Release Notes

AudioCodes One Voice™ Operations Center

AudioCodes Routing Manager (ARM)

Version 9.8.200 Fix 1

Table of Contents

List of Tables

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: December-19-2023

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 Revision Record

LTRT	Description
41959	SecureLogix – Hosted Call. Custom REST API Request. Preparing and Sending the Request. Using the Response. Adding UMP Server. New Actions in File Repository. Scheduling Synchronization for Each File Repository. DIDs Count. Configuring DIDs Count. Reroute Peer Connection for REFER and 3XX Requests.
42360	[9.8.200 Fix 1] Resolved Issues

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 9.8.200 Fix 1 of the AudioCodes Routing Manager (ARM).

1.1 Managed AudioCodes Devices

ARM 9.8.200 Fix 1 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 9.8.200 Fix 1

Device	Major Versions
MP-1288 Gateway	7.20A.258.119 and later
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 9.8.200 Fix 1

This section covers the new features and capabilities introduced in ARM 9.8.200 Fix 1.

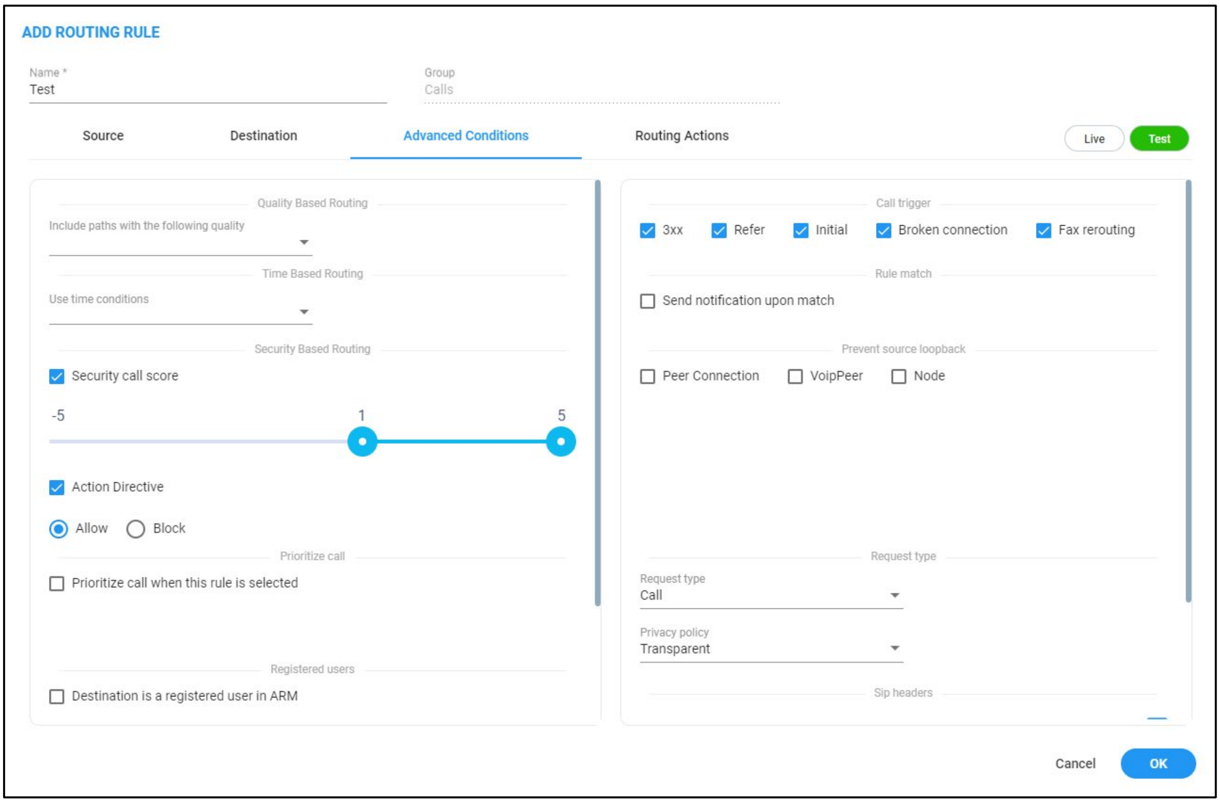
2.1 SecureLogix – Hosted Call

Security-based routing can be applied to calls that receive an **Action Directive** from SecureLogix's Orchestra One as part of the pre-routing process.

Security-based routing is applied as part of the ARM Routing Rule and must first be enabled when editing the Routing Rule in the 'Advanced Conditions' tab settings.

The options for **Action Directive** are **Allow** and **Block.**

When enabled, the Routing Rule will use the 'action directive' value returned from SecureLogix as part of the match. If no 'action directive' is returned from SecureLogix or the 'action directive' value doesn't match the **Action Directive** selection, the rule will not be matched.



ADD ROUTING RULE

Name * Test

Group Calls

Quality Based Routing

Include paths with the following quality

Time Based Routing

Use time conditions

Security Based Routing

☑ Security call score

-5 1 5

☑ Action Directive

● Allow ○ Block

Prioritize call

□ Prioritize call when this rule is selected

Registered users

□ Destination is a registered user in ARM

Call trigger

☑ 3xx ☑ Refer ☑ Initial ☑ Broken connection ☑ Fax rerouting

Rule match

□ Send notification upon match

Prevent source loopback

□ Peer Connection □ VoipPeer □ Node

Request type

Request type Call

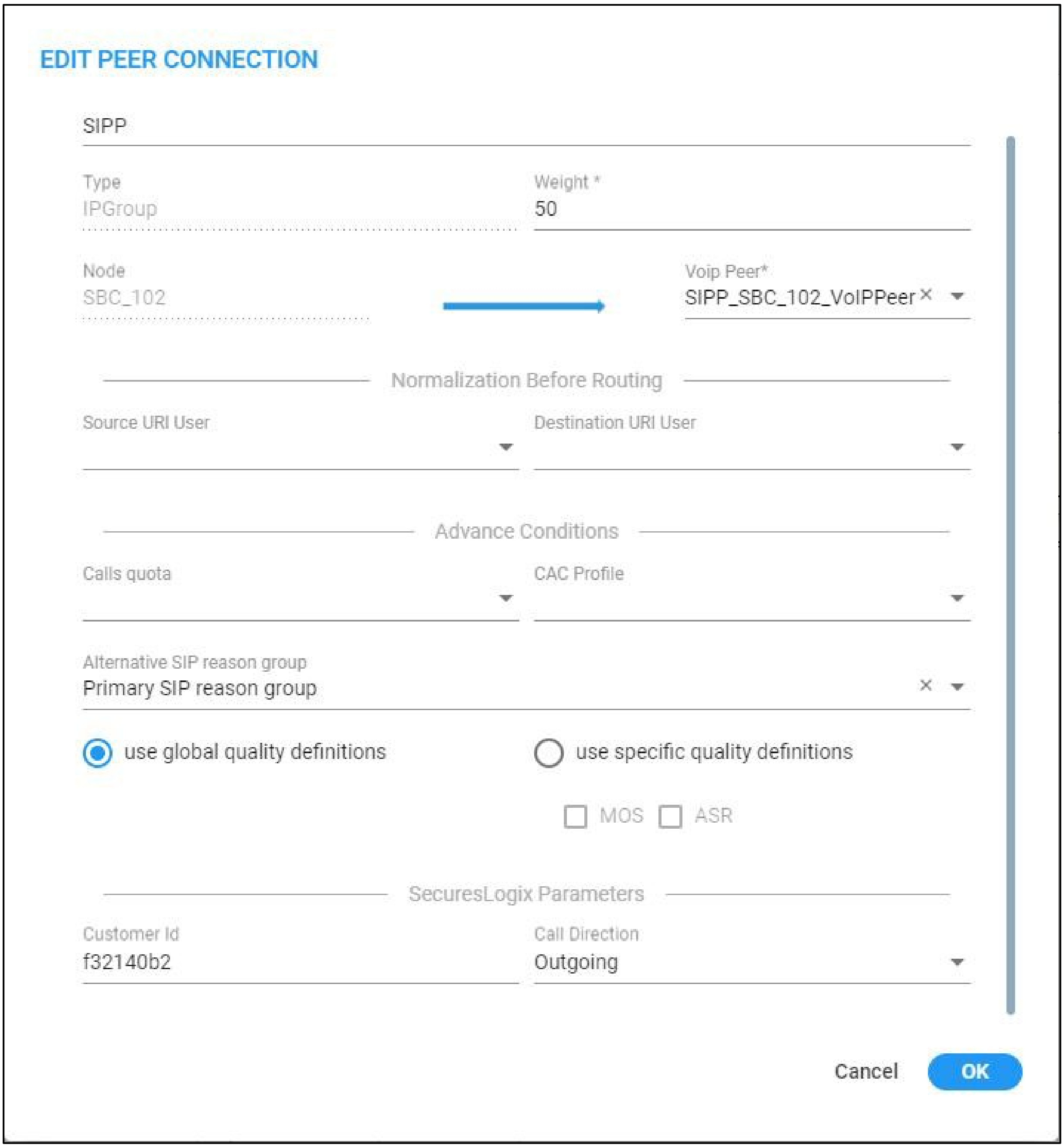
Privacy policy Transparent

Sip headers

The ARM query to SecureLogix's Orchestra One may include new attributes:

- Customer ID
- Call Direction

The operator is able to configure these attributes by editing a Peer Connection.



EDIT PEER CONNECTION

SIPP

Type: IPGroup

Weight *: 50

Node: SBC_102

Voip Peer*: SIPP_SBC_102_VoIPPeer

Normalization Before Routing

Source URI User:

Destination URI User:

Advance Conditions

Calls quota:

CAC Profile:

Alternative SIP reason group: Primary SIP reason group

● use global quality definitions ○ use specific quality definitions

□ MOS □ ASR

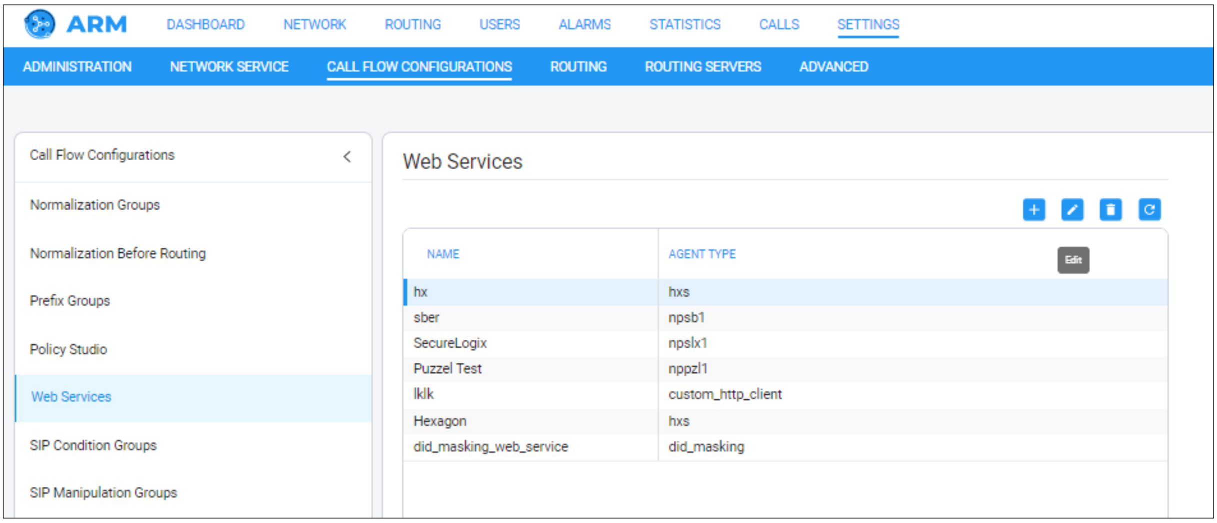
SecuresLogix Parameters

Customer Id: f32140b2

Call Direction: Outgoing

2.2 Custom REST API Request

The Web Services page in ARM 9.8.200 Fix 1 (**Settings** > **Call Flow Configurations** > **Web Services**) enables operators to define a ‘custom’ web service.

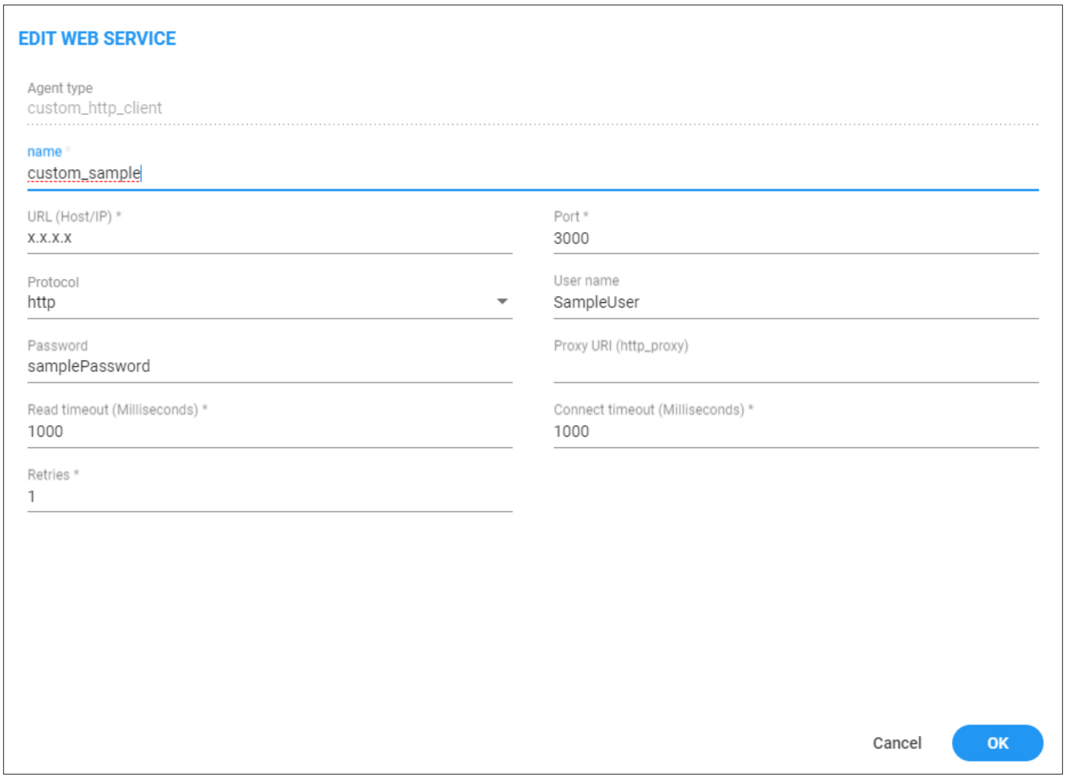


Web Services

NAME	AGENT TYPE
hx	hxs
sber	npsb1
SecureLogix	npslx1
Puzzel Test	nppzl1
lklk	custom_http_client
Hexagon	hxs
did_masking_web_service	did_masking

With the help of a SIP Manipulation Group, you can use a ‘custom’ web service and send a GET/POST/PUT/PATCH REST API request.

The figure below shows the new ARM screen in which operators can add | edit a ‘custom’ web service.



EDIT WEB SERVICE

Agent type: custom_http_client

name *: custom_sample

URL (Host/IP) *: x.x.x.x

Port *: 3000

Protocol: http

User name: SampleUser

Password: samplePassword

Proxy URI (http_proxy):

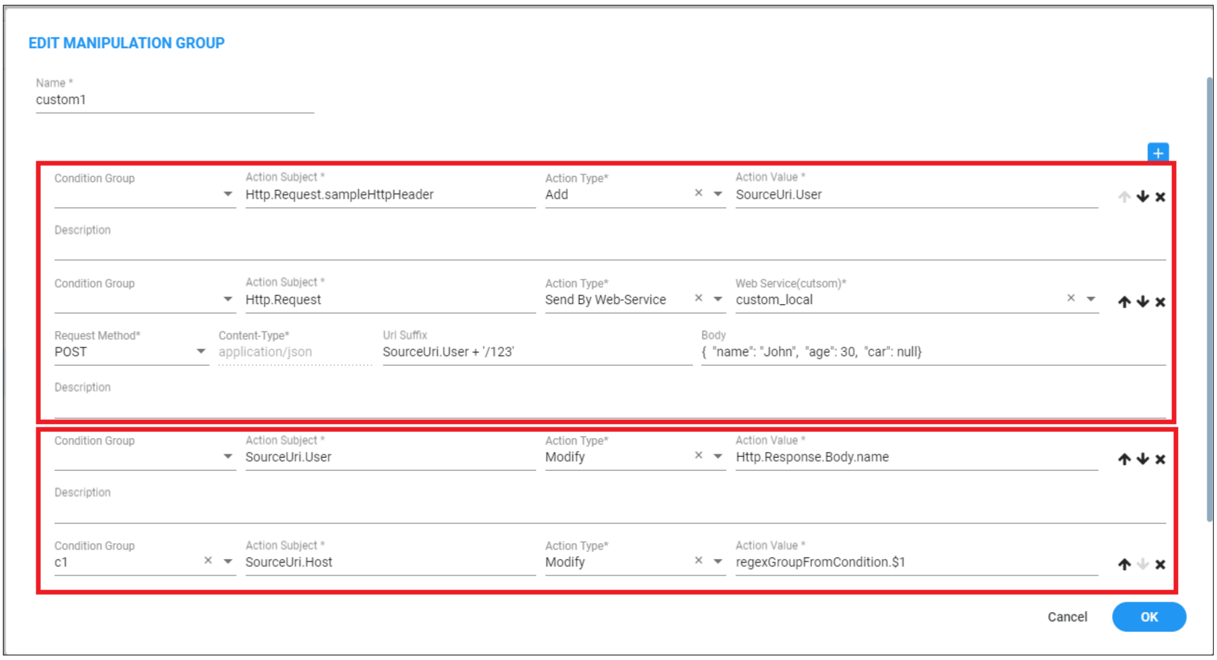
Read timeout (Milliseconds) *: 1000

Connect timeout (Milliseconds) *: 1000

Retries *: 1

Using a ‘Custom HTTP Client’ type web service in the ‘Edit Manipulation Group’ page shown in the figure below:

- The uppermost screen section indicated in the figure below enables operators to prepare the request and its sending. See Preparing and Sending the Request below.
- The lowermost screen section indicated in the figure below enables operators to use the response. See Using the Response below



EDIT MANIPULATION GROUP

Name *: custom1

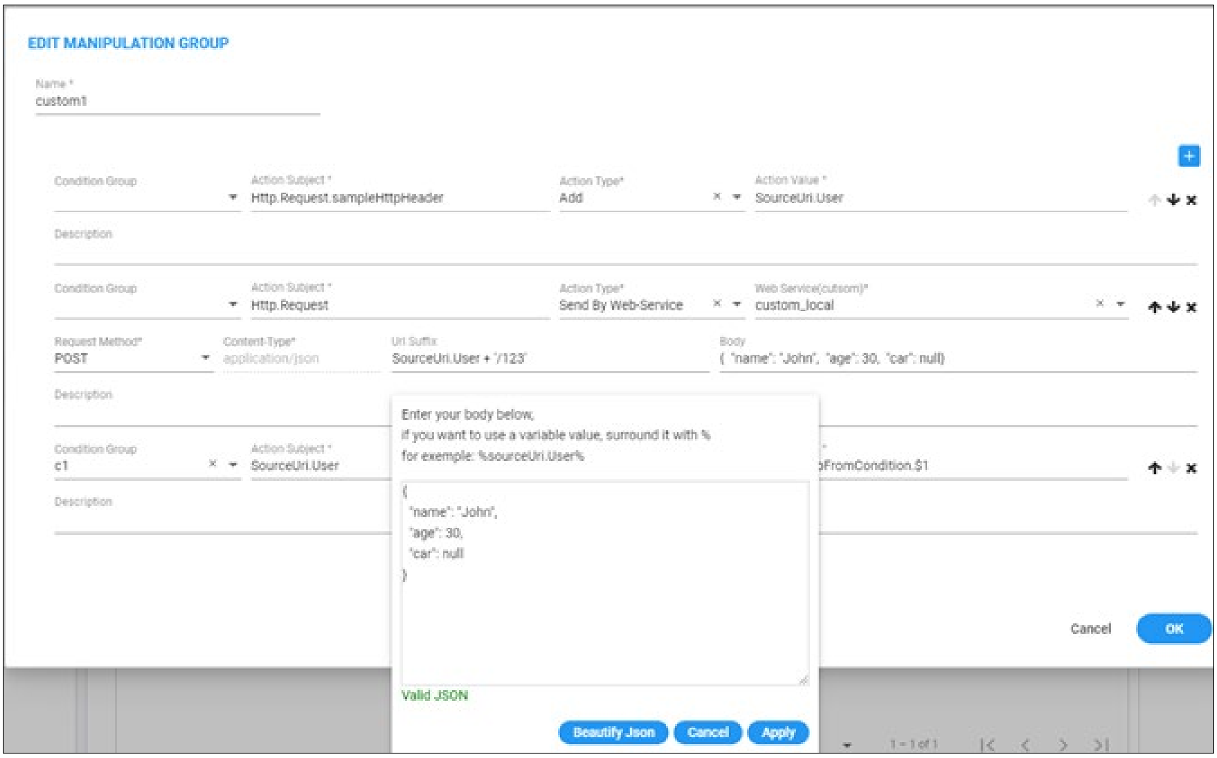
Condition Group: | Action Subject *: Http.Request.sampleHttpHeader | Action Type*: Add | Action Value *: SourceUri.User
Description:

Condition Group: | Action Subject *: Http.Request | Action Type*: Send By Web-Service | Web Service(cutsom)*: custom_local
Request Method*: POST | Content-Type*: application/json | Url Suffix: SourceUri.User + '/123' | Body: { "name": "John", "age": 30, "car": null}
Description:

Condition Group: | Action Subject *: SourceUri.User | Action Type*: Modify | Action Value *: Http.Response.Body.name
Description:

Condition Group: c1 | Action Subject *: SourceUri.Host | Action Type*: Modify | Action Value *: regexGroupFromCondition.$1

2.2.1 Preparing and Sending the Request



EDIT MANIPULATION GROUP

Name *: custom1

Condition Group: | Action Subject *: Http.Request.sampleHttpHeader | Action Type*: Add | Action Value *: SourceUri.User
Description:

Condition Group: | Action Subject *: Http.Request | Action Type*: Send By Web-Service | Web Service(cutsom)*: custom_local
Request Method*: POST | Content-Type*: application/json | Url Suffix: SourceUri.User + '/123' | Body: { "name": "John", "age": 30, "car": null}
Description:

Condition Group: c1 | Action Subject *: SourceUri.User | ...FromCondition.$1
Description:

Enter your body below,
if you want to use a variable value, surround it with %
for exemple: %sourceUri.User%

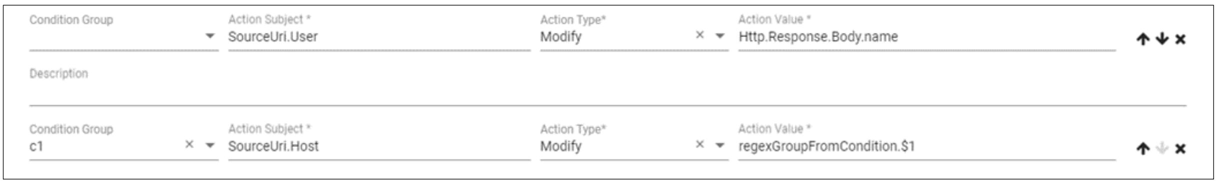
```
{
 "name": "John",
 "age": 30,
 "car": null
}
```

Valid JSON

- In the first rule, an HTTP header is added to the request
  - Its key will be ‘sampleHttpHeader’
  - It will receive its value from ‘SourceUri.User’
- In the second rule, a POST request will be sent by ‘custom_local’ web-service.
  - The URL suffix can be built in a similar way to ‘action value’ field.
  - The body of the request is built by writing full and valid JSON. Inside the JSON it is possible to combine values from the system/headers by wrapping them with the '%' character, as shown in the figure.

2.2.2 Using the Response

In the same Manipulation Group, operators can add rules after the request, and in these rules operators can use the response values, for example:

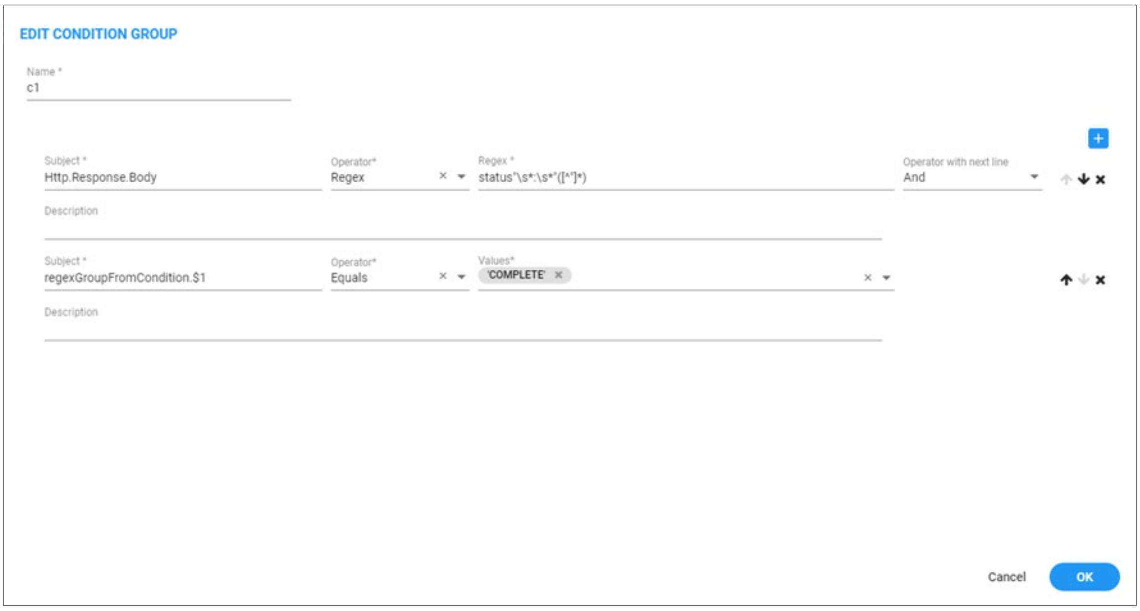


Condition Group
Action Subject *
SourceUri.User
Action Type*
Modify
Action Value *
Http.Response.Body.name
Description

Condition Group
c1
Action Subject *
SourceUri.Host
Action Type*
Modify
Action Value *
regexGroupFromCondition.$1

- [Refer to the preceding figure] In the first rule, in the ‘sourceUri.User’ field, the ‘name’ field will be applied if it exists in the body of the received response (currently, first-level JSON search is supported).
- In the second rule, note that first a condition-group (built separately) is called.
  Two new related options are available:
  - If the ‘Condition Group’ contains a Regular Expression (regex) condition that is matched, you can use its groups in the same Condition Group or in linked manipulation-group window (in the preceding figure, see ‘regexGroupFromCondition.$1’)
  - If a condition group is linked to a manipulation group like here, the operator is able to use the ‘Http.Response…’ values also in the linked condition-group.

  The following figure shows the linked condition-group.



EDIT CONDITION GROUP

Name *
c1

Subject *
Http.Response.Body
Operator*
Regex
Regex *
status"\s*:\s*"([^"]*)
Operator with next line
And
Description

Subject *
regexGroupFromCondition.$1
Operator*
Equals
Values*
'COMPLETE'
Description

In the first rule, a regex is performed on the body of the HTTP Response (possible if this Condition-Group is called from a Manipulation-Group).

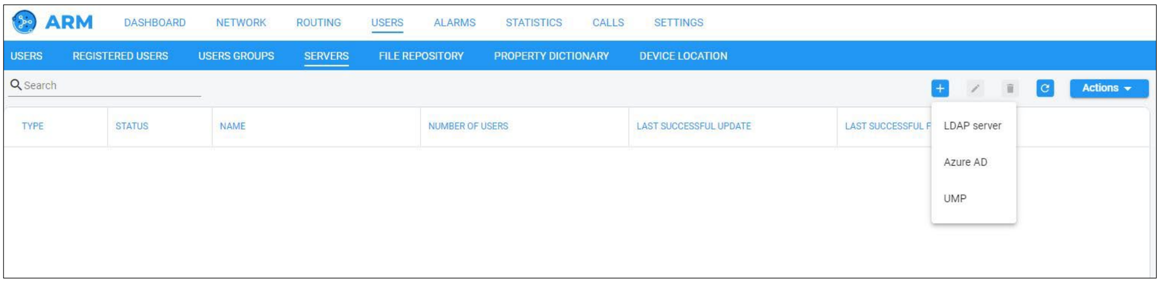
In the second rule, a test is made on the first group found in the last regex (if found).

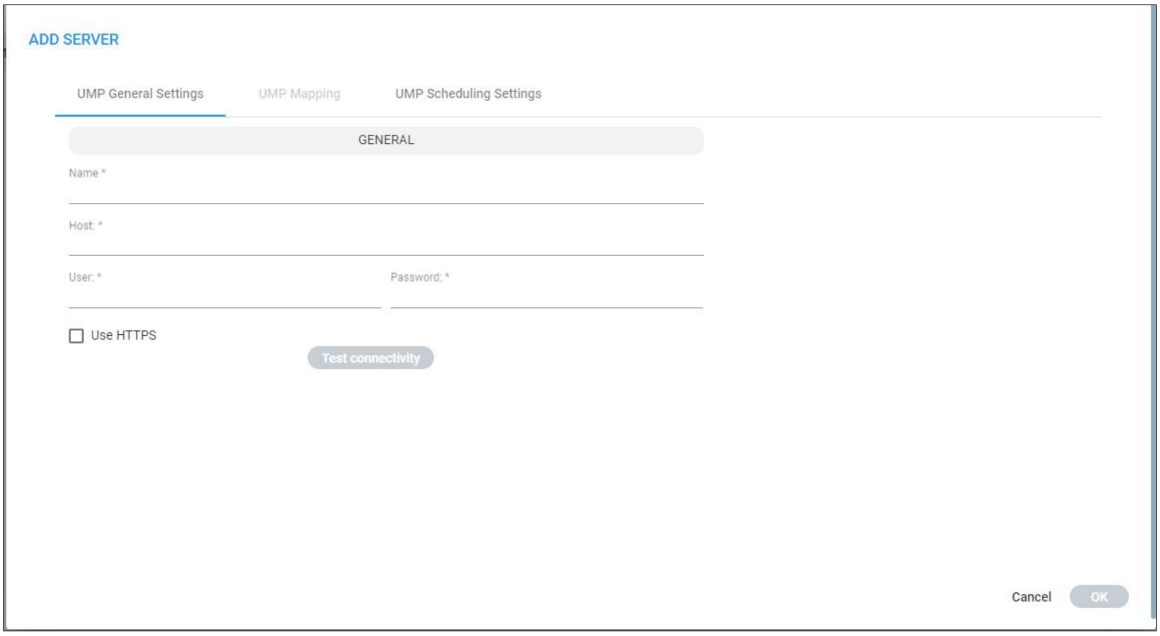
2.3 Adding a UMP Server

The Servers page (**Users** > **Servers**) in ARM 9.8.200 Fix 1 enables operators to add a User Management Pack (UMP) server for retrieving user data (click **+** and select **UMP**).

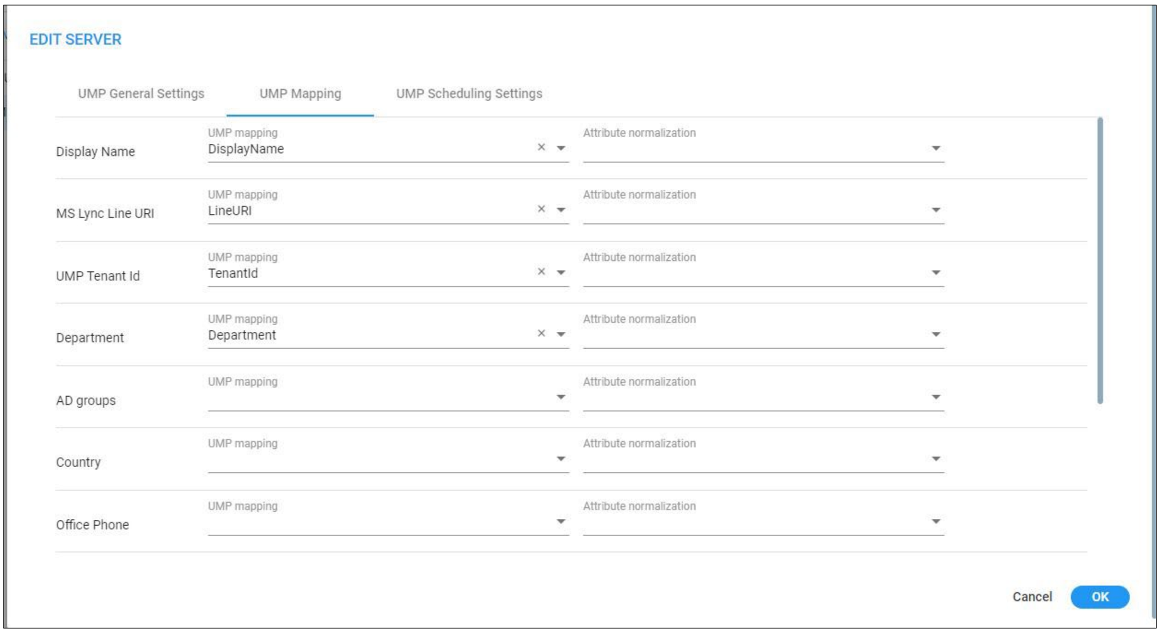
In earlier ARM versions, there was a limitation using the Azure AD server. Microsoft Graph API (RESTful web API that enables you to access Microsoft Cloud service resources) doesn't support retrieving Teams/Lync attributes such as 'EnterpriseVoiceEnabled' or 'OnPremLineURI' which are needed for routing.

By configuring a UMP server, the ARM can learn the users with their Teams/Lync attributes.





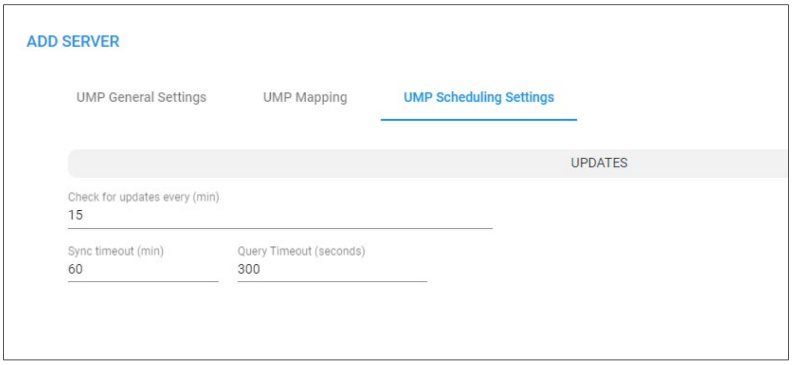
The operator enables the capability to map the local properties to the values from the UMP server.



EDIT SERVER

	UMP mapping	Attribute normalization
Display Name	DisplayName	
MS Lync Line URI	LineURI	
UMP Tenant Id	TenantId	
Department	Department	
AD groups		
Country		
Office Phone		

The operator enables the capability to configure the ‘Scheduling’ setting of the UMP server.



ADD SERVER

Check for updates every (min): 15

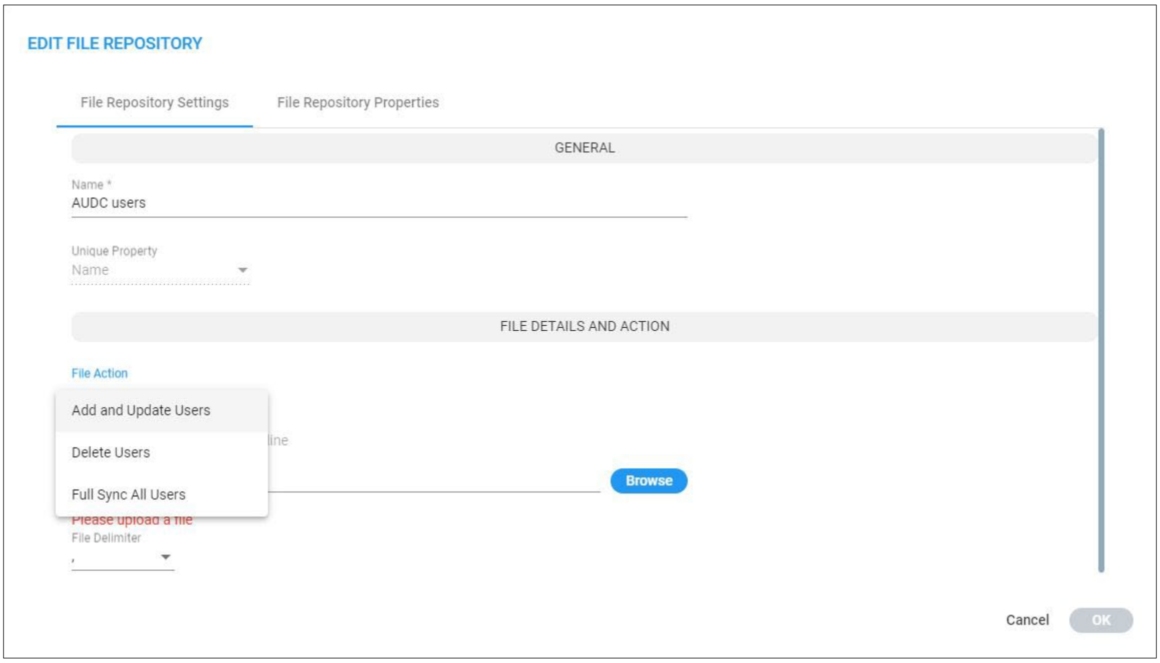
Sync timeout (min): 60

Query Timeout (seconds): 300

2.4 New Actions in File Repository

Three new management actions are supported by the ‘Edit File Repository’ screen (**Users** > **File Repository** > click ‘Edit’ icon) in ARM 9.8.200 Fix 1:

- Add and Update Users – enables operators to add new users and update existing users
- Delete Users - enables operators to delete existing users
- Full Sync All Users - enables operators to add new users, update existing users and delete existing users that aren’t in the File Repository



EDIT FILE REPOSITORY

GENERAL

Name *
AUDC users

Unique Property
Name

FILE DETAILS AND ACTION

File Action

Please upload a file

File Delimiter
,

2.5 Scheduling Synchronization for Each File Repository

From ARM version 9.8.200 Fix 1, operators can configure a synchronization schedule for each File Repository.

The synchronization source can be:

- Local file
- Azure Storage

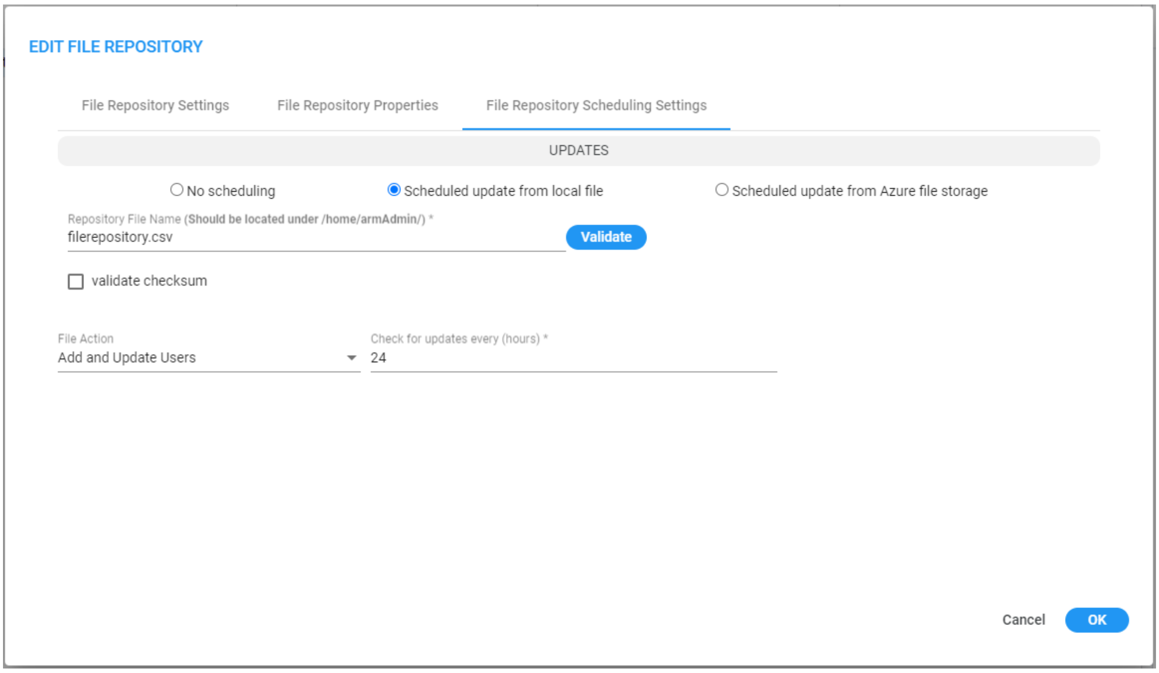
In the ‘Add file repository’ / ‘Edit file repository’ screen, there is a new tab:

File Repository scheduling settings

Under the new tab, three options are available:

- **No Scheduling** (default for new/existing File Repository)
- **Scheduled update from local file**
- **Scheduled update from Azure file storage**

Scheduled update from local file



EDIT FILE REPOSITORY

File Repository Settings | File Repository Properties | File Repository Scheduling Settings

UPDATES

○ No scheduling ● Scheduled update from local file ○ Scheduled update from Azure file storage

Repository File Name (Should be located under /home/armAdmin/) *
filerepository.csv

□ validate checksum

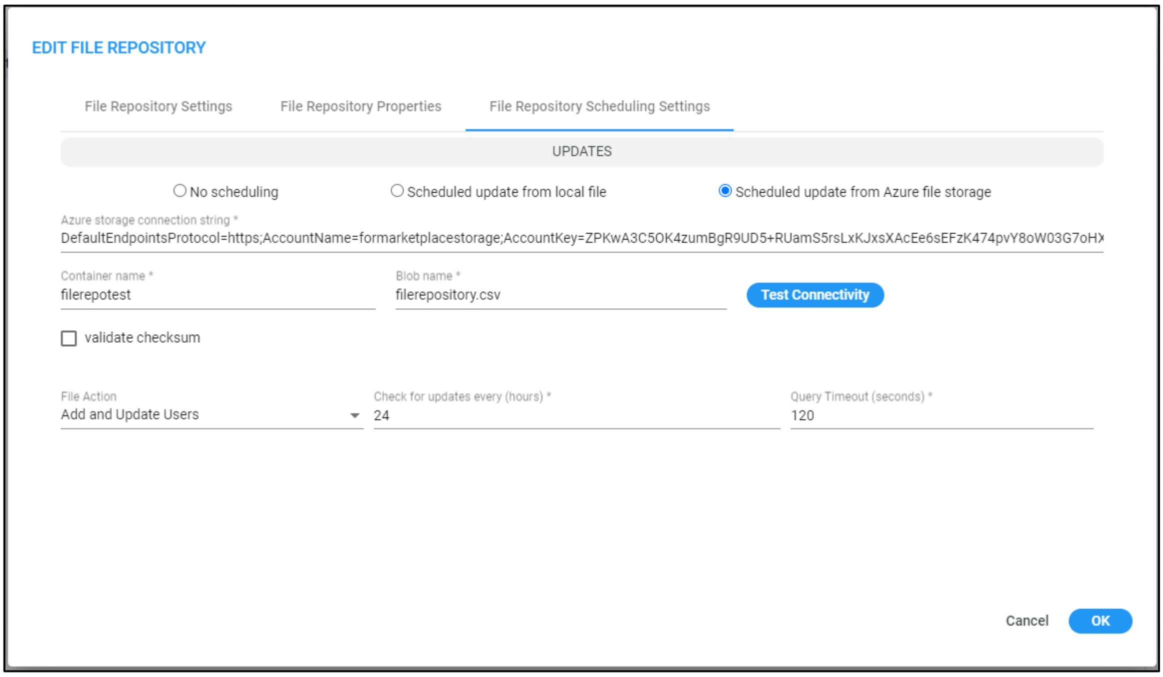
File Action: Add and Update Users

Check for updates every (hours) *: 24

The table below refers to the preceding figure.

Setting	Description
Repository File Name	• The name must include file format .csv. • The file must be located under the ‘/home/armAdmin/’ directory (Configurator machine).
Validate checksum	If this option is selected, a .txt file with checksum must be provided in addition to the .csv file. See explanation here.
File action	From the drop-down, the operator can select either: • **Add and Update Users** • **Delete Users** • **Full Sync All Users**
Update frequency (hours)	The operator can configure 1 hour *minimum*, in increments of 1 hour.

Scheduled update from Azure file storage



EDIT FILE REPOSITORY

File Repository Settings
File Repository Properties
File Repository Scheduling Settings
UPDATES

○ No scheduling ○ Scheduled update from local file ● Scheduled update from Azure file storage

Azure storage connection string *
DefaultEndpointsProtocol=https;AccountName=formarketplacestorage;AccountKey=ZPKwA3C5OK4zumBgR9UD5+RUamS5rsLxKJxsXAcEe6sEFzK474pvY8oW03G7oH

Container name *
filerepotest

Blob name *
filerepository.csv

□ validate checksum

File Action
Add and Update Users

Check for updates every (hours) *
24

Query Timeout (seconds) *
120

The table below refers to the preceding figure.

Setting	Description
Azure Storage connection string	See Azure documentation for more information.
Container name	See Azure documentation for more information.
Blob name	The name of the file; must include file format .csv.
Validate checksum	If this option is selected, a .txt file with checksum must be provided in addition to the .csv file. See explanation here.
File action	From the drop-down, the operator can select either: • **Add and Update Users** • **Delete Users** • **Full Sync All Users**
Check for updates every (hours)	The operator can configure 1 hour *minimum*, in increments of 1 hour.
Query Timeout (seconds)	Range: 1-6000 seconds

- If the option 'Validate checksum' is selected, the operator must supply a .txt file with the same name as that of the Repository File, which contains the 'sha256' checksum of the Repository File. For example, two files must be provided to ARM:
  - Users.csv (Repository File)
  - Users.txt (sha256 of the 'Users.csv' file)
- After the sync is completed, the Repository File and the Checksum file (if it exists) will be *removed* from Azure storage/local machine.
- The synchronized file must include the same headers as the initial File Repository.

Note: Limitations are the same as when manually uploading File Repository, namely:

- 1 GB max file size
- .csv file format
- Utf-8 encoding

2.6 DIDs Count

ARM 9.8.200 Fix 1 supports phone numbers (DIDs) counting. The feature allows operators to prevent DDOS/DOS and calls flooding attacks on the enterprise.

When configured, the operator can count either source or destination phone numbers (DIDs), calling or called, within a defined time period.

Operators can decide how to handle specific phone numbers by using ARM's generic routing capabilities.

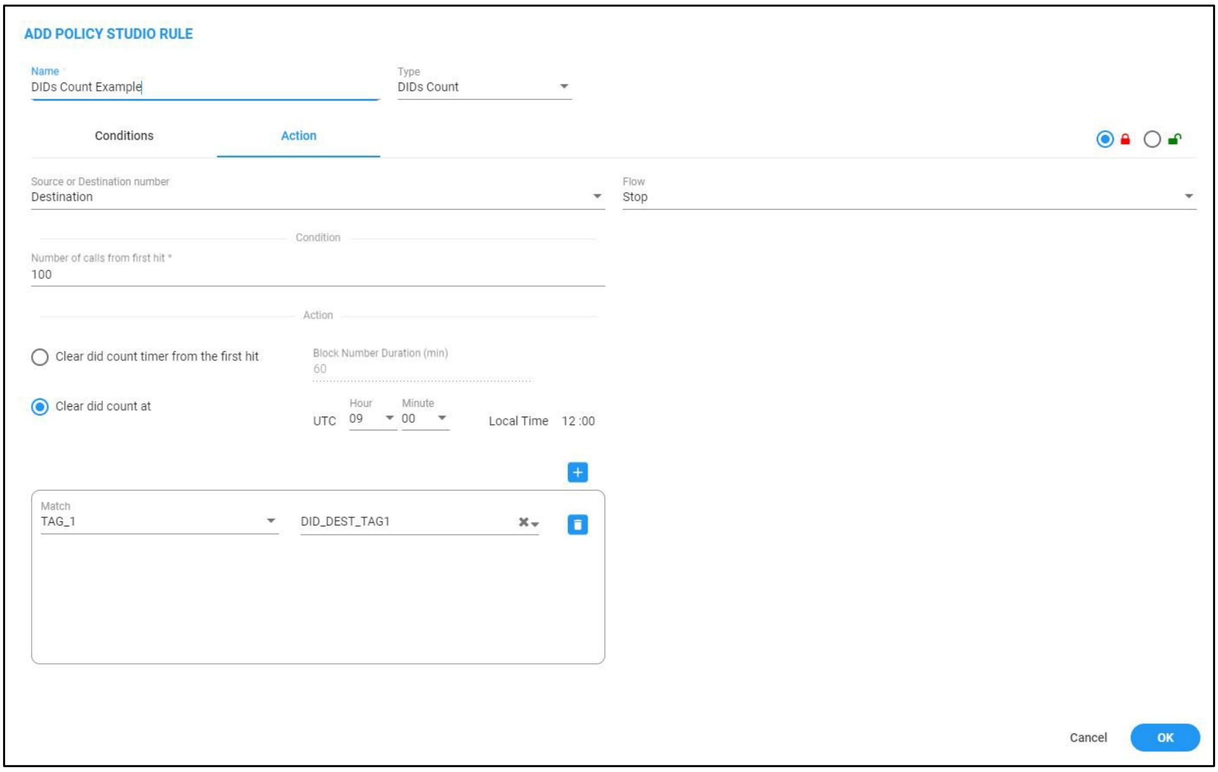
2.6.1 Configuring a DIDs Count

Network operators must configure a Policy Studio Rule with some criteria to count a phone number.

A new 'Type' has been added to Policy Studio: **DIDs Count**

DIDs Count conditions are identical to when **User** is defined for 'Type'.

When adding a new Policy Studio Rule, operators can choose the following criteria under the **Action** tab:



Name: DIDs Count Example

Type: DIDs Count

Source or Destination number: Destination

Flow: Stop

Number of calls from first hit *: 100

○ Clear did count timer from the first hit — Block Number Duration (min): 60

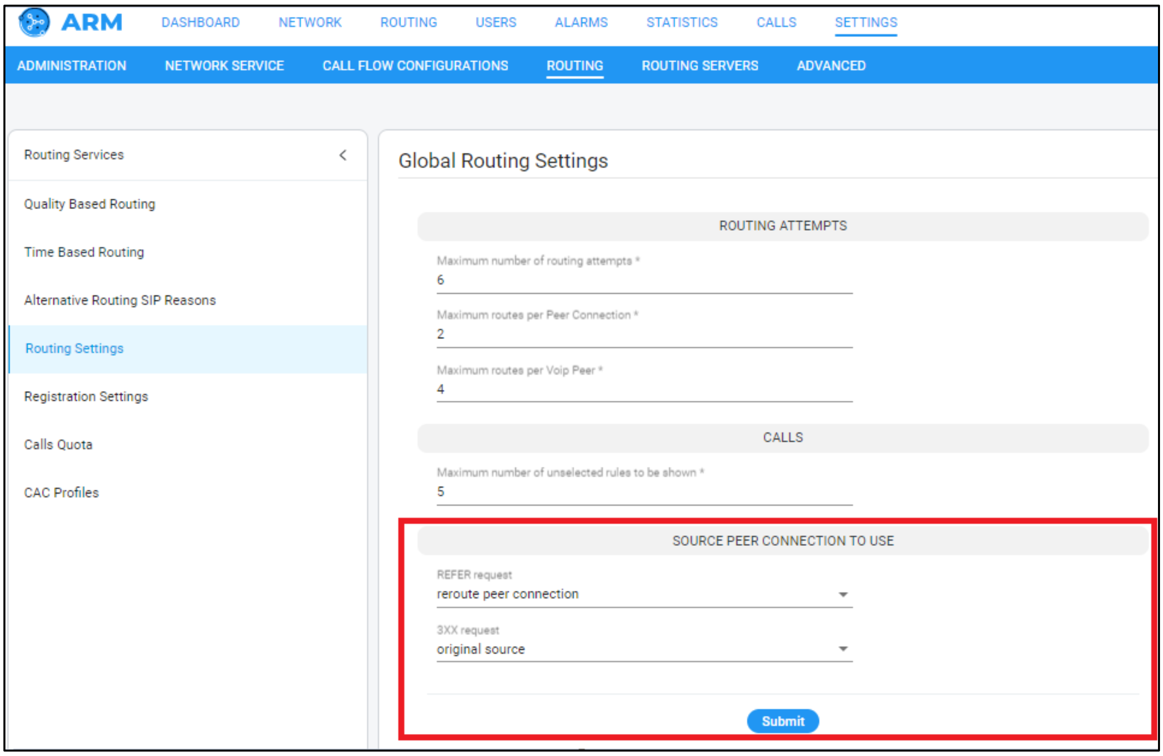
● Clear did count at — UTC Hour: 09 Minute: 00 Local Time 12 :00

Match: TAG_1 — DID_DEST_TAG1

- Source or Destination number – the phone number of the caller/callee to be counted.
- Number of calls from first hit: the number of calls after which to add the tags for this Policy Studio.
- Clear DID count timer from the first hit - the duration, in minutes, from the first hit until the count clearing.
- Clear DID count at - the specific time of day for the count clearing.
- Adding tags = Tag_1/2/3. Tag info is used in Routing Rules.

2.7 Reroute Peer Connection for REFER and 3XX Requests

For REFER and 3XX requests, operators can configure from ARM version 9.8.200 Fix 1 whether the source Peer Connection (which will then be used to find appropriate Routing Rules) will be the original Peer Connection or whether the Reroute Peer Connection will be used instead.



Global Routing Settings

ROUTING ATTEMPTS

Maximum number of routing attempts *
6

Maximum routes per Peer Connection *
2

Maximum routes per Voip Peer *
4

CALLS

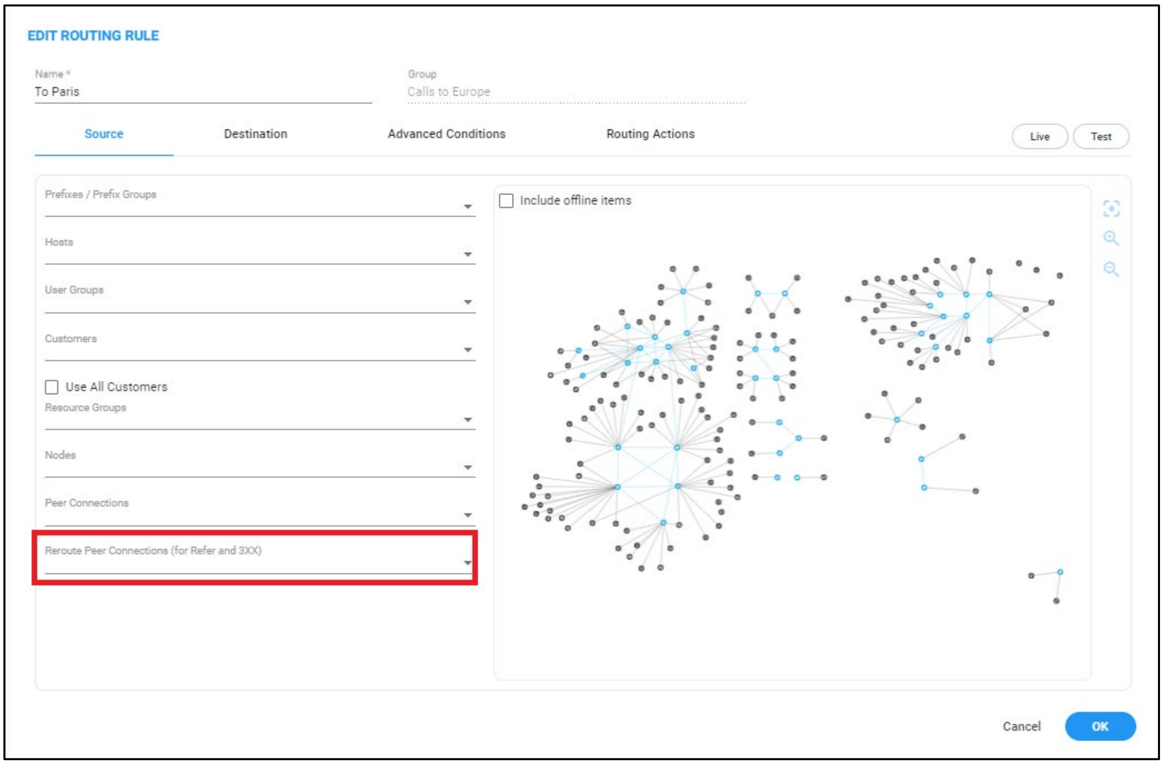
Maximum number of unselected rules to be shown *
5

SOURCE PEER CONNECTION TO USE

REFER request
reroute peer connection

3XX request
original source

A new field ‘Reroute Peer Connection’ has been added to the Add / Edit Routing Rule screen under the ‘Source’ tab.



EDIT ROUTING RULE

Name * To Paris

Group Calls to Europe

Prefixes / Prefix Groups

Hosts

User Groups

Customers

□ Use All Customers

Resource Groups

Nodes

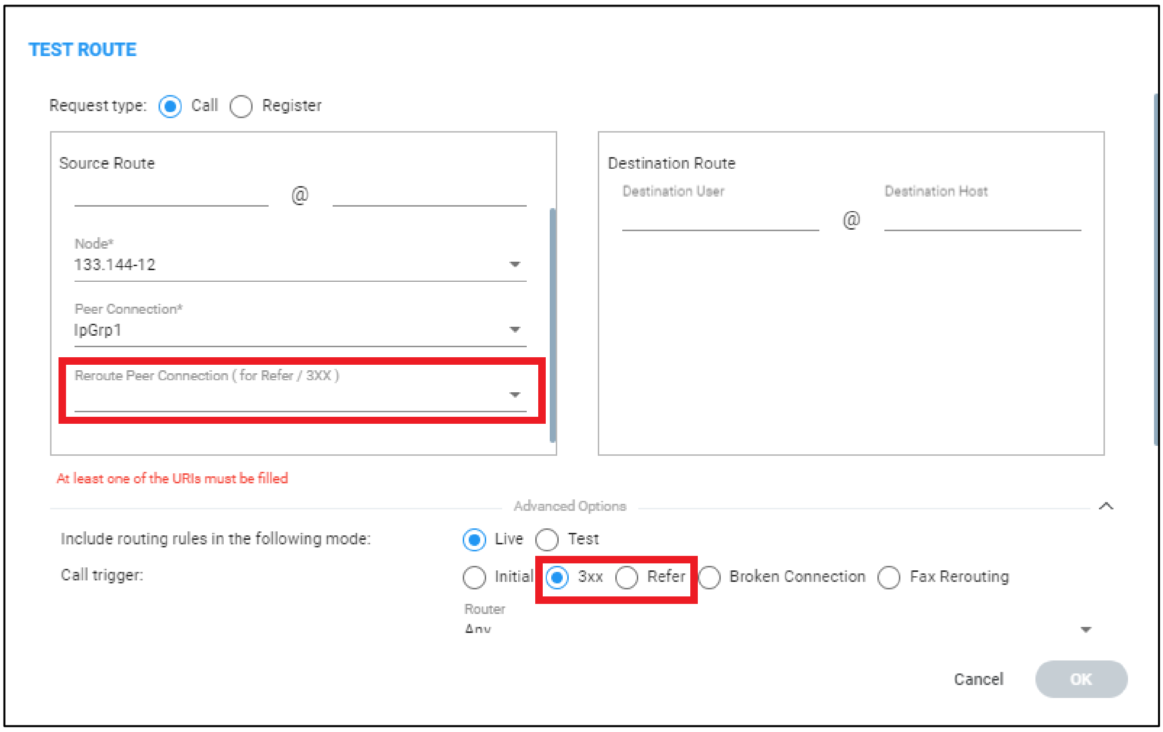
Peer Connections

Reroute Peer Connections (for Refer and 3XX)

□ Include offline items

The field is only relevant if the call was triggered from REFER/3XX, which allows a Routing Rule to be triggered using a ‘Reroute Peer Connection’.

Modifications have also been made to the Test Route screen. As a result of the previously described changes, it’s now also possible in the Test Route screen to select Reroute Peer Connection for REFER/3XX.



TEST ROUTE

Request type: ● Call ○ Register

Source Route

@

Node* 133.144-12

Peer Connection* IpGrp1

Reroute Peer Connection (for Refer / 3XX)

Destination Route

Destination User @ Destination Host

At least one of the URIs must be filled

Advanced Options

Include routing rules in the following mode: ● Live ○ Test

Call trigger: ○ Initial ● 3xx ○ Refer ○ Broken Connection ○ Fax Rerouting

Router Any

This page is intentionally left blank.

3 Supported Platforms

ARM 9.8.200 Fix 1 supports the platforms shown in the table below.

Table 3-1: ARM 9.8.200 Fix 1 Supported Platforms

ARM	Platform	Application
GUI	Web Browser	Firefox, Chrome, Edge
Deployment	VMWare	VMware ESXI 6.7, 7.0 Update 2
	HyperV	Windows Server 2016 Hyper-V Manager Microsoft Corporation Version: 10.0.14393.0
	KVM	KVM environment on CentOS 7
	OpenStack	Xena release on CentOS Stream 8

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	-

#	Feature	Earliest Node Software Supporting It	Comments
10	Support for combined ARM and SIP based Routing decision (Route based on Request URI)	Version 7.20A.256.391	Supported for SBC only
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	-
16	Delayed Alternative Routing	Official build from SBC 7.4.200 stream	-
17	Story of a call: Integration with Voca. Additional information in ARM calls information.	Official build from SBC 7.4.200 stream	-
18	Support for more efficient way of synchronization of SBC IP groups with ARM	Official build from SBC 7.4.200 stream	If the customer runs earlier SBC SW, the synchronization will work in a pre-ARM 9.6 way.
19	Support for multiple connections from SBC to ARM Router	Official build from SBC 7.4.300 stream	-
20	Support for SIP Conditions and Manipulations for SIP Header fields (SBC-level bug fixes)	Latest official build from SBC 7.2.250, SBC 7.4.250, SBC 7.4.400	-

#	Feature	Earliest Node Software Supporting It	Comments
21	Preserving of the same Call ID when traversing more than one SBC.	SBC 7.40A.490.134 or later	

This page is intentionally left blank.

5 Resolved Issues in ARM 9.8.200 Fix 1

The table below lists major issues which were encountered by customers in previous releases but which are resolved in ARM 9.8.200 Fix 1.

Table 5-1: Resolved Issues in ARM 9.8.200 Fix 1

Incident	Problem / Limitation
ARM-6330	Removing and adding P-Asserted Identity doesn't work in manipulation.
ARM-6293	ARM failing inbound calls. The issue was encountered when the following two circumstances occurred in combination: - Source is third-party node - Unselected rule (in this case because registered user isn't found)
ARM-6229	[ARM 9.6.21] ARM cannot sync with SBCs.
ARM-6228	There is no possibility to save the value of AC-Session-ID and send by another header. AC-Session-ID is taken from the new header.
ARM-6140	ARM Radius login issue.
ARM-6135	[ARM 9.8] The Azure AD option 'Fetch all groups and all their members' does not work.
ARM-6123 ARM-6341	ARM Dashboard was displaying some active alarms when no alarms were active because the active alarms were acknowledged. Starting from ARM 9.8.200 Fix 1, ARM must display only **un**acknowledged alarms on the Dashboard. A note must be added indicating that 'Only unacknowledged alarms are displayed'.
ARM-6121	When call duration was more than 40 minutes, no outgoing end call was displayed in Call Summary.
ARM-5938	ARM unable to get all groups from Azure AD.
ARM-5907	ARM can't retrieve Azure users.
ARM-6193	ARM upgrade from 9.6.19 to 9.8.113 failed because of the AVX configuration in VMware.
ARM-5439	If an alternative route exists in a call from/to a customer, ARM Router sends call-info via the third alternative route even though it's not relevant to 'customers' (does not have 'customer' as a Routing Rule condition but only matches other call criteria).
ARM-5440	When collecting logs from ARM Router, the dump command results in an error message with no database output.
ARM-5447	RADIUS authentication doesn't allow a password of more than 16 characters; RADIUS 2.0, which allows more characters, is not yet supported.
ARM-5460	An operator with a security level of 'Admin' cannot change their own password. An operator of security level 'Security Admin can change their own password but configuring the operator with this security level is not an option when the customer wants to prevent the operator from changing/deleting/creating other accounts.
ARM-6475	Update of Media Realm and other attributes related to ARM Connection are not reflected at the SBC level.
ARM-6474	In an Azure deployment, the Python version is 3.6 (and not 3.8 as in VMWare). This causes issues with related scripts (like backup/restore).

Incident	Problem / Limitation
ARM-6466	In the Call Details page, the information displayed for the Source and Dest URI of a call does not reflect manipulation (changed information should be displayed after manipulation).
ARM-6459	In the Calls page, an issue is encountered with calls navigation (paging).
ARM-6458	Sometimes, during Configurator's bring-up, the CDR manager starts before Mongo resulting in no calls (CDRs) information in the ARM.
ARM-6457	In the Calls page, ARM does not display Call Details (no pop-up with details).
ARM-6455	CVE-2023-46604 (upgrade activemq version)

6 Tested ARM Capacities

The table below lists the tested capacities of ARM Low Profile and ARM High Profile. 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	ARM Low Profile	ARM High Profile
CAPs (assuming the average call duration is 100 seconds)	50 CAPs per ARM Router	300 CAPs per ARM Router
	ARM total: 100 CAPs	ARM total: 3000 CAPs
ARM Routers	4	150
Routing Groups	100	2,000
Routing Rules per ARM	1,000	10,000
ARM Users (either local or LDAP/Azure AD)	100,000	1 million Possible extension to 4 million when ordering a special Feature Key. Requires 16 GB memory for Routers.
'Customer' entities (Teams tenants)	2,000	Up to 20,000
Nodes number	10	150 by default Possible extension to 300 requires ARM Configurator with 8 CPUs and 32 GB memory.
Peer Connections	Per Node: 60	Per Node: 600
	ARM total: 100	ARM total: 30,000
Connections	100	10,000
Prefix Groups	200	2,000
Prefixes in a single Prefix Group	200	2,000
Normalization rules	50	2,000
Calls history	1 million	10 million
Threshold alarms	10 threshold rules 5 elements/entities per rule	150 threshold rules 25 elements/entities per rule
Statistics history	3 months	1 year

This page is intentionally left blank.

7 Known Limitations and Workarounds

The table below lists the known limitations and workarounds in ARM 9.8.200 Fix 1.

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 *must* be **ARMTopology** 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.	-
ARM-6403	SBC Media Security Mode "offer-both-answer-prefer-protected" is not supported by ARM.	If the operator has to configure SBC level IP profile with this mode, this IP profile should not be used by ARM.
Installation Manual	The chapter 'Deploying the ARM from Microsoft's Azure Marketplace' in the 'Installation Manual' includes screenshots of an old Azure version which are currently inaccurate.	These screenshots will be changed in the document in the next major release.

Incident	Problem / Limitation	Comments/Workaround
Security		
-	The ARM does not prevent the opening of HTTP for debugging purposes. HTTPS should be used for debugging purposes.	Operators should consider security threats when enabling HTTP communication between ARM network components and SBCs.
ARM-5846 CVE: 2021-41617	ARM 9.8.200 Fix 1 uses openSSH version 8.0. This openSSH version is potentially exposed to CVE: 2021-41617. ARM OS default settings make sure that the ARM machine is not exposed to this CVE finding even though version 8.0 is used.	ARM operators are required not to change the default configuration of AuthorizedKeysCommand and AuthorizedPrincipalsCommand attributes (default = disabled).
CESA-2021:4150	When ARM 9.8.200 Fix 1 is deployed in Azure and if the customer chooses to activate the WALinuxAgent agent, it will imply usage of Python 36 which is exposed to security vulnerability.	For Azure ARM users, it's up to the customer to either deactivate WALinuxAgent or to use an older version of Python (Python 36).
Breaking changes		
-	ARM 9.8.200 Fix 1 does not support 'Build Star' and 'Build Mash' capabilities. These capabilities were removed from the GUI and REST API starting from ARM 9.4 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 *REST API Developer's Guide for ARM 9.2 and later*. If customers develop scripts based on this REST API, these scripts should be adjusted to the new REST API when moving to ARM 9.2 and later.
-	Starting from ARM 9.4, the REST API for getting all VoIP Peers (VoIP Peers GET API) is changed. 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 *REST API Developer's Guide for ARM 9.4*. 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 and later.
-	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, ARM 9.2, ARM 9.4, ARM 9.6, ARM 9.8, ARM 9.8.100 and ARM 9.8.200 Fix 1.	Customers upgrading from a previous version will still view the action but are advised to exclude it from routing definitions.
-	In ARM 9.8.200 Fix 1 (starting from ARM 9.4), when an alarm for a Routing Rule is generated, the detailed alarm information is placed in both **Additional Info 1** and **Additional Info 2**.	Operators should use information from both fields. This is done to provide detailed information about the alarm without truncation.

Incident	Problem / Limitation	Comments/Workaround
-	ARM 9.8.200 Fix 1 (starting from ARM 9.6) REST API is not backward compatible in the definition (Add / Edit / Delete) of 'Alternative Routing SIP Reason'. This is due to the new feature (Sets of SIP Reasons for Alternative Routing).	Customers should take this into consideration. The new REST API for managing SIP reasons is described in the *REST API Developer's Guide* for *ARM 9.6*.
Upgrade		
-	Direct upgrade from ARM 9.0 and earlier to ARM 9.8.200 Fix 1 is not supported.	For these cases, a two-step upgrade is required: Step 1: Upgrade to ARM 9.2 (not regular upgrade, including OS upgrade) Step 2: Upgrade to ARM 9.8.200 Fix 1 **Note**: The following direct upgrades are supported: • ARM 9.2 > ARM 9.8.200 Fix 1 • ARM 9.4 > ARM 9.8.200 Fix 1 • ARM 9.6 > ARM 9.8.200 Fix 1 • ARM 9.8 > ARM 9.8.200 Fix 1
-	For pre-ARM 9.2 deployments, the upgrade to ARM 9.8.200 Fix 1 is not a regular upgrade. It requires two steps: first to ARM 9.2 and then to ARM 9.8.200 Fix 1. The upgrade to ARM 9.2 upgrades the OS of all components to CentOS Stream from CentOS6. Note that for ARM 9.2 and ARM 9.4 deployments (running CentOS8), the upgrade is smooth. ARM 9.6, ARM 9.8, ARM 9.8.100 and ARM 9.8.200 Fix 1 also run on CentOS Stream.	Make the following preparations: • 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). • Request from AudioCodes a Feature Key with all the ordered features and ordered number of sessions for the new VM in ARM 9.8.200 Fix 1. • Prepare temporary IP and VM resources required for each server upgrade. • Prepare extended storage for the ARM Configurator (the ARM Configurator allocates 80 GB in ARM 9.8.200 Fix 1 – like in ARM 9.4, ARM 9.6 and ARM 9.8).
-	To upgrade to ARM 9.8.200 Fix 1 in a VMware environment, the customer must have VMware ESXI 6.7 or 7.0 update 2 (earlier versions are not supported with CentOS Stream).	-
-	Miscellaneous issues with the ARM GUI after upgrading from previous releases.	Customers are requested to clear the browser cache after performing a software upgrade (**Ctrl**+**F5**).
GUI Incidents		
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-6392	Advanced search for Peer Connections with a specific 'Operative state' filter does not function correctly.	-

Incident	Problem / Limitation	Comments/Workaround
ARM in Azure		
ARM-4676	[ARM in Azure with SBCs behind Load Balancer] 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	[ARM in Azure with SBCs behind Load Balancer] 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 **Keep connection properties synchronized** 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.

ARM-6108	In an Azure deployment, if the customer upgrades to ARM 9.8.200 Fix 1 from ARM 9.6 or earlier, the WALinuxAgent agent is not present. Or if upgrading from ARM 9.8 to ARM 9.8.200 Fix 1, the agent will not be activated.	Check whether the Azure Linux Agent is installed:
		`dnf list installed WALinuxAgent`
		If the Azure Linux Agent is installed, enable and restart it using the following commands:
		`systemctl enable waagent`
		`service waagent restart`
		To install the Azure Linux Agent, run the following commands:
		`dnf install -y WALinuxAgent`
		`systemctl enable waagent`
		`service waagent restart`
		Check the status of the Azure Linux Agent service:
		`systemctl status waagent`

This page is intentionally left blank.

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

AudioCodes Inc.
80 Kingsbridge Rd.,
Piscataway, NJ 08854, USA
Tel: +1-732-469-0880
Fax: +1-732-469-2298

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