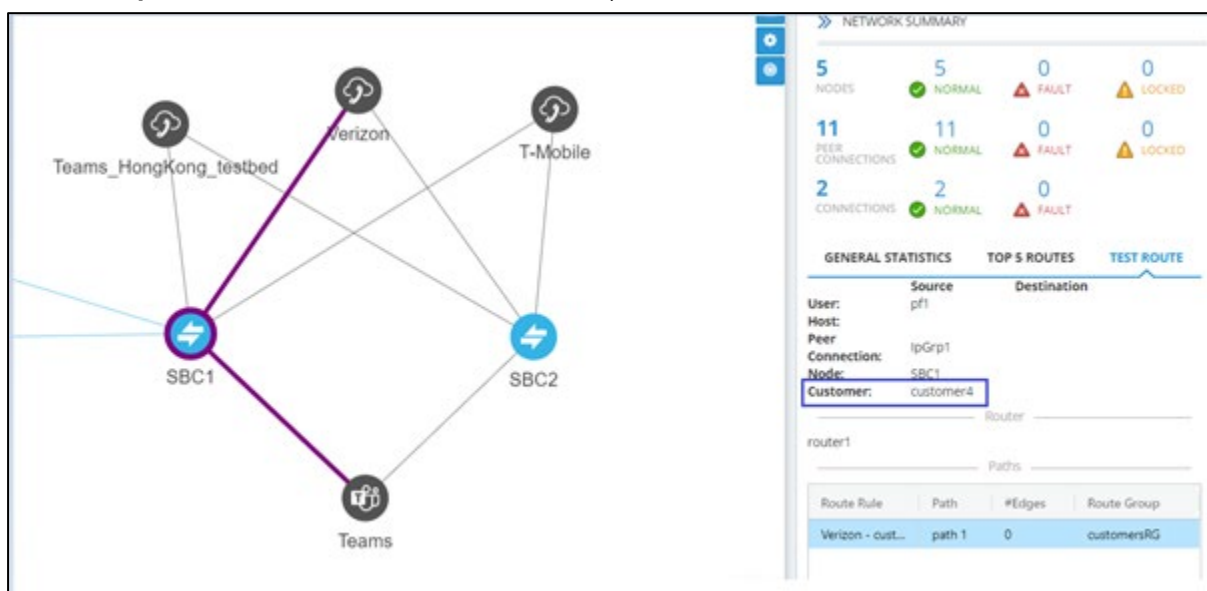
Destination URI User

☒ Use contact host from destination customer

Request URI

#### 2.1.1.4 'Customer' Entity Supported in Test Route

The 'customer' entity is also supported by the ARM's Test Route. The following example shows a test call coming from customer 4 (from Teams) toward Verizon SIP trunk. The ARM identifies the customer (shown in the Test Route Summary) based on the source DID (prefix **pf1** used for identification of **customer4**).



### 2.1.1.5 'Customer' Entity in ARM Calls

The 'customer' entity is reflected in calls per call; there's an indication of INCOMING or OUTGOING CUSTOMER (if a call is classified as to/from a 'customer' entity).

NETWORK	ROUTING	USERS	ALARMS	STATISTICS	CALLS	SETTINGS				
DESTINATION	DATE	INCOMING NODE	INCOMING PCON	INCOMING CUSTOMER	OUTGOING NODE	OUTGOING PCON	OUTGOING CUSTOMER	ROUTING RULE	SIP REASON	
any@172.17.129.12	08-Apr-21 10:07:51	SBC1	IpGrp3	customer2	SBC1	IpGrp0		AT&T - customer_incom...	BYE	
any@172.17.129.12	08-Apr-21 10:07:51	SBC1	IpGrp3	customer2	SBC1	IpGrp0		AT&T - customer_incom...	BYE	
any@172.17.129.12	08-Apr-21 10:07:51	SBC1	IpGrp1	customer2	SBC1	IpGrp0		AT&T - customer_incom...	BYE	
any@172.17.129.12	08-Apr-21 10:07:51	SBC1	IpGrp1	customer2	SBC1	IpGrp0		AT&T - customer_incom...	BYE	
any@172.17.129.12	08-Apr-21 10:07:50	SBC1	IpGrp3	customer2	SBC1	IpGrp0		AT&T - customer_incom...	BYE	
any@172.17.129.12	08-Apr-21 10:07:50	SBC1	IpGrp3	customer2	SBC1	IpGrp0		AT&T - customer_incom...	BYE	
any@172.17.129.12	08-Apr-21 10:07:50	SBC1	IpGrp1	customer2	SBC1	IpGrp0		AT&T - customer_incom...	BYE	
any@172.17.129.12	08-Apr-21 10:07:50	SBC1	IpGrp1	customer2	SBC1	IpGrp0		AT&T - customer_incom...	BYE	
any@172.17.129.12	08-Apr-21 10:07:50	SBC1	IpGrp1	customer2	SBC1	IpGrp0		AT&T - customer_incom...	BYE	
any@172.17.129.12	08-Apr-21 10:07:50	SBC1	IpGrp1	customer2	SBC1	IpGrp0		AT&T - customer_incom...	BYE	
any@172.17.129.12	08-Apr-21 10:07:50	SBC1	IpGrp3	customer2	SBC1	IpGrp0		AT&T - customer_incom...	BYE	
any@172.17.129.12	08-Apr-21 10:07:27	SBC1	IpGrp1	customer4	SBC1	IpGrp2		Verizon - customer_inco...	BYE	
any@172.17.129.12	08-Apr-21 10:07:08	SBC1	IpGrp1	customer4	SBC1	IpGrp2		Verizon - customer_inco...	BYE	
pf1@172.17.129.12	08-Apr-21 10:07:08	SBC1	IpGrp3				customer4	No match		
any@172.17.129.12	08-Apr-21 10:07:08	SBC1	IpGrp1	customer4	SBC1	IpGrp2		Verizon - customer_inco...	BYE	
any@172.17.129.12	08-Apr-21 10:07:08	SBC1	IpGrp1	customer4	SBC1	IpGrp2		Verizon - customer_inco...	BYE	
pf1@172.17.129.12	08-Apr-21 10:07:08	SBC1	IpGrp3				customer4	No match		
any@172.17.129.12	08-Apr-21 10:07:07	SBC1	IpGrp1	customer4	SBC1	IpGrp2		Verizon - customer_inco...	BYE	

The Calls List page in ARM 9.4 also provides new 'customer' entity related filters; the operator can filter calls per the selected 'Incoming customer' and 'Outgoing customer':

CALLS LIST						
<div> <div> <div> <div>Filters</div> <div> <div>Destination:</div> <div>Session Id:</div> <div>Incoming Node:</div> <div>Incoming Peer Connection:</div> <div>Incoming Customers: customer2 X</div> <div>Outgoing Node:</div> <div>Outgoing Peer Connection:</div> <div>Outgoing Customers:</div> <div>Routing Rule:</div> </div> <div>Search</div> </div> </div> </div>						
SOURCE	DESTINATION	DATE	INCOMING NODE	INCOMING PCON	INCOMING CUSTOMER	
1301@192.168.1.101	any@172.17.129.12	08-Apr-21 10:53:35	SBC1	IpGrp1	customer2	
1301@192.168.1.101	any@172.17.129.12	08-Apr-21 10:53:25	SBC1	IpGrp3	customer2	
1301@192.168.1.101	any@172.17.129.12	08-Apr-21 10:53:24	SBC1	IpGrp3	customer2	
1301@192.168.1.101	any@172.17.129.12	08-Apr-21 10:53:24	SBC1	IpGrp3	customer2	
1301@192.168.1.101	any@172.17.129.12	08-Apr-21 10:53:23	SBC1	IpGrp1	customer2	
1301@192.168.1.101	any@172.17.129.12	08-Apr-21 10:53:23	SBC1	IpGrp1	customer2	
1301@192.168.1.101	any@172.17.129.12	08-Apr-21 10:53:23	SBC1	IpGrp3	customer2	
1301@192.168.1.101	any@172.17.129.12	08-Apr-21 10:53:23	SBC1	IpGrp1	customer2	
1301@192.168.1.101	any@172.17.129.12	08-Apr-21 10:53:23	SBC1	IpGrp3	customer2	
1301@192.168.1.101	any@172.17.129.12	08-Apr-21 10:53:23	SBC1	IpGrp1	customer2	
1301@192.168.1.101	any@172.17.129.12	08-Apr-21 10:53:23	SBC1	IpGrp3	customer2	
1301@192.168.1.101	any@172.17.129.12	08-Apr-21 10:47:04	SBC1	IpGrp1	customer2	
1301@192.168.1.101	any@172.17.129.12	08-Apr-21 10:46:53	SBC1	IpGrp1	customer2	
1301@192.168.1.101	any@172.17.129.12	08-Apr-21 10:46:53	SBC1	IpGrp3	customer2	
1301@192.168.1.101	any@172.17.129.12	08-Apr-21 10:46:53	SBC1	IpGrp3	customer2	
1301@192.168.1.101	any@172.17.129.12	08-Apr-21 10:46:53	SBC1	IpGrp1	customer2	
1301@192.168.1.101	any@172.17.129.12	08-Apr-21 10:46:53	SBC1	IpGrp3	customer2	

If a call to/from a 'customer' entity is dropped due to the number of simultaneous sessions (if a CAC Profile is attached to a 'customer' entity), it's also reflected in Call Details and in Call Summary:

CALL SUMMARY
Call Status: <b>Failure</b>
Source URI: 1301@192.168.1.101
Destination URI: any@172.17.129.12
Session Id: 5465801
Termination reason: 480
Description: CAC on incoming customer 'customer2'

### 2.1.1.6 'Customer' Entity Related Statistics

ARM 9.4 adds a new set of Statistics to provide operators visibility on 'customer' entity calls. A new tab has been added for this under **Statistics > Customers** in which operators can select 'Customer sessions over time'. On the right side of the page, the operator can select a specific 'customer' entity to review related sessions. The following statistics can be selected per 'customer' entity:

- Incoming average
- Incoming maximum
- Incoming minimum
- Outgoing average
- Outgoing minimum
- Outgoing maximum
- Total average
- Total minimum
- Total maximum

For the operator's convenience, when showing statistics over time, the ARM also displays the associated CAC Profile simultaneous sessions limit. This allows the operator to view the correlation and the number of sessions available for a 'customer' entity.



## 2.1.2 Quota (Calls Time Limit) per Peer Connection / Set of Peer Connections

ARM 9.4 introduces a new feature which allows the operator to put a quota on calls duration in minutes, on either a single Peer Connection or on a group of Peer Connections.

Using the ARM GUI or northbound REST API, the operator can define a time limit on calls, in minutes, and periodicity. Based on these definitions, the operator can define an action to block outgoing calls if the quota (limit) is reached, to be automatically applied by the ARM. An alarm is always generated if the limit is reached.

When applying the feature:

- The quota can be attached to either a single Peer Connection or to a group of Peer Connections gathered in a Resource Group of type 'Peer Connection'.
- The ARM counts only outgoing calls time (outgoing Peer Connections).
- Operators can define an alternative route (an Action in a Routing Rule) with an alternative Peer Connection if they want to handle a call when the primary Peer Connection is blocked due to the quota being reached.
- The ARM starts counting calls minutes from the moment the quota is attached to the Peer Connection or set of Peer Connections (and not from the beginning of the interval).

- Emergency calls are allowed regardless of the quota (even if the resource is blocked).
- If a customer wants to reset the quota, they can detach the quota from the entity or edit an existing one (increase the numbers, for example).
- The 'CDR calls' feature must be enabled in the ARM (**Settings > Advanced > Calls** and then select the option **Enable CDR calls**). The ARM uses calls information to get every call's duration and calculates the accumulated minutes of all calls per Peer Connection.

Note that in rare cases a call duration might go missing (if a specific call is not present in the CDRs for some reason).

### 2.1.2.1 Defining a Calls Quota

A quota can be put on calls, defined in **Settings > Routing > Calls Quota**. After selecting the **Calls Quota** tab, the Calls Quota screen opens displaying the following options:

- **Add** - to add a new Quota (row)
- **Edit** - to edit an existing Quota's settings
- **Delete**
- **Refresh**

Calls Quota			
<div> <span>Add</span> <span>Edit</span> <span>Delete</span> <span>Refresh</span> </div>			
NAME	QUOTA	PERIODICITY	BLOCK CALLS
manual_test	10	DAILY	<input checked="" type="checkbox"/>

To add a new quota, the operator selects **Add** and provides the following information:

- **Name** – Mandatory – user-defined unique name of the quota
- **Quota (minutes)** – Mandatory - number of minutes allowed in the selected period
- **Periodicity** – define the period for the quota to be applied:
  - **Daily** – the quota count, in minutes, will be reset *daily* (00:00-23:59)
  - **Weekly** – the quota count, in minutes, will be reset *weekly*. In this case, the operator must select from which day in the week counting should start and be reset (Example: Monday).
  - **Monthly** – the quota count, in minutes, will be allocated *monthly*. In this case, operators must select the day in the month from which counting of the minutes starts (Example: 5 days of each month).

Note that if operators select the start day to be after the 28<sup>th</sup> of the month, they'll receive the following warning:

☒ Monthly , Count from 

Months with less days will be counted until the last day of the month

☒

- **Block calls** – an action to be taken if the quota is reached during the specified period. If the operator selects this option, the Peer Connection's outgoing calls - except for emergency calls - will be blocked when the calls quota is reached. Note that an alarm is always generated when a quota is reached; the alarm cannot be disabled by the operator.



ADD CALLS QUOTA

Name: \*

VerizonQuota

Quota (minutes): \*

1000

Periodicity:

☐ Daily
 ☒ Weekly
 ☐ Monthly

, Count from

Mon

Block Calls:

☐

OK

Close

☒ Monthly
 

, Count from

04

☐

OK

Close

Submit

10

11

12

13

14

15

16

The selected row (quota) can be edited using the **Edit** button. All settings can be edited and reapplied. If operators change the frequency of the period when editing a quota, they must take the following into consideration:

CONFIRMATION

Changing the frequency will reset the calls duration count on a Peer Connection or Resource Group using this quota

Update

Cancel

To delete an existing quota, the operator selects it and clicks the **Delete** button. Note that a quota cannot be deleted while it is attached to a Peer Connection or a Resource Group. If the operator attempts to delete it, an error message is displayed along with the names of the specific topology elements currently using the quota.

The following table summarizes all defined quota information:



NAME	QUOTA	PERIODICITY	BLOCK CALLS
q1	2	MONTHLY (1)	×
VerizonQuota1	1000	MONTHLY (10)	✓
myQuota	10	WEEKLY (MON)	×

< < 1 > >> 25 Items per page Items 1-3 Items of 3

### 2.1.2.2 Defining a Calls Quota Threshold

The operator can adjust the calls quota threshold for the generation of an alarm in **Settings > Routing > Calls Quota**. When selecting the **Calls Quota** tab in **Calls Quota Configuration**:

CALLS QUOTA CONFIGURATION

Calls Quota Threshold: 75 %

Submit

The ARM can generate two alarms: One on hitting the Quota threshold and the other on crossing the Quota value. The ARM always generates Quota-related alarms regardless of the operator's setting to block (or not to block) a Peer Connection if the Quota is reached.

Note that the same threshold value (as a percentage) applies to all quotas defined in the ARM. To change the Calls Quota Threshold, the operator must click the **Submit** button.

### 2.1.2.3 Attaching a Calls Quota to a Peer Connection or a Resource Group

A calls quota can be attached either to a single Peer Connection or to a group of Peer Connections gathered in a Resource Group of type 'Peer Connection'.

Note that the same quota can be attached multiple times (reused for multiple Peer Connections or Resource Groups).

To attach a quota to a specific Peer Connection, the operator should select and edit the specific Peer Connection either from the Network Map or from the Peer Connections page:

EDIT PEER CONNECTION

Type:IPGroup

Name: \*IpGrp1

Weight:50

Node172.17.133.31-2Voip Peer2\_1

Normalization Before Routing

Source URI User:

Destination URI User:

Advance Conditions

Calls quota:manual\_test

CAC Profile:

☒ use global quality definitions
 ☐ use specific quality definitions
 

☐ MOS
 ☒ ASR

OKCancel

From the 'Calls quota' drop-down, the operator can select one of the previously defined quotas.

In the Peer Connections page (**Network > Peer Connections**), the quota is shown in the 'Calls Quota' column.

STATUS	NODE	NAME	VOIP PEER	IP GROUP	OPERATIVE STATE	ADMINISTRATIVE STATE	QUALITY	CALLS QUOTA	CAC PROFILE
✓	172.17....	IpGrp0	1_0	IpGrp0	✓	🔒	UNKNOWN		
✓	172.17....	IpGrp1	1_1	IpGrp1	✓	🔒	UNKNOWN		
✓	172.17....	C1619741350535	2_0	IpGrp0	✓	🔒	UNKNOWN		
🔒	172.17....	IpGrp1	2_1	IpGrp1	✓	🔒	UNKNOWN	manual_test	

When the Peer Connections page is used, the operator can filter all Peer Connections using the same defined quota:

Search and filter interface for Peer Connections. The 'Calls Quota' dropdown is open, showing the following options:

- q1
- VerizonQuota1
- myQuota

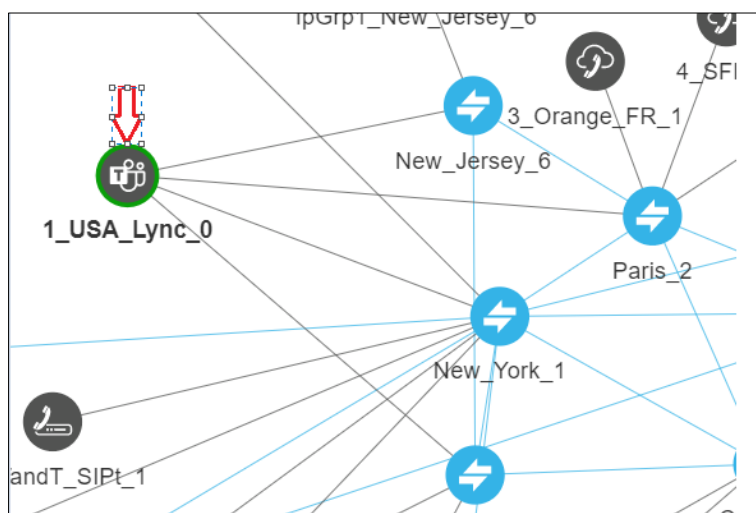
When selecting a Peer Connection with an attached quota, the following information related to quota counting is displayed:

Calls quota:	
Quota name:	myQuota
Calls duration (minutes):	0
Outgoing calls:	✓
Warning threshold reached:	No
Quota reached:	No

The Calls Quota can also be attached to several Peer Connections grouped in the same Resource Group. Note that only a Resource Group of type 'Peer Connection' can be associated with a Calls Quota. In this case, the calls balance, in minutes (defined by the Quota), is shared by all Peer Connections in the group.

If the operator wants to have a calls balance, in minutes, associated with a VoIP Peer (for example, with a specific PBX), and there are multiple Peer Connections connected to this VoIP Peer, all these Peer Connections should be gathered into a Resource Group (**Network > Resource Group**). After that, the Quota can be attached to this Resource Group.

In the following Network Topology, for example, Peer Connections come from four different SBCs to Teams:



To apply a quota to this VoIP Peer, the operator first defines a Resource Group made up of these four Peer Connections (coming from four different SBCs – New\_Jersey, Paris, New\_York and Texas), and then attaches the Quota:

Note that the possibility of selecting a 'Calls Quota' becomes available only when the operator selects Resource Group type to be Peer Connection. The attached Quota is shown in the table of the Resource Groups:

NAME	TYPE	ELEMENTS	CALLS QUOTA
1	Peer C...	IpGrp0 (New_York_1)	
pCons	Peer C...	IpGrp2 (69), IpGrp0 (69), IpGrp1 (69)	
Teams_PCons_Group	Peer C...	IpGrp0 (New_York_1), IpGrp0 (Paris_2), IpGrp0 (New_Jersey_6)	Teams_calls_Budget

When a Resource Group with an attached Quota is selected, relevant information about the Calls Quota status is displayed on the right side of the page:

**Note:**

- If the operator tries to attach a Quota to a Resource Group and one of the Peer Connections in this group already has a Quota, the operation will fail.
- If the operator tries to add a Quota to a Peer Connection that is attached to a Resource Group with a Quota, the operation will fail.
- When there are two Resource Groups with the same Peer Connection, if a Quota is attached to one of the groups and the operator tries to attach a Quota to the other group, the operation will fail.

### 2.1.2.4 Map Representation of the Quota Status

If the Peer Connection has a Quota attached, when clicking this Peer Connection in the Topology Map the operator will see information related to the Calls Quota in the Summary.

Calls quota:	
Quota name:	quota_2
Calls duration (minutes):	1
Outgoing calls:	✗
Warning threshold reached:	Yes
Quota reached:	Yes

If the Peer Connection is attached to a Resource Group with a Quota, the ARM shows the Resource Group name in the Peer Connection summary; the ARM doesn't show the Quota information.

The ARM allows the operator to view the Quota-related status of the Peer Connection in the Network Map (and review which Peer Connections are blocked due to the Calls Quota being reached). To view the Quota status of the network, the operator can select the quota layer in the ARM Topology Map.

Layers^	
topology	<input checked="" type="checkbox"/>
quality	<input type="checkbox"/>
quota	<input checked="" type="checkbox"/>
CAC	<input type="checkbox"/>

**Apply**



Note that it can be combined with other layers in the customer's network.



### 2.1.2.5 Quota Threshold Alarms

The ARM always generates Quota-related alarms regardless of the operator's setting to block (or not to block) a Peer Connection if the Quota balance is reached. The operator can choose whether to block the Peer Connection when the Quota is reached, or not.


The following severities are supported for Quota-related alarms:

- **Warning** – generated for a Network Topology element when the time spent by a specific Peer Connection (or Resource Group) reaches the Threshold limit (as a percentage) defined in **Settings > Routing > Calls Quota**.
- **Critical** – generated when the Quota is reached for a specific Network Topology element (Peer Connection or Resource Group).
- **Clear** – generated when the end of the period resets the quota for the relevant Network Topology element. The quota alarm also can be cleared when the quota is deleted from the Peer Connection or Resource Group, or when the limit or periodicity of a quota is changed.

The following example shows a generated alarm and its fields:

SEVERITY	DATE AND TIME	NAME	ALARM SOURCE	DESCRIPTION
	11-Feb-21 14:58:32	Calls duration quota usage	Node#172.17.133.30-1/PeerConnection#IpGrp0	Peer Connection IpGrp0 calls quota cleared
	11-Feb-21 14:34:00	Calls duration quota usage	Node#172.17.133.30-1/PeerConnection#IpGrp0	Peer Connection IpGrp0 calls quota limit has been reached

>> HISTORY ALARMS SUMMARY

Severity:  Critical

Date & Time: 11-Feb-21 14:34:00

Name: Calls duration quota usage

Source: Node#172.17.133.30-1/PeerConnection#IpGrp0

Alarm Type: Other

Probable Cause: Threshold Crossed

Description: Peer Connection IpGrp0 calls quota limit has been reached

Additional Info 1:

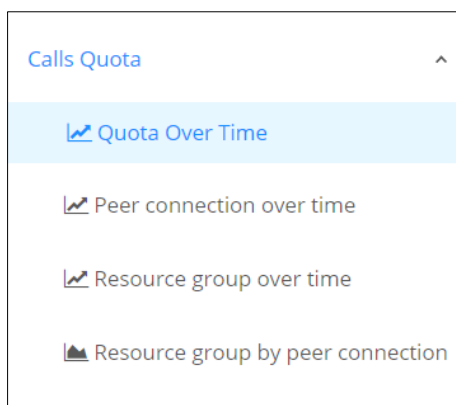
Additional Info 2:

Additional Info 3:

### 2.1.2.6 Statistics

To provide visibility on the balance of calls minutes assigned to ARM Peer Connections, ARM 9.4 introduces new statistics associated with Quota information.

These statistics are added as a new sub-tab **Calls Quota** under the **Statistics** tab.



Quota-related statistics are collected at an interval of every five minutes (like all other statistics in the ARM).

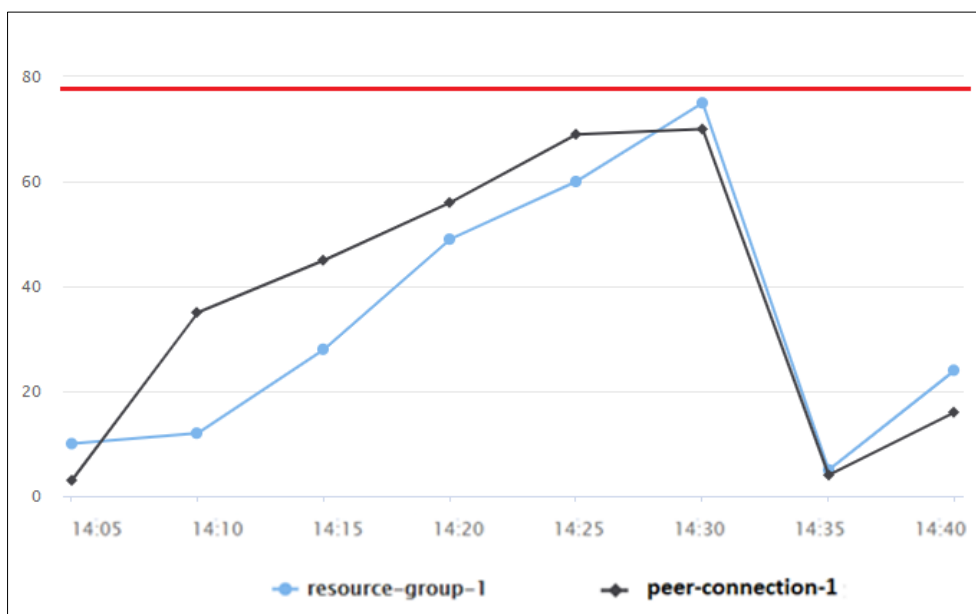
### 2.1.2.6.1 Quota over Time

This sub-tab presents the accumulated number of calls minutes for all Peer Connections or for Resource Groups associated with a specific quota. The operator can select a quota, Network Topology element type to be represented (either Peer Connections or Resource Group) and the ARM will automatically filter relevant Network Topology elements (for example, a Peer Connection to which the quota is attached).

 A screenshot of a web application form titled "Quota over Time". The form is divided into two main sections: "FILTERS" and "STATISTICS".  
 The "FILTERS" section includes a "DATE" subsection with two options: "Date range:" (unselected) and "Date relative time:" (selected). The "Date range:" option shows a date range from "17-Feb-21 00:00" to "17-Feb-21 23:59". The "Date relative time:" option shows "Last: 3" and "Hours".  
 The "STATISTICS" section includes a "Type:" dropdown menu set to "Accumulated Duration". Below it is a "Quota name:" dropdown menu set to "Teams\_calls\_Budget". Then is a "Quota element type:" dropdown menu set to "Resource Group". Finally, there is an "Elements:" section with a tag "Teams\_PCons\_Group" and a close button (X).  
 At the bottom of the form is a blue "Submit" button.

When submitted, the ARM will represent minutes spent by each selected Network Topology element (for example, Peer Connections to which the calls quota was assigned).

In the example below, a reset occurred because the period defined in the quota that was assigned to both Peer Connections, ended:



**Note:**

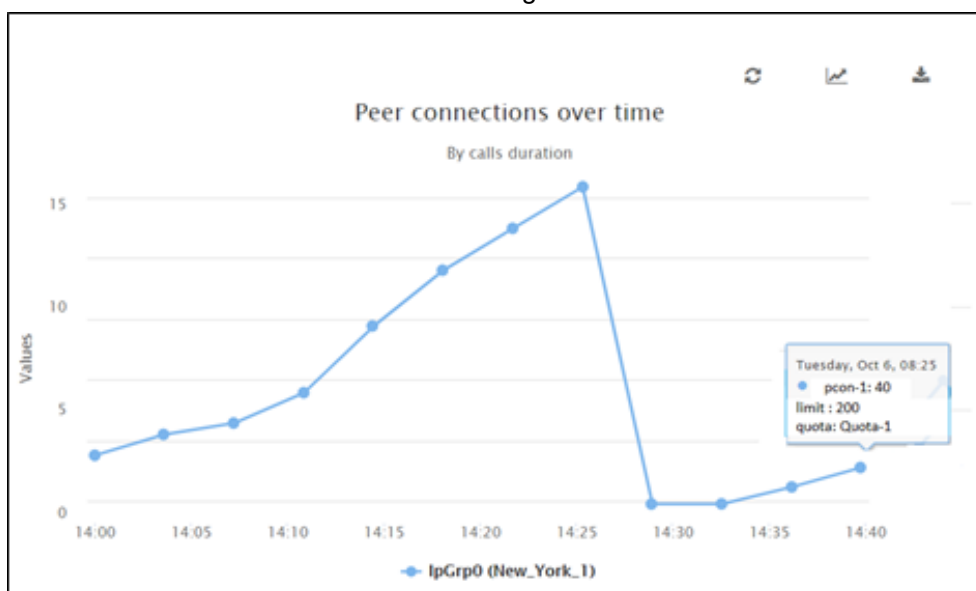
If a call starts before the quota is reached:

- the ARM will not drop the call
- the call will be calculated

In this case, the quota can be exceeded and it will be shown in the statistics.

## 2.1.2.6.2 Peer Connection over Time

This statistic allows the operator to select a specific Peer Connection (or multiple Peer Connections – where each can have a different Quota) and view the calls time (minutes) over time. Moreover, a tooltip displays for each graph the name of the quota associated with the Peer Connection and the minutes assigned.

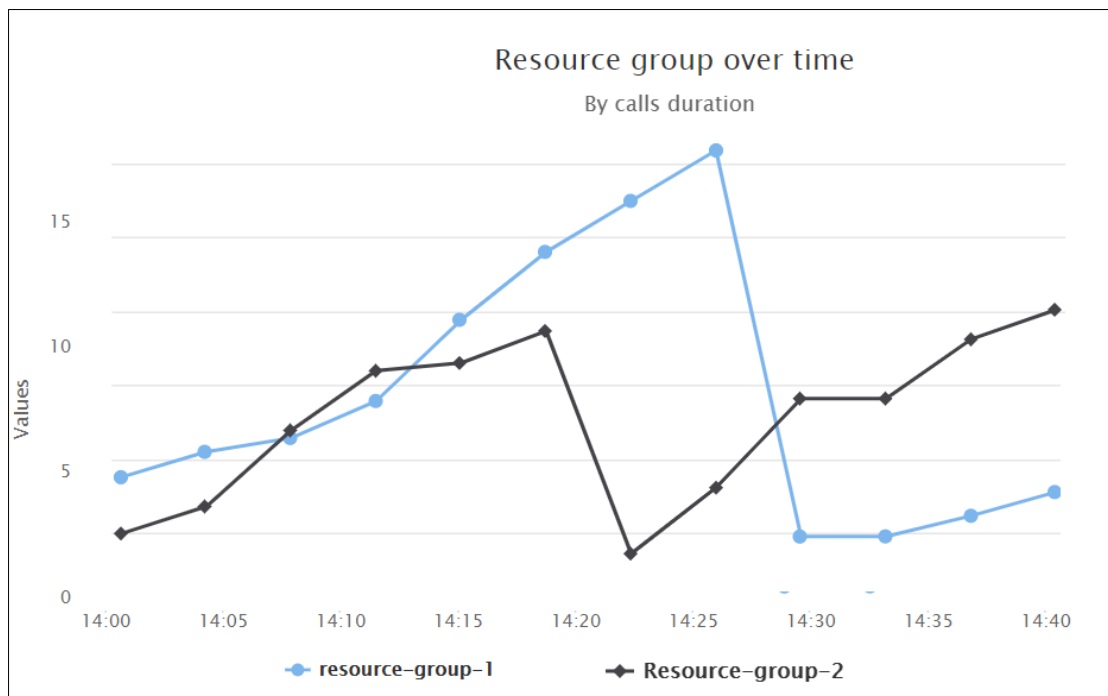




### 2.1.2.6.3 Resource Group over Time

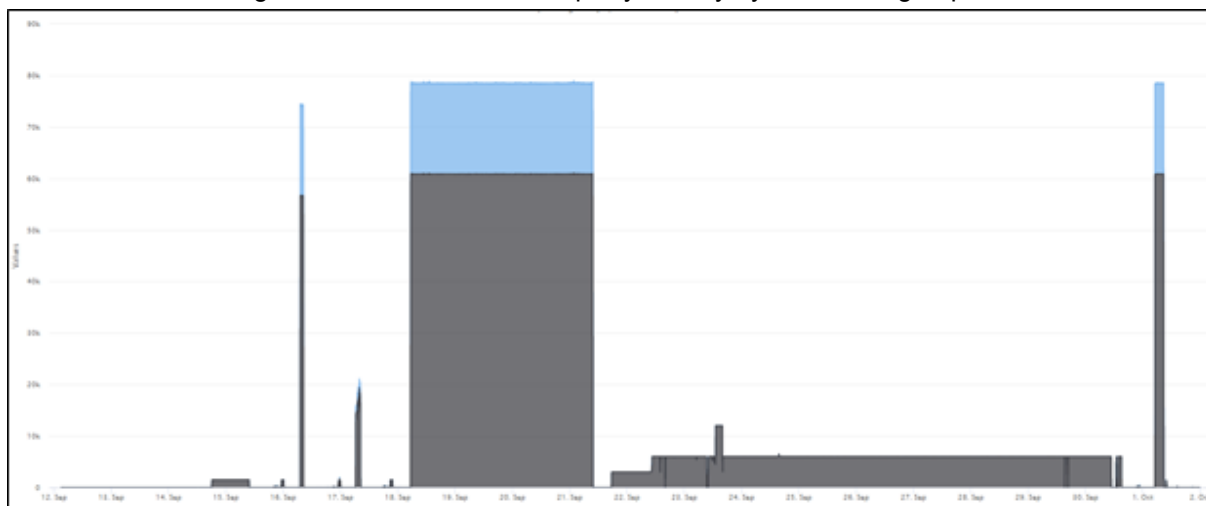
This statistic allows the operator to select a specific Resource Group (or multiple Resource Groups where each can have a different quota) and view the calls time (in minutes) over a timeline per Resource Group (the accumulated value for all Peer Connections in the Resource Group).

Note that only Resource Groups of type 'Peer Connection' can be selected. A tooltip displays for each graph the name of the quota associated with the Resource Group, and the limit (the number of minutes defined in the quota balance).



### 2.1.2.6.4 Resource Group by Peer Connection

This statistic (a stacked area by default) allows the operator to view consumption of calls minutes per Peer Connection in a specific Resource Group with an attached quota. In this way, the operator can see, for example, that a quota allocated to a Resource Group connecting Teams is consumed unequally, mainly by one of the group's Peer Connections.



## 2.1.3 CAC Profiles

Call Admission Control (CAC) is the practice or process of regulating traffic volume in voice communications, usually reflected by a maximum number of allowed simultaneous sessions in the network.

ARM 9.4 introduces the capability to define CAC Profiles that can later be attached to 'customer' entities (Teams Super Trunk tenants), Peer Connections and VoIP Peers, giving operators another way to balance and control the number of sessions throughout the entire network and to prevent oversubscription.

Operators will be able to limit the

- incoming Peer Connection / Customer or the connected VoIP Peer
- outgoing Peer Connection / Customer or the connected VoIP Peer
- total session

Operators will also be able to

- control the threshold of the warning alarm
- disable the entire CAC feature

### 2.1.3.1 Defining a CAC Profile

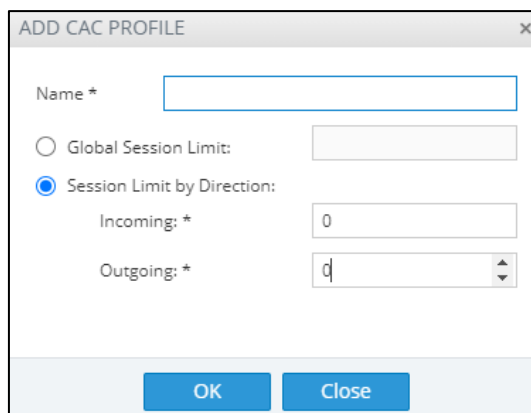
CAC Profiles can be defined in the ARM GUI under **Settings > Routing > CAC Profiles**. After selecting the **CAC Profiles** tab, the CAC Profile screen is displayed. The following actions are available:

- **Add**
- **Edit**
- **Delete**
- **Refresh**

CAC Profiles			
<div> <span>Add</span> <span>Edit</span> <span>Delete</span> <span>Refresh</span> </div>			
NAME	TOTAL LIMIT	INCOMING LIMIT	OUTGOING LIMIT
cac1	10		
demo_outgoing_limit			10
cac_profile	10		
AutoProfileName1619595981659	10		

To add a new CAC Profile definition, the operator selects the **Add** button and provides the following quota information:

- **Name** - mandatory - user defined unique name of the IP Profile  
One of the following:
- **Global Session Limit** – the limit on the total count of outgoing and incoming sessions  
-or-
- **Session Limit by Direction** – Limit by either or by both:
  - **Incoming** – Limit by the incoming sessions
  - **Outgoing** – Limit by the outgoing sessions



In the CAC Profiles page, the selected row (CAC Profile) can be edited using the **Edit** button. All settings can be edited and reapplied. If the CAC profile is edited (changed), the status of the network elements to which it is attached will be recalculated and appropriate alarms will be raised or cleared.

### 2.1.3.2 Defining a CAC Profile Threshold

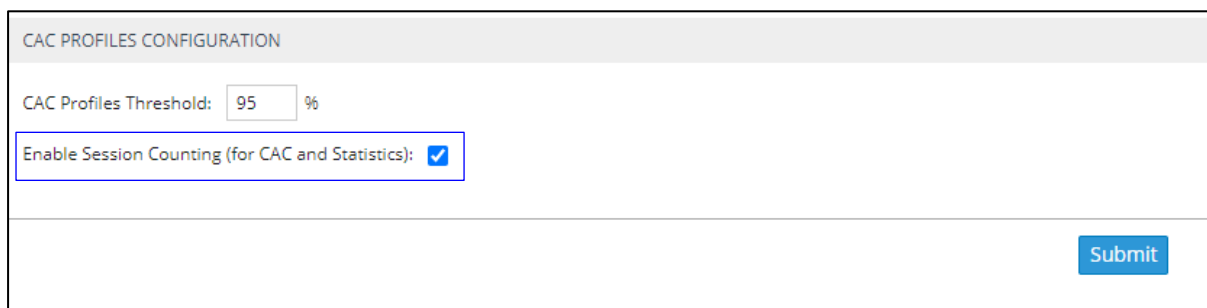
The ARM GUI lets operators adjust the threshold for generating a warning alarm, under **Settings > Routing > CAC Profiles**.



Note that the same CAC Profiles Threshold (percentage) value is applicable for all CAC Profiles defined in the ARM. To change the CAC profile, click **Submit**.

### 2.1.3.3 Disabling CAC and Session Counting

The ARM GUI lets operators disable CAC and Session Counting under **Settings > Routing > CAC Profiles**.



### 2.1.3.4 Attaching a CAC Profile to a Peer Connection

A CAC Profile can be attached to a Peer Connection. The same CAC Profile can be reused for multiple Peer Connections.

To attach a CAC Profile to a specific Peer Connection, the operator must select and **Edit** a specific Peer Connection either from the Network Map page or from the Peer Connections page.

From the 'CAC Profile' drop-down, the operator can select one of the previously defined profiles.

In the Peer Connections page (**Network > Peer Connections**), the CAC Profile is shown in the 'CAC Profile' column.

STATUS	NODE	NAME	VOIP PEER	IP GROUP	OPERATIVE STATE	ADMINISTRATIVE STATE	QUALITY	CALLS QUOTA	CAC PROFILE
✓	172.17...	IpGrp0	1_0	IpGrp0	✓	🔒	UNKNOWN		
✓	172.17...	IpGrp1	1_1	IpGrp1	✓	🔒	UNKNOWN		
✓	172.17...	C1619741350535	2_0	IpGrp0	✓	🔒	UNKNOWN		
🔒	172.17...	IpGrp1	2_1	IpGrp1	✓	🔒	UNKNOWN	manual_test	cac_profile
✗	172.17...	C1619945825937	1_2	IpGrp2	✗	🔒	UNKNOWN		

When the Peer Connections page is used, the operator can filter all Peer Connections using the same CAC Profile.

Search filters in the ARM GUI:

- Free Text:
- Operative State:
- Administrative State:
- Quality:
- Calls Quota:
- CAC Profile: 
  - cac1
  - test1
  - cac\_profile\_total\_limit\_name
  - cac\_profile\_outgoing\_limit\_name
  - demo\_outgoing\_limit
  - cac\_profile**
  - vpeer\_total\_limit
- MOS:
- ASR:

When selecting a Peer Connection with an attached IP Profile, information about the status of the CAC is also displayed.

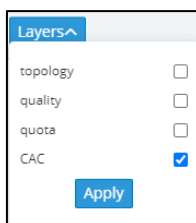
PEER CONNECTIONS SUMMARY

Name:	IpGrp1
Administrative State:	Locked
Operative State:	AVAILABLE
IPGroup Name:	IpGrp1
Weight:	50
Node name:	172.17.133.31-2
Peer connection type:	IPGroup
Quality:	UNKNOWN
MOS:	UNKNOWN
ASR:	UNKNOWN
Calls quota:	
Quota name:	manual_test
Calls duration (minutes):	0
Outgoing calls:	✓
Warning threshold reached:	No
Quota reached:	No
CAC Profile:	<b>cac_profile</b>
CAC State:	<b>UNBLOCK</b>

#### 2.1.3.4.1 Map Representation of CAC Status

The ARM GUI allows operators to view status information related to the CAC Profile of the Peer Connection, in the Network Map (and review which Peer Connections are blocked due to the CAC being reached).

To view the CAC status of the network, the operator selects the CAC layer in the ARM Network Map page. Note that it can be combined with other layers in the customer's network.



Blocked entities (due to CAC) are shown red. There's also an indication of direction, if relevant. Operators should be aware which layer is selected to correctly correlate the map colors (to understand that a red color is not due to Quality, for example).

#### 2.1.3.4.2 Peer Connection CAC Threshold Alarms

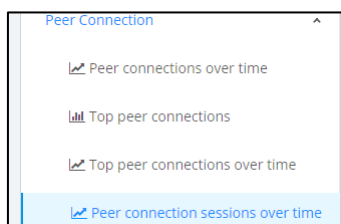
The ARM generates alarms when specified thresholds are crossed. The following severities are supported for CAC Profile related alarms:

- **Warning** – generated for a Peer Connection when the number of sessions reaches the threshold limit (as a percentage) defined under **Settings > Routing > CAC Profiles**.
- **Critical** – generated when the number of sessions reaches the defined session limit.
- **Clear** – Generated to clear 'set' alarms when the number of sessions drops under the defined limit or when the CAC Profile is detached.

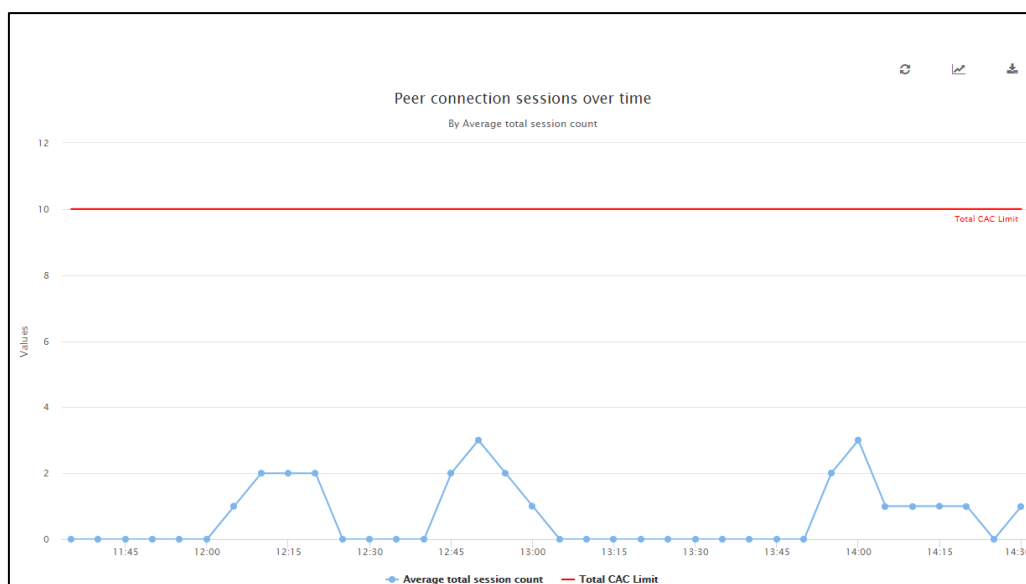
SEVERITY	DATE AND TIME	NAME	ALARM SOURCE	DESCRIPTION
■	05-Apr-21 14:16:59	CAC	Node#172.17.133.30-1/PeerConnection#lpGrp0	Peer Connection lpGrp0 total CAC is normal
■	05-Apr-21 14:16:59	CAC	Node#172.17.133.30-1/PeerConnection#lpGrp0	Peer Connection lpGrp0 total CAC has exceeded 95%
■	05-Apr-21 14:16:59	CAC	Node#172.17.133.30-1/PeerConnection#lpGrp0	Alarm with different severity was raised
■	05-Apr-21 14:16:43	CAC	Node#172.17.133.30-1/PeerConnection#lpGrp0	Peer Connection lpGrp0 total CAC has exceeded 100%

#### 2.1.3.4.3 Peer Connection Session Statistics

To provide visibility on the number of sessions per element, ARM 9.4 introduces a new statistic type 'Session count'. It was added to the three supported elements Peer Connection, VoIP Peer and Customers.



Following is an example of an 'Average' session count for a Peer Connection. Notice the Total CAC Limit which is only present if a CAC was attached to the element.



### 2.1.3.5 Attaching a CAC Profile to a VoIP Peer

A CAC Profile can be attached to a VoIP Peer. The same CAC Profile can be reused for multiple topology elements.

When attaching a CAC Profile to a VoIP Peer, the ARM counts all sessions of all Peer Connections connected to the VoIP Peer for both incoming and outgoing.

To attach a CAC profile to a VoIP Peer, the operator must select and **Edit** the VoIP Peer either from the Network Map or from the VoIP Peers page.

The 'EDIT VOIP PEER' dialog box shows the following fields:

- Name \*: 1\_0
- Peer Type: TEAMS
- CAC Profile: (highlighted with a red box)

Buttons: OK, Cancel

From the 'CAC Profile' drop-down, the operator can select one of the previously defined profiles.

In the VoIP Peers page (**Network > VoIP Peers**), the CAC Profile is shown in the 'CAC Profile' column.

NAME	TYPE	CAC PROFILE	CAC STATE	PEER CONNECTIONS
1_0	TEAMS	cac_profile	UNBLOCK	IpGrp0(172.17.133.30-1)
1_1	IP_PBX			IpGrp1(172.17.133.30-1)

When the VoIP Peers page is used, the operator can filter all VoIP Peers using the same CAC Profile.

The search filter dialog box includes the following fields:

- Enter search string: (empty)
- Name: (empty)
- CAC State: (dropdown)
- Peer Connections: (dropdown)
- CAC Profile: (dropdown)

Buttons: Search, Cancel

### 2.1.3.5.1 VoIP Peer CAC Threshold Alarms

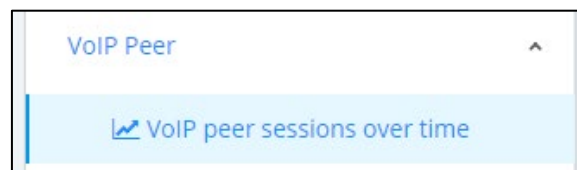
The ARM generates alarms when specified thresholds are crossed. The following severities are supported for CAC Profile related alarms:

- **Warning** – generated for VoIP Peers when the number of sessions reaches the threshold limit (as a percentage) defined in **Settings > Routing > CAC Profiles**.
- **Critical** – generated when the number of sessions reaches the defined session limit.
- **Clear** – generated to clear 'set' alarms when the number of sessions drops under the defined limit or when the CAC Profile is detached.

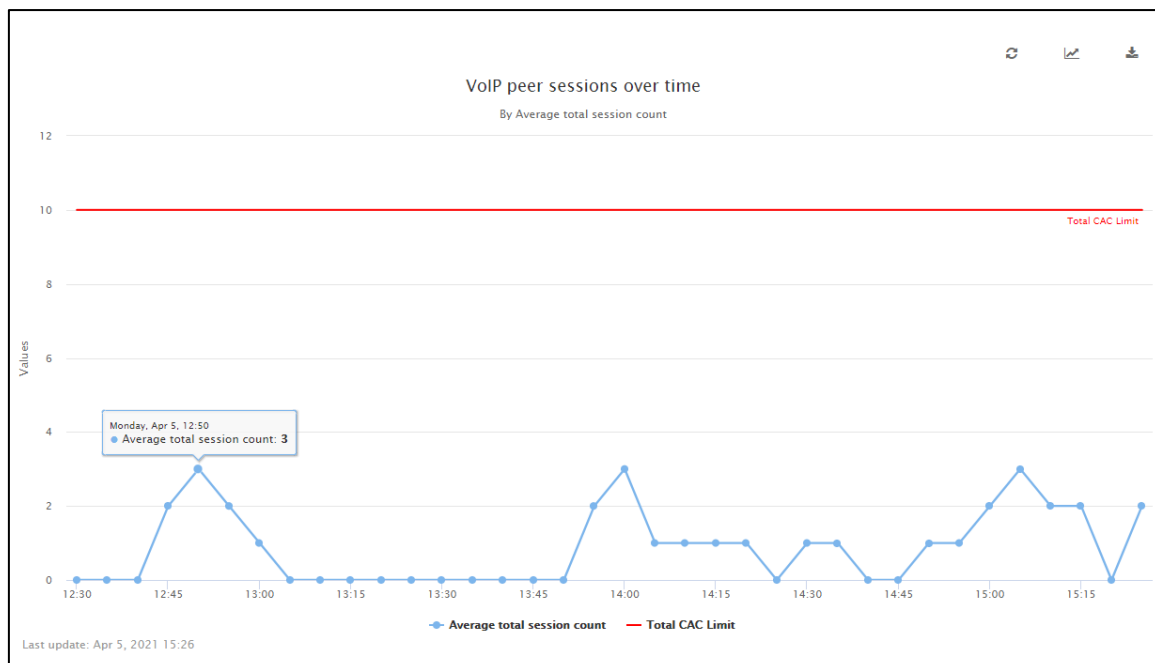
■	05-Apr-21 15:25:14	CAC	VoIP Peer#1_0	VoIP Peer 1_0 total CAC is normal
■	05-Apr-21 15:25:02	CAC	VoIP Peer#1_0	VoIP Peer 1_0 total CAC has exceeded 100%
■	05-Apr-21 15:25:02	CAC	VoIP Peer#1_0	Alarm with different severity was raised
■	05-Apr-21 15:25:02	CAC	VoIP Peer#1_0	VoIP Peer 1_0 total CAC has exceeded 95%

### 2.1.3.5.2 VoIP Peer Session Statistics

To provide visibility on the number of sessions per element, ARM 9.4 introduces a new statistic type 'Session count'. It was added to the three supported elements Peer Connection, VoIP Peer and Customers.



Following is an example of an 'Average' session count for a VoIP Peer. Notice the Total CAC Limit which is only present if a CAC was attached to the element.





### 2.1.4 Prefix Group Usage Visibility

ARM 9.4 provides greater visibility on using a Prefix Group for Routing. As deployment of the ARM has expanded, customer-managed dialing plans have grown more and more extensive (many Prefix Groups are being used in hundreds of Routing Rules and Policy Studio definitions). Sometimes, it's difficult to understand why a specific Routing Rule was selected by the ARM for Call Routing and where a specific Prefix Group is being used.

That's why in addition to the **Exact Match** DID search described in Section 2.2, ARM 9.4 adds a detailed description of the selected Prefix Group usage in ARM Routing. It covers both Policy Studio (pre-routing mechanism) and Routing Groups/Routing Rules.

When a Prefix Group is selected, its summary is displayed on the right side of the page:

>> PREFIX GROUP DETAILS

Name: test\_did

Type: NUMBER

Values: number1, number2

Policy studio

Used in policy studio: numberPS

Routing rule

Used in routing rules:

- customer\_test
- test\_prefix\_group
- NUMBER\_src
- NUMBER\_dst

If the selected Prefix Group is not used in Policy Studio, Policy Studio will be indicated as 'None'. The same applies to Routing Groups. If a Prefix Group is used in multiple Routing Groups, all of them will be listed.

## >> PREFIX GROUP DETAILS

Name: AG10

Type: PREFIX

Values: 101000905[1000-9999]#, 101000904[1000-9999]#, 101000193[1000-9999]#, 101000354[1000-9999]#, 101000245[1000-9999]#, 101000017[1000-9999]#, 101000336[1000-9999]#, 101000832[1000-9999]#, 101000522[1000-9999]#, 101000301[1000-9999]#, 101000679[1000-9999]#, 101000632[1000-9999]#, 101000932[1000-9999]#, 101000290[1000-9999]#, 101000697[1000-9999]#,...

Policy studio

Used in policy studio: None

Routing rule

Used in routing rules:

▼ Calls To Israel

Nati

▼ AttributeGroup1

routingAttributeGroup\_19

▼ AttributeGroup4

routingAttributeGroup\_49

▼ RG\_PHOENIX

RR\_PHX\_92PORTESDEF16

RR\_PHX\_92PORTESDEF15

## 2.2 New Engine for Validation of Prefix/DID Uniqueness

ARM 9.4 introduces new capability for validation of a prefix or a specific DID. As deployment of the ARM has expanded, customer-managed dialing plans have grown more and more extensive (many Prefix Groups with hundreds of prefixes, or complete phone numbers in a single group). Sometimes, it's difficult to preserve the uniqueness of a specific DID (or prefix) definition so operators may sometimes erroneously define Routing Rules with a specific prefix (or DID) but the same prefix (or DID) matches a different Prefix Group / Routing Rule.

The new engine gives operators the capability to validate if a specific DID (phone number) is part of an existing Prefix Group.

The validation/search engine is in the 'Prefix Groups' page in the ARM GUI (**Settings > Call Flow Configurations > Prefix Groups**).

Before version 9.4, the operator could search for the Name of a Prefix Group, filter its type and search for an exact string ('Value') if it appeared as part of the Prefix Group. This functionality is preserved when the 'Value' option is selected or a 'Search string' is provided.

ARM 9.4 adds the capability to select the **Exact Match** option in the Search filter to find all Prefix Groups that match the exact phone number.

Note that the **Exact Match** option finds a number even if it fits a 'range' or another pattern in the Prefix Group. In the following example, an **Exact Match** search was applied for DID **2121004811005** and it was found as part of Prefix Group **AG21** (for example) because it is in the range **212100481[1000-9999]#**.

Prefix Groups		
<div> Add Edit Delete Refresh </div> <div> <input type="text" value="match:2121004811005"/> </div>		
NAME	TYPE	VALUES
AG21	PREFIX	212100481[1000-9999]#, 212100285[10...
nonEmptyPG	PREFIX	[1-2], [2-3]
tt	PREFIX	1, 2, 3, 4, 5...

Note that searching in the **Search string** option doesn't search by **Exact Match** (only searching by name and value does).

Name:

Type:

☐ Value:

☒ Exact Match:

Search

Cancel

## 2.3 ARM Integration with Azure AD

Before Version 9.4, the ARM only supported LDAP and Microsoft Azure AD (Active Directory) *on-premises*.

With the ARM's expansion and massive deployment in the Azure environment, support for ARM integration with Azure AD became inevitable; customers who operate fully in an Azure cloud environment want to utilize Azure AD based on the Graph REST API (rather than LDAP).

ARM 9.4 introduces this functionality.

The functionality covers two aspects:

- Azure AD as source for users in the ARM
- Azure AD for operator authentication

### 2.3.1 Configuring the ARM in the Azure Portal

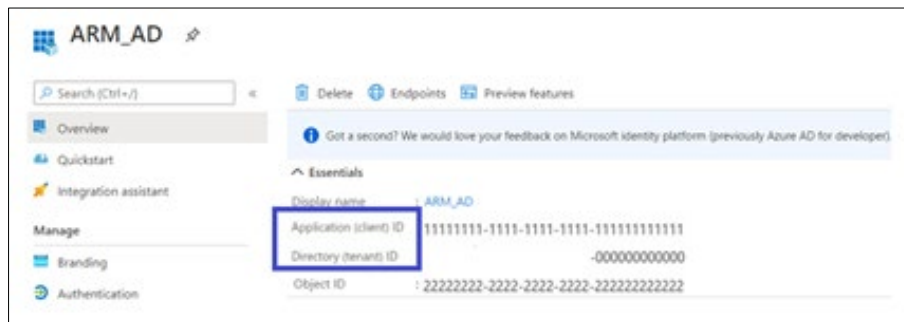
The following section is relevant for both Azure AD authentication and Azure AD users. To add the Azure AD to the ARM, operators must first register the ARM as an application and provide the ARM with the following information:

- Tenant ID
- Client ID
- Client secret

➤ **To configure the ARM in the Azure Portal:**

1. Register the ARM as an application; see the instructions under:  
<https://docs.microsoft.com/en-us/azure/active-directory/develop/quickstart-register-app#register-an-application>
2. Retrieve the **Client ID** and the **Tenant ID**.

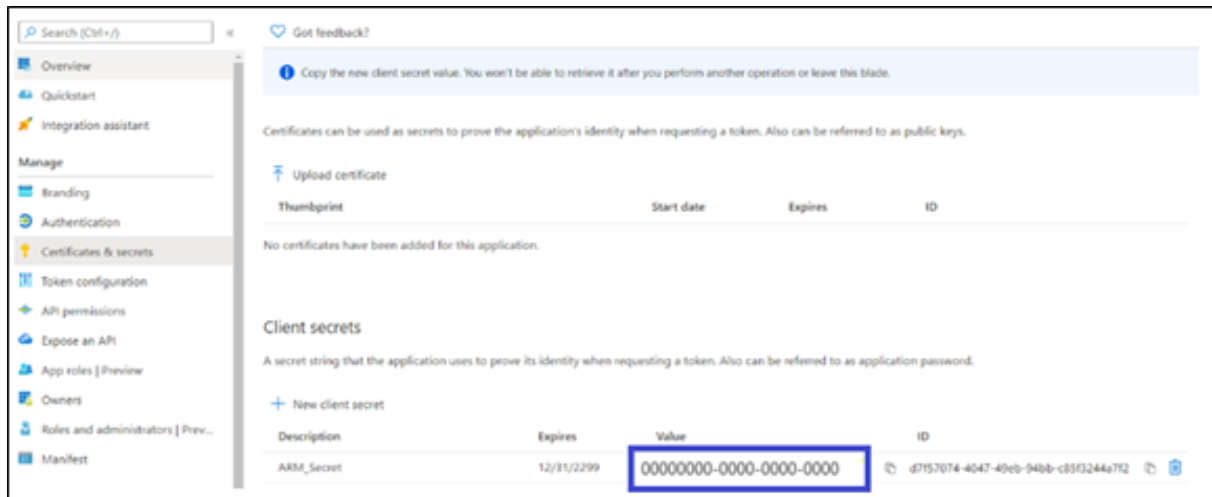
When registration finishes, the operator must provide the **Client ID** and **Tenant ID** displayed in the app registration's 'Overview' pane.



3. **Client secret**

The operator must:

- a. Create a client secret by clicking **New client secret**.
- b. Copy the client secret value (not the ID) to a safe location; it becomes visible immediately after creation; only then can it be copied; later, it's displayed with stars, e.g., **hsjfhj\*\*\*\*\*k** and cannot be copied.



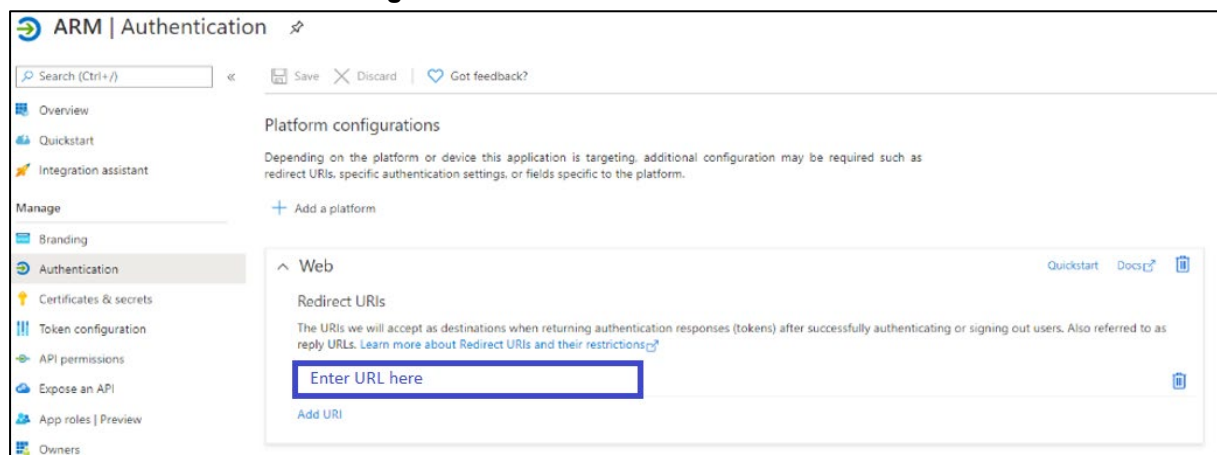
#### 4. Add Redirect URL

Relevant for Azure AD authentication and not for Azure AD users.

The operator must enter the ARM Redirect URL to the registered application in the Azure portal.

In your Azure AD:

- Under Manage: Authentication, click **add platform**.
- Choose **Web**.
- In 'Redirect URIs', enter the URL.
- Click **Configure**.



The format should be **https://{IP address/Hostname}/ARM/armui/login**

The selected communication method (IP address or hostname) must match the 'Communication method' configured in the ARM (under **Settings > Administration > Security** tab).

For simplicity, operators can just copy the Redirect URL from the **Settings > Administration > Azure Authentication** tab.

Note that any change made to the 'Communication method' setting (**Settings > Administration > Security**) will be automatically reflected in the Azure Redirect URL link. Make sure that the same is configured in the Azure AD.

## 5. API Permissions

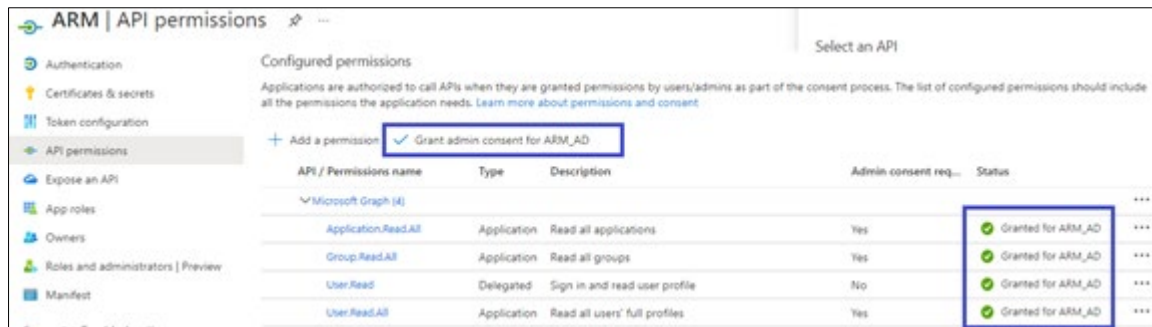
The ARM uses Microsoft's Graph API v.1.0 to retrieve a user's information and app roles. In your Azure AD, go to the **API permissions** tab and add the following permissions (of Microsoft Graph):

- **User.Read** (Delegated) – allows the ARM to sign in on behalf of the user and read the user profile.
- **Application.Read.All** (Application) – allows the ARM to retrieve all app roles in the Azure AD for the purpose of testing connectivity.

For AD users, operators must also add the following permission:

- **User.Read.All** (Application) - allows the ARM to retrieve all the users and their properties from Azure AD.
- **Group.Read.All** (Application) – allows the ARM to retrieve the user's membership groups.

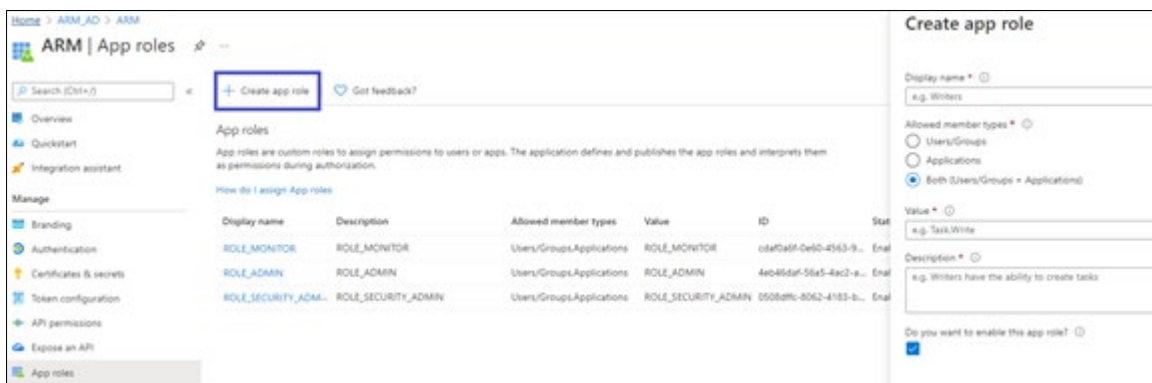
- Click **Grant admin consent** to enable these permissions.



## 6. Add app roles.

The operator must create app roles that will be mapped to ARM access roles – Security Admin, Admin and Monitor. In Azure Active Directory, under **Manage**, select **App registrations** and select the application you defined in the first step. Select **App roles | Preview** and then select **Create app role**. In the **Create app role** pane, enter the settings for the role.

- **Allowed member types** - Specifies whether this app role can be assigned to users, applications, or both. To support authentication via the REST API, both (**Users/Groups + Applications**) options should be selected, else select **Users/Groups**. AudioCodes recommends selecting the **Both** option which support authentication of both the REST API and the GUI.
- **Value** - Specifies the value of the roles claim that the application should expect in the token. This value should match the roles mapping in **Authorization level settings** in the ARM.

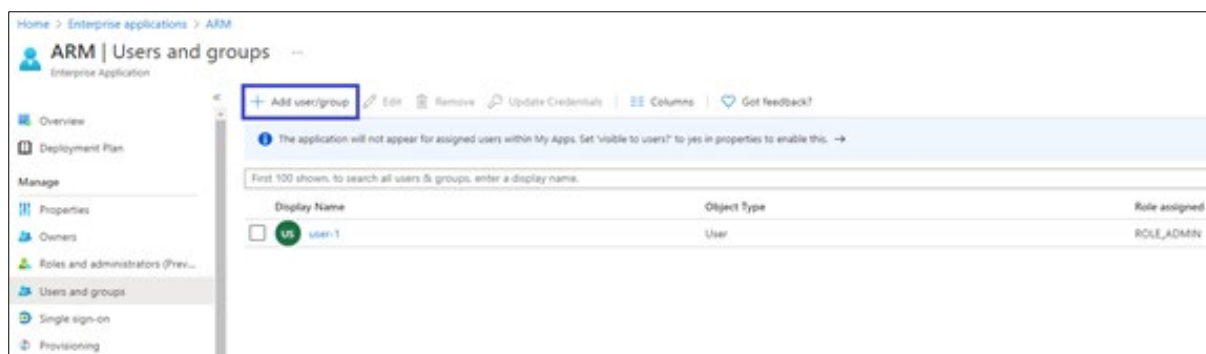


## 7. Assign users / groups to roles.

After you add app roles in your application, you can assign users and groups to the roles.

- In **Azure Active Directory** under **Manage** select **Enterprise applications** in the left-hand navigation menu.
- Select **All applications** to view a list of all your applications and then select the application in which you want to assign users or a security group to roles.
- Under **Manage** select **Users and groups**.
- Select **Add user/group** to open the **Add Assignment** pane.
- Select the **Users** or **groups** selector from the **Add Assignment** pane; a list of users and security groups is displayed.
- After you have selected users and groups, select the **Select** button to proceed.
- Select **Select a role** in the **Add assignment** pane; all the roles you defined for the application are displayed.
- Choose a role and select the **Select** button.
- Select the **Assign** button to finish the assignment of users and groups to the app.



**Note:**

- If you're using Azure B2C, adding app roles and assigning users / groups to roles is performed differently.
- Customers without Azure AD Premium cannot assign app roles to security groups. For these customers, app role assignment to users must be done individually by the administrator or an owner of the app.

More information about the app roles configuration and assignment is available here: <https://docs.microsoft.com/en-us/azure/active-directory/develop/howto-add-app-roles-in-azure-ad-apps>

### 2.3.2 Azure AD as a Source for Users in the ARM

ARM 9.4 provides the capability to access Azure AD natively and to add users from there; without interfacing Azure AD with the REST API, 'Azure AD Domain Services' was used as the interface between Azure AD and regular LDAP protocol to access users from Azure AD. The ARM uses Microsoft's Graph API v.1.0 to retrieve users and the groups in which they're members. These users are treated as regular users in the ARM and can be used for regular operations - such as Users Groups.

The ARM supports most Azure AD flavors such as B2C and to a certain extent B2B (due to limitations in Microsoft's Graph API, for example, B2C doesn't support mapping of the "memberOf" attribute).



**Note:** Operators cannot map Teams / Skype for business properties such as EnterpriseVoiceEnabled, OnPremLineURI, HostedVoiceMail, VoiceRoutingPolicy as they're currently not retrievable by Microsoft's Graph API.

To add Azure AD to the ARM, operators first need to register the ARM as an application in Azure (see Section 2.3.1) and provide the ARM with the following information:

- Tenant ID
- Client ID
- Client secret

Operators can also define parameters for the synchronization process, such as the frequency (in days), and the time.



**Note:** Due to limitations in Microsoft's Graph API, the ARM doesn't support regular synchronization (Delta) against Azure AD; only full synchronization is supported.

With support for Azure AD as a valid source of ARM users, the submenu 'LDAP Servers' under the **Users** menu changed to 'Servers':

USERS				
REGISTERED USERS	USERS GROUPS	<b>SERVICES</b>	FILE REPOSITORY	PROPERTY DICTIONARY
<div> Add Edit Delete Refresh </div>				
TYPE	STATUS	NAME	NUMBER OF USERS	LAST UPDATE
		Audiocodes_	1013	03-May-21 09:26:03
		Idap2016	250000	03-May-21 09:26:01
		AzureAD	1158	03-May-21 05:00:06

ACTIVE DIRECTORIES SUMMARY
Name: AzureAD
Type: Azure Active Directory
Status: available
Tenant Id: a0423ecb-b74b-4e4e-95f1-e56a13ee3ce9

When adding a new server, operators can select to add the LDAP server or the Azure AD:

USERS				
REGISTERED USERS	USERS GROUPS	<b>SERVICES</b>	FILE REPOSITORY	PROPERTY DICTIONARY
<div> Add Edit Delete Refresh </div>				
<div> LDAP Server </div>				
<div> Azure Active Directory </div>				
STATUS	NAME	NUMBER OF USERS	LAST UPDATE	
	Audiocodes_	1011	20-Apr-21 10:19:58	
	Idap2016	250000	20-Apr-21 10:19:56	
	AzureAD	1158	20-Apr-21 05:00:06	

Operators must provide information from Azure (as described in Section 2.3.1) and perform **Test connectivity**. The parameters under 'Updates' are related only to *full synchronization*.

AZURE AD SETTINGS

AZURE AD SETTINGS

AZURE AD PROPERTIES

GENERAL

Name: \*

AzureAD

Tenant Id: \*

a0423ecb-b74b-4e4e-95f1-e56a13ee3ce9

Client Id: \*

5ad1366c-994c-4ae0-8604-b4c0a051e192

Client Secret:

Page size:

999

Test connectivity

UPDATES

Perform full update every (days):

1

At:

3

0

Sync timeout (min):

60

Query Timeout (seconds):

120

OK

Cancel

When successfully connecting to the AD, operators will be able to map the local properties to the values from Azure AD; the 'Azure AD Properties' drop-down fields will display the relevant attributes from the Azure AD.

PROPERTY	LDAP MAPPING	ATTRIBUTE NORMALIZATION
Display Name	displayName x	ARMUI x
Phone	mail x	
Address	streetAddress x	
Mobile	mobilePhone x	0->+972 x
Country	country x	
Office Phone	businessPhones x	
2		
Physical address		
Emergency		
registration		
Origin		
MS.Lync.Line.URL		



**Note:**

- Most fields of the type 'User' resource are available for mapping.
- See the list in the following Microsoft documentation:  
<https://docs.microsoft.com/en-us/graph/api/resources/user?view=graph-rest-1.0>

### 2.3.3 Azure AD for Operators Authentication

In addition to support for Azure AD as a source of ARM users, ARM 9.4 also supports Azure AD for operator login authentication. This new feature augments local operator login authentication supported in previous ARM releases and comes in addition to LDAP and RADIUS authentication.

The Azure portal must firstly be configured to allow the ARM as a valid application as described in Section 2.3.1.

Azure AD is added to the ARM in a new 'Azure Authentication' page (**Settings > Administration > Azure Authentication**).

Only operators with a security level of 'Security Admin' can edit Azure authentication attributes.

ARM Version 9.4 also features the capability to test connectivity with Azure AD (using the **Test** button shown in the preceding figure) (available for operators whose security level is Admin or Secure Admin).

Under the section 'Authorization Level Settings' in the page, the operator also provides mapping of the ARM's access roles ('Security Admin', 'Admin' and 'Monitor') with the Azure AD's app roles.

In the connectivity test, the ARM also validates the Authorization-level mappings; if an Azure AD membership group does not contain the authorization mappings, a warning message is displayed. After Azure authentication is enabled, the following button is displayed in the login screen:

When selecting **Sign in with Microsoft**, the browser redirects to the Microsoft login page and after authentication with Microsoft, it redirects back to the ARM GUI.

## 2.3.4 Azure AD for REST Requests Authentication

Operators who operate the ARM using the official ARM REST API can also use Azure AD for authentication.

To use the ARM REST API with an Azure AD user, operators must follow these steps:

1. Configuration in Azure portal

In Azure Active Directory under **Manage** select **App registrations**, select the default ARM application. Under **Manage** select **Expose an API**:

- a. Click **Add a scope**
- b. Click **Save and continue**; the default value will be created: "api://{client-id}".

Then register your own REST application for REST authentication.

In the **Azure Active Directory** pane click **App registrations** and choose **New registration**.

In the new application:

- a. Create a client secret – as described previously.
- b. Add permission to access the default ARM application:  
Under **API permissions** click **Add permission**.  
Then select **my APIs**, select **application** and then select the exposed API previously defined in the app and select the role for the REST authentication (from the app roles defined previously in the application).  
Then click on **Grant admin consent**.

## 2. Acquire access token from Microsoft

To acquire access token from Microsoft using REST client:

Send a request to Microsoft Identity platform's token endpoint, as follows:

```
POST
https://login.microsoftonline.com/<tenant-id>/oauth2/v2.0/token
```

Using **x-www-form-urlencoded** as 'Body content type' and the following 'Body' content:

```
grant_type=client_credentials&
client_id=<rest-app-client-id>&
client_secret=<rest-app-client-secret>&
scope = api://<client-id>/.default
```

Replace **tenant-id** and **client-id** with **tenant id** and **client id** of the default ARM application.

Replace **rest-app-client-id** and **rest-app-client-secret** with the **client id** and **client secret** of your own REST application.

A successful response will contain an access token:

```
{
  "token_type": "Bearer",
  "expires_in": 3599,
  "ext_expires_in": 3599,
  "access_token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsI..."
}
```

## 3. Access ARM's REST API using an access token:

To access ARM's REST API using the access token, send a Post request with the token received from Microsoft to:

```
POST
<ARM_Configurator_IP>/ARM/v1/login/microsoft/authentication/to
ken
```

with the following body:

```
{
  accessToken: String,
  authenticationType: ACCESS_TOKEN
}
```

The ARM will validate the Microsoft access token and will generate an ARM token with the received role.

8. In any REST Request to the ARM, use the received token in the authorization Header like this:

Header name	Header value
authorization	Bearer {token}

### 2.3.5 Revoking Azure User Tokens

Operators with a security level of 'Security Admin' can revoke all tokens created for Azure AD users.

To revoke all tokens, the following REST request must be sent:

```
DELETE <ARM_Configurator_IP>/ARM/v1/security/authentication/token/revoke
```

## 2.4 Appending | Deleting Prefixes in a Prefix Group via the REST API

For operators using ARM REST API-based scripting for managing prefixes in a Prefix Group, ARM 9.4 extends these capabilities.

ARM 9.4 adds an easy way to append or delete a prefix (or group of prefixes) to an existing Prefix Group. This feature does not affect the ARM GUI but makes updates via the REST API much more convenient.

PATCH <ARM\_Configurator\_IP>/ARM/v1/routing/attributeGroup/{id}

with the following JSON:

	<pre>{   "add": [ String ],   "remove": [ String ] }</pre>	Add and remove values
--	--	-----------------------

Response status:

- **200 OK** if no error was thrown
- **409 Conflict** if an error occurred

## 2.5 VoIP Peers Page

ARM 9.4 adds support for the Topology VoIP Peers page located under **Network > VoIP Peers** (in addition to the Peer Connections and the Connections pages).

NAME	TYPE	CAC PROFILE	CAC STATE	PEER CONNECTIONS
Pcon-1,10.7.12.114_VoIPPeer	NA			Pcon-1(10.7.12.114)
MDR_10.7.12.114_VoIPPeer	NA			MDR(10.7.12.114)

The page became essential when capability was introduced to associate a Topology element with a CAC Profile. Now, when a CAC Profile can be attached to a VoIP Peer, it's much more convenient to manage it in the VoIP Peers page.

In the page, operators can apply **Edit**, **Delete** or **Refresh** actions.

The following information is available per VOIP Peer:

**Name** (Editable), **Type** (Editable), **CASC Profile** (can be attached by the operator), **CAC State** (available as read-only if a CAC Profile is attached), **Peer Connections** (list of associated Peer Connections).

Operators can apply the following search and filters on the VoIP Peers table in the page:

Name:

CAC State:

Peer Connections:

CAC Profile:

It's useful to filter, for example, all VOIP Peers with a specific CAC Profile attached, or to filter VoIP Peers blocked due to an attached CAC Profile.

## 2.6 Customized ARM Connection (IP Group Name, User-Defined IP Profile & Media Realm)

When adding / editing an ARM Connection, the operator can select and configure a corresponding name of an IP Group for each node.

Default 'Name' options are taken from the SOURCE and DESTINATION interface IDs, for example, **ARM\_2.7\_3.9**.

Elements displayed in the 'IP Profile' | 'Media realm' drop-downs are those that are used by or created by the Routing Server in the SBC.

ADD CONNECTION

Name: \*

Weight:

50

Transport Type:

TCP

Node 1

Node 2

Node: \*

172.17.133.30-1

×

▼

172.17.133.31-2

×

▼

Routing Interface: \*

SIP-0

×

▼

SIP-0

×

▼

Name: \*

ARM\_2.7\_3.9

×

▼

ARM\_3.9\_2.7

×

▼

Ip Profile: \*

ARM\_IP\_Profile

×

▼

ARM\_IP\_Profile

×

▼

Media realm:

▼

▼

SIP Group name: \*

Advanced Conditions

☒ Keep connection properties synchronized

☒ use global quality definitions

☐ use specific quality definitions

☐ MOS
☐ ASR

OK

Cancel



**Note:**

- If one of the IP Profile names exists in the SBC when adding or editing a connection, the connection will fail to be created.
- IP Profile and Media Realm are available from SBC versions 7.20A.258-0313, 7.20A.260-180 and 7.40A.005.



## 2.7 Authentication Order

ARM 9.4 introduces the capability to manage the *authentication order* for LDAP and RADIUS authentication.

Operators can define whether the ARM will first check the external service (LDAP or RADIUS), or the local database (the Operators table); the default behavior is to first check the external service.

Change can be applied for each authentication method, depending on which one is used, in the following path:

**Settings > Administration > LDAP Authentication**

**Settings > Administration > RADIUS Authentication**



The screenshot shows a web interface for configuring authentication. At the top, there is a header bar labeled "AUTHENTICATION MODE". Below this, on the left, is the label "Authentication Order:". To the right of this label is a dropdown menu. The dropdown menu is currently open, showing three options: "External first" (which is highlighted with a blue background), "External first", and "Local first".

This page is intentionally left blank.

## 3 Supported Platforms

ARM 9.4 supports the platforms shown in the table below.

**Table 3-1: ARM 9.4 Supported Platforms**

ARM	Platform	Application
GUI	Web Browser	Firefox, Chrome, Edge
Deployment	VMWare	VMware ESXI 6.5, 6.7
	HyperV	Windows Server 2016 Hyper-V Manager Microsoft Corporation Version: 10.0.14393.0

This page is intentionally left blank.

## 4 Earliest SBC/GW Software Versions Supported by ARM Features

Some ARM features are developed in coordination with nodes (AudioCodes' SBCs and Media Gateways). To activate and use an ARM feature, the node needs to be upgraded to the earliest software supporting that feature if it's configured with software that does not support it.

The following table displays ARM features supported by the earliest node software.

**Table 4-1: ARM Features Supported by the Earliest Node Software**

#	Feature	Earliest Node Software Supporting It	Comments
1	Quality-based routing	Version 7.2.158 and later	The quality-based routing feature is not supported when operating with nodes version 7.0 (for Mediant 3000).
2	Separate interface at the node level for ARM traffic	Version 7.2.158 and later	The capability to configure a separate interface at the node level for ARM traffic is not supported when operating with nodes earlier than version 7.2.154 (for Mediant 3000).
3	Call preemption	Version 7.2.158 and later	The call preemption for emergency calls feature is not supported when operating with nodes version 7.20A.154.044 or earlier (not applicable for Mediant 3000).
4	Number Privacy	Version 7.2.250 or later	-
5	Support of IP Group of type User without 'dummy' IP	7.20A.250 and later	Network administrators who want to use a node's IP Group of type 'User' as the ARM Peer Connection can avoid configuring a dummy IP Profile if using node version 7.20A.250 and later.  Customers who use ARM version 8.4 with node version earlier than 7.2.250 and who want to configure an IP Group of type 'User' as the ARM Peer Connection, must configure a dummy IP Profile (with a dummy IP address) at the node level, to be associated with this IP Group.
6	Support of ARM Routers group and policies.	Version 7.20A.240 or later	-
7	Support of ARM Routed Calls/CDRs representation	Version 7.20A.250.205 or later	-
8	Support of Forking in ARM (SBC only)	Version 7.20A.252 or later	-
9	Support for Registered users in ARM	Version 7.20A.254.353 or later	-
10	Support for combined ARM and	Version 7.20A.256.391	Supported for SBC only

#	Feature	Earliest Node Software Supporting It	Comments
	SIP based Routing decision (Route based on Request URI)		
11	Support for combined ARM and SBC Routing decision	Version 7.20A.256.391	Supported for SBC only
12	ARM as an Information Source for Users Credentials	Version 7.20A.256.713	Supported for SBC only
13	Support for Microsoft Teams LMP (Local Media Optimization) and additional IP Profiles	Versions: 7.20A.258 -0313, 7.20A.260-180 7.40A.005 (official release) and later	-
14	ARM connection with ABC level defined IP Profile and Media Realm	Versions: 7.20A.258 -0313, 7.20A.260-180 7.40A.005 (official release) and later	SBC only
15	ARM 'Customer' entity (Team multi-tenancy) - support for Contact header manipulation	7.40A.005.509 or later	

## 5 Resolved Issues in ARM 9.4

The table below lists major issues which were encountered by customers in previous releases but which are resolved in ARM 9.4.

**Table 5-1: Resolved Issues in ARM 9.4**

Incident	Problem / Limitation
ARM-4613	When a rule is deleted, the ARM always jumps back to the top Routing Group.
ARM-4555	'Session count over time' statistics are no longer being displayed.
ARM-4551	The ARM doesn't send a notification when a rule is matched for some rules.
ARM-4548	After the upgrade of the Mediant 9000 SBC, calls from DELEJ-PA-17100015 are not established.
ARM-4534 ARM-4441	The Termination Reason in the ARM doesn't match the OVOC for cancelled calls.
ARM-4500	The ARM no longer sends a notification when the Routing Rule is hit.
ARM-3577	The ARM's upgrade mechanism has been improved.
ARM-4477	ARM Registered User Routing is not functioning for Mediant 800 SBCs.
ARM-4442	A hyphen cannot be used in the 'SIP Group Name' when editing a connection.
ARM-4426	FQDN including an underscore needs to be accepted.

This page is intentionally left blank.



## 6 Tested ARM Capacities

Table 6-1 lists tested ARM capacities. The table presents the results of *the maximum capacities* tested. If customers require *higher capacities* tested, they should communicate this to AudioCodes.

**Table 6-1: Tested ARM Capacities**

Item	Maximum Capacity Tested
CAPs (assuming the average call duration is 100 seconds)	300 CAPs per ARM Router
	ARM total: 3,000 CAPs
ARM Routers	40
Routing Groups	2,000
Routing Rules per ARM	10,000
ARM Users (either local or LDAP/Azure AD)	1 million Possible extension to 4 million when ordering a special Feature Key. Requires 16 GB memory for Routers.
'Customer' entities (Teams tenants)	Up to 20,000
Nodes number	40
Peer Connections	Per Node: 600
	ARM total: 1,000
Connections	200
Prefix Groups	2,000
Prefixes in a single Prefix Group	2,000
Calls history	10 million
Statistics history	30 days

This page is intentionally left blank.

## 7 Known Limitations and Workarounds

The table below lists the known limitations and workarounds in ARM 9.4.

**Table 7-1: Known Limitations and Workarounds**

Incident	Problem / Limitation	Comments/Workaround
-	Attaching / detaching a user to / from an Active Directory Group is reflected in the ARM's Users page (and Users Groups page) only after performing a full update (synchronization) with the LDAP server (by default performed automatically every 24 hours).	Network administrators should take this into consideration
-	When defining a Users Group, the condition is applied to the pre-manipulated value of the property used in the condition definition (the original value taken from the Active Directory).	Network administrators should take this into consideration
	For VMware users, after rebooting or upgrading an ARM Configurator, its clock 'drifts'. This can sometimes cause inconsistency between ARM Configurator and ARM Router data.	Make sure the clock in the machine (Host) and the VM (Guest) are the same. Both should be synchronized with the same NTP.
-	For customers who use auto-detect mode to add a new node (SBC / gateway) to the ARM, the name of the Configurator Web service configured at the node level for auto-discovery <i>must</i> be <b>ARMTopology</b> else the ARM data center recovery mechanism will not work correctly for the node; it will not be redirected to the new Configurator.	Generally, it's preferable to add a node using the ARM GUI rather than auto-detection.
-	When the ARM is used with Load Balancing CE SBC in an Azure environment, the operator should make sure to define the FQDN / IP Address as the Hostname of the LB CE SBC, and add the LB CE SBC in the ARM using that Hostname.	
<b>Breaking changes</b>		
-	ARM 9.4 does not support 'Build Star' and 'Build Mash' capabilities. These capabilities were removed from the GUI and REST API as they are not widely used by customers and are potentially problematic.	Operators should add Connections and build the ARM Network Topology based on customer requirements.
-	For operators of the pre-9.2 ARM version: ARM 9.2 changes the REST API for ARM Users management (Add, Delete, Modify) in a way that is not backward compatible.	Customers must take this into consideration. The new REST API for users is described in the ARM 9.2 and the <i>ARM 9.4 REST API Developer's Guide</i> . If customers develop scripts based on this REST API, these scripts should be adjusted

Incident	Problem / Limitation	Comments/Workaround
		to the new REST API when moving to ARM 9.2 or ARM 9.4.
-	ARM 9.4 changes the REST API for getting all VoIP Peers (VoIP Peers GET API). This non-backward compatible change was implemented to support Paging.	Customers should take this into consideration. The new REST API for getting the VoIP Peers is described in the <i>ARM 9.4 REST API Developer's Guide</i> . If customers develop scripts based on this REST API, these scripts should be adjusted to the new REST API when moving to ARM 9.4.
-	For a two-step upgrade (for customers upgrading from ARM 8.6 or earlier): The redesigned ARM 8.8 Add Routing Rule – Routing Actions screen does not feature the 'via' action as previous versions did. The same applies to ARM 9.0.	Customers upgrading from a previous version will still view the action but are advised to exclude it from routing definitions.
-	In ARM 9.4, when an alarm for a Routing Rule is generated, the detailed alarm information is placed in both <b>Additional Info 1</b> and <b>Additional Info 2</b> .	Operators should use information from both fields. This is done to provide detailed information about the alarm without truncation.
<b>Upgrade</b>		
-	Direct upgrade from ARM 8.6 and earlier to ARM 9.4 is not supported.	For these cases, a two-step upgrade is required: Step 1: Upgrade to ARM 9.0 or ARM 9.2 Step 2: Upgrade to ARM 9.4 <b>Note:</b> The following direct upgrades are supported: <ul style="list-style-type: none"><li>• ARM 8.8 &gt; ARM 9.4</li><li>• ARM 9.0 &gt; ARM 9.4</li><li>• ARM 9.2 &gt; ARM 9.4</li></ul>
-	For pre-ARM 9.2 deployments, the upgrade to ARM 9.4 is not a regular upgrade as it upgrades the OS of all components to CentOS8 (first version with CentOS8 is ARM 9.2).	Make the following preparations: <ul style="list-style-type: none"><li>• Make sure you downloaded not only the upgrade but also the installation images for the ARM Configurator and the ARM Router (not as for the usual upgrade).</li><li>• Request from AudioCodes a Feature Key with all the ordered features and ordered number of sessions for the new VM in ARM 9.4.</li><li>• Prepare temporary IP and VM resources required for each server upgrade.</li><li>• Prepare extended storage for the ARM Configurator (the ARM Configurator allocates 80 GB in ARM 9.4).</li></ul>
-	To upgrade to ARM 9.4 in a VMware environment, the customer must have VMware ESXI 6.5, 6.7 (earlier versions are not supported with CentOS 8).	-

Incident	Problem / Limitation	Comments/Workaround
-	For a two-step upgrade (for customers performing an upgrade from ARM 8.6 and earlier): Upgrading from ARM 8.6 to ARM 8.8/9.0 does not preserve calls (CDRs) information on calls run by ARM 8.6. Note that upgrading from ARM 8.8/ARM 9.0, ARM 9.2 to ARM 9.4 preserves calls information during the upgrade.	If a customer needs calls information from ARM 8.6, contact AudioCodes support (R&D) for the procedure to back up calls (CDRs) information.
-	Miscellaneous issues with the ARM GUI after upgrading from previous releases.	Customers are requested to clear the browser cache after performing a software upgrade ( <b>Ctrl+F5</b> ).
<b>GUI Incidents</b>		
ARM-3249 ARM - 2724	Prefixes in a Prefix Group cannot be edited. Double-clicking an existing prefix to modify it doesn't work.	The customer can remove the old prefix and define a new prefix.
ARM-4528	In the <b>Alarms &gt; Journal</b> , the calls Quota Name is not shown in the 'Description'.	-
ARM-4699	In the VoIP Peers page ( <b>Network &gt; VoIP Peers</b> ), the column indicating Peer Connections is populated only after 'Refresh'.	Operators must take this into consideration.
-	Basic ARM operational statistics - either at the ARM level or for a specific Network Topology element - are displayed in the right pane of <b>Network &gt; Map</b> under 'General Statistics'. Opening these graphs as a separate popup window does not shows the statistic graph (blank).	Zooming into the details of a specific statistic is available in the ARM from the <b>Statistics</b> menu.
<b>ARM in Azure with SBCs behind Load Balancer</b>		
ARM-4676	After a switchover of an SBC occurs, the node can temporally (for few seconds) switch between available and unknown state in the ARM; calls are unaffected as routing continues regularly.	The issue occurs as it takes time for the Load Balancer (usually up to 10 seconds) to switch to the secondary SBC.
ARM-4676	After a switchover of an SBC occurs, the connections to the HA SBC are indicated for a few minutes as unavailable.	The connection between the HA SBCs behind the Load Balancer and the other nodes should have <b>Keep connection properties synchronized</b> disabled. Also, the IP of the proxy set towards the node behind the Load Balancer should be configured manually (at the SBC level) with the Load Balancer's IP.

**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/>

©2021 AudioCodes Ltd. All rights reserved. AudioCodes, AC, HD VoIP, HD VoIP Sounds Better, IPmedia, Mediant, MediaPack, What's Inside Matters, OSN, SmartTAP, User Management Pack, VMAS, VolPerfect, VolPerfectHD, Your Gateway To VoIP, 3GX, VocaNom, AudioCodes One Voice, AudioCodes Meeting Insights, AudioCodes Room Experience and CloudBond are trademarks or registered trademarks of AudioCodes Limited. All other products or trademarks are property of their respective owners. Product specifications are subject to change without notice.

LTRT-41953

