

Initial Configuration Guide

Virtual Multimedia Resource Function

User Guide

Copyright

© Ericsson AB 2016, 2017. All rights reserved. No part of this document may be reproduced in any form without the written permission of the copyright owner.

Disclaimer

The contents of this document are subject to revision without notice due to continued progress in methodology, design and manufacturing. Ericsson shall have no liability for any error or damage of any kind resulting from the use of this document.



Contents

1	About This Document	1
2	vMRF Initial Configuration Tasks	2
3	Configure vMRF Application	3
3.1	Manual Configuration	4
3.2	Importing Configuration Data from NETCONF Template	12
3.3	Initial Configuration Data for the vMRF Application	14
4	Configure SNMP for Fault Management	17
5	Configure the networkManagedElementId Attribute	18





1 About This Document

This document describes the configuration that is a prerequisite for vMRF to process traffic. The configuration tasks in this document are performed after the vMRF VNF has been deployed using the relevant *deployment instructions*.

This document is written for vMRF operator personnel who are responsible for the deployment of vMRF. The vMRF operator is assumed to be a cloud service consumer on a cloud service.



2 vMRF Initial Configuration Tasks

Initial configuration includes the following tasks:

Steps

- vMRF application and security configuration. This is a mandatory configuration task. See [Configure vMRF Application](#) on page 3.
- SNMP configuration for fault management. This is a recommended configuration task. See [Configure SNMP for Fault Management](#) on page 17.
- Configuration of the `networkManagedElementId` attribute in the *ManagedElement* MO. This is a recommended configuration task. See [Configure the networkManagedElementId Attribute](#) on page 18.
- Performance management jobs configuration. This is a recommended configuration task, for more information refer to *Performance Management*.

3 Configure vMRF Application

Initial vMRF application configuration consists of configuring user plane networks and H.248 signaling links.

The following methods are available, as shown in [Figure 1](#):

- Manual configuration after instantiation, as described in [Manual Configuration](#) on page 4.
- Importing configuration data using the deployment template during instantiation, as described in the relevant *deployment instructions* (the deployment template references the NETCONF template that contains the actual configuration data), and configuring security, as described in [Additional Security Configuration](#) on page 14.
- Importing configuration data using the deployment template and configuring security after instantiation, as described in [Importing Configuration Data from NETCONF Template](#) on page 12.

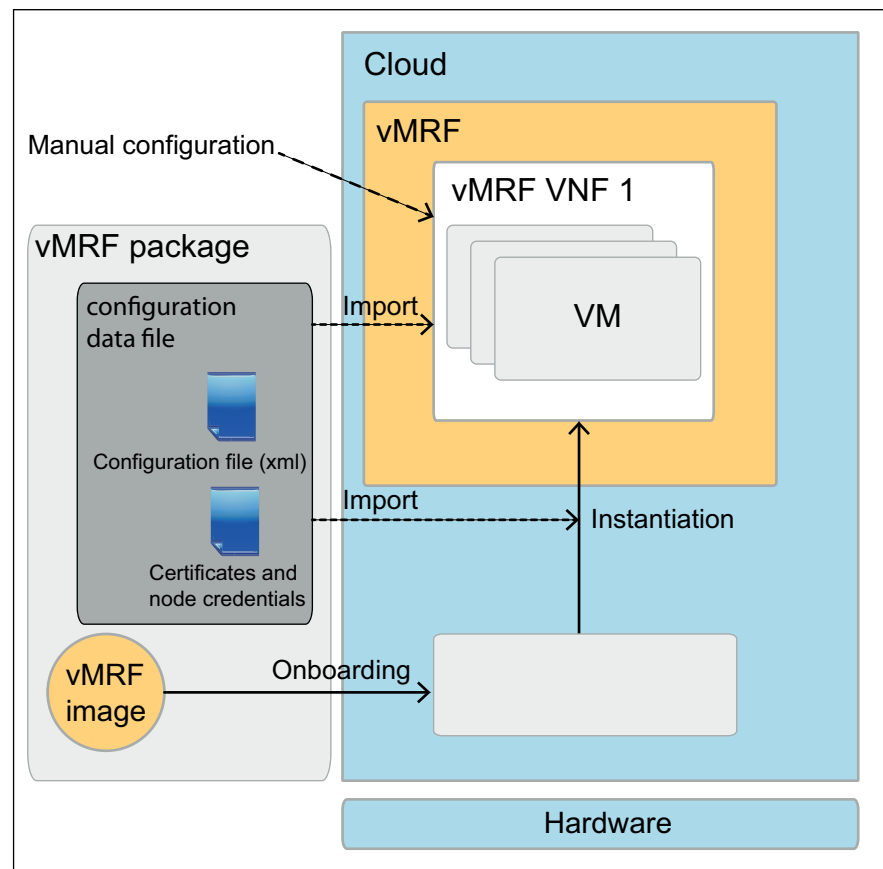


Figure 1 Options for Initial Configuration of vMRF Application



3.1 Manual Configuration

Manual configuration means configuring each MO and MO attribute using the ECLI or a NETCONF client. This procedure is the recommended method for initial VNF configuration.

A basic configuration with example values is provided in the software delivery package. This configuration can be used as a base for VNF configuration. Its lines of valid configuration data can be pasted using the ECLI.

Manual configuration requires emergency user credentials for logging in to the vMRF VNF until security configuration is completed.

If ECLI is used, do the following:

Steps

1. Connect to the O&M IP address of the vMRF VNF by issuing the following command:

```
ssh <user ID>@<O&M IP address>
```

2. Start a session by issuing the `cliss` command.

If a NETCONF client is used, connect to the O&M IP address of the vMRF VNF using the client.

3.1.1 Configure Security

For information on security configuration, refer to *vMRF Security Management*.

3.1.2 Configure vMRF User Plane

Steps

1. In the MOM, navigate to `ManagedElement=1,MediaResourceFunction=1`. Check that the `mediaIpVersion` attribute is configured correctly.

Note: The `mediaIpVersion` attribute is only visible in configure mode.

See [Table 2](#) for the details.

3.1.3 Emergency and Priority Call Handling

A certain amount of processing capacity (priority pool) can be reserved by the vMRF for emergency and priority calls. The `capacityForPriorityCalls` attribute of the *MediaResourceFunction* MO defines a fraction of the total



processing capacity and internal resources that is reserved for emergency and priority calls. The attribute is a numeric value, where 1 is 1/1000 fraction of the total processing capacity. The default value is 20.

3.1.3.1 Configuration of Priority Pool

This procedure describes how to set the size of the priority pool.

Prerequisites:

- An Ericsson Command-Line Interface (ECLI) session in Exec mode is open.

Steps

1. Navigate to the *MediaResourceFunction* MO and enter configure mode:

```
>ManagedElement=1,MediaResourceFunction=1
(MediaResourceFunction=1)>configure
```

2. Modify the value of the `capacityForPriorityCalls` attribute:

```
(config-
MediaResourceFunction=1)>capacityForPriorityCalls=<value>
```

3. Commit the changes:

```
(config-MediaResourceFunction=1)>commit
```

4. The job is completed.

3.1.4 EVS Configuration

EVS configuration consists of the following activities:

- Configuring EVS bandwidth and bit rate

3.1.4.1 Configure EVS Bandwidth and Bit Rate

This procedure describes how to configure allowed bandwidth range and maximum allowed bit rate for EVS to control EVS codec configuration and negotiation.

EVS codec negotiation is affected by bandwidth and bit rate configurations according to the following scenarios:

- For incoming SDP offer, the common subset of the configured bandwidth and bit rate values, and the bandwidth and bit rate values received in the offer, are sent back in the SDP answer. The bandwidth and bit rate sent in the SDP answer are used for the media stream.



- For outgoing SDP offer, the configured bandwidth and bit rate values are sent in the SDP offer. The bandwidth and bit rate values received in the SDP answer are used for the media stream.

The possible bandwidth and bit rate combinations are shown in [Table 1](#).

Table 1 EVS Bandwidth and Bit Rate Combinations

Bandwidth	Bit Rate
Narrowband (300 – 3400 Hz)	5.9 kbps, 7.2 kbps, 8 kbps, 9.6 kbps, 13.2 kbps, 16.4 kbps, 24.4 kbps
Wideband (50 – 7000 Hz)	5.9 kbps, 7.2 kbps, 8 kbps, 9.6 kbps, 13.2 kbps, 16.4 kbps, 24.4 kbps, 32 kbps, 48 kbps, 64 kbps, 96 kbps, 128 kbps
Super-wideband (50 – 14000 Hz)	9.6 kbps, 13.2 kbps, 16.4 kbps, 24.4 kbps, 32 kbps, 48 kbps, 64 kbps, 96 kbps, 128 kbps
Fullband (20 – 20000 Hz)	16.4 kbps, 24.4 kbps, 32 kbps, 48 kbps, 64 kbps, 96 kbps, 128 kbps

Prerequisites

An ECLI session in Exec mode is open.

Steps

1. Navigate to the *MrfConfiguration* MO and enter configure mode:

```
>ManagedElement=1,MediaResourceFunction=1,MrfConfiguration=1
```

```
(MrfConfiguration=1)>configure
```

2. Create an *MrfData* MO:

```
(config-MrfConfiguration=1)>MrfData=1
```

3. Navigate to the *EvsService* MO and enter configure mode:

```
>ManagedElement=1,MediaResourceFunction=1,MsProcessing=1,EvsService=1
```

```
(EvsService=1)>configure
```

4. Create an *EvsConfData* MO.

```
(config-EvsService=1)>EvsConfData=1
```

5. Set the supported bandwidth range:



```
(config-
EvsConfData=1)>supportedBwRange=<supported_bandwidth_range>
```

6. Set the minimum value for the supported bit rate range:

```
(config-
EvsConfData=1)>supportedBitRatesRangeBegin=<value>
```

7. Set the maximum value for the supported bit rate range:

```
(config-
EvsConfData=1)>supportedBitRatesRangeEnd=<value>
```

8. Commit the changes:

```
(config-EvsConfData=1)>commit
```

9. Navigate to the *MrfData* MO in which the configuration changes are needed and enter configure mode:

```
>ManagedElement=1,MediaResourceFunction=1,MrfConfiguration=1,MrfData=<1>
```

```
(MrfData=1)>configure
```

10. Modify the value of the attribute *evsConfDataMoRef* so that it refers to the *EvsConfData* created in [Step 4](#).

```
(config-
MrfData=1)>evsConfDataMoRef="ManagedElement=1,MediaResourceFunction=1,MsProcessing=1,EvsService=1,EvsConfData=1"
```

11. Commit the changes:

```
(config-MrfData=1)>commit
```

12. Navigate to the *MrfH248Interface* MO in which the configuration changes are needed, and enter configure mode:

```
>ManagedElement=1,MediaResourceFunction=1,MrfH248Control=1,MrfH248Interface=1
```

```
(MrfH248Interface=1)>configure
```

13. Modify the *mrfDataMoRef* attribute of the *MrfH248Interface* MO so that it refers to the *MrfData* MO that contains a reference to the *EvsConfData* created in [Step 4](#).

```
(config-
MrfH248Interface=1)>mrfDataMoRef="ManagedElement=1,MediaResourceFunction=1,MrfConfiguration=1,MrfData=1"
```

14. Commit the changes:



```
(config-MrfH248Interface=1)>commit
```

3.1.5 Configure vMRF Signaling

Steps

1. In the MOM, navigate to `ManagedElement=1,MediaResourceFunction=1,MrfH248Control=1`, and check that the default setting for the local SCTP port is correct. Modify the settings, if necessary.

See [Table 3](#) for the details.

2. Navigate to `ManagedElement=1,Transport=1,SctpProfile=1`, and check that the default settings for SCTP are correct. Modify the settings, if necessary.

See [Table 8](#) for the details.

3. In the MOM, navigate to `ManagedElement=1,MediaResourceFunction=1,MrfH248Control=1`, and create a *MrfH248Interface* MO for the H.248 connection.

In the created *MrfH248Interface* MO, configure the following attributes:

- `mrfH248InterfaceId`
- `remoteIpAddress`
- `remotePortNumber`

See [Table 4](#) for the details.

4. To activate the H.248 link towards the MTAS, set the administrative state of the given *MrfH248Interface* MO to `UNLOCKED`.

See [Table 4](#) for the details.

3.1.6 Announcement Configuration

3.1.6.1 Modify the Default Language Code

This procedure describes how to modify the default language code of announcements. The default language code is used as language code for the basic and variable announcements in case the H.248 (or SIP) message does not contain any language code for the requested announcement.

Prerequisites

An Ericsson Command-Line Interface (ECLI) session in Exec mode is open



Steps

1. In the MOM, navigate to the *Announcements* MO and enter configure mode:

```
>ManagedElement=1,MediaResourceFunction=1,Announcements=1
```

```
(Announcements=1)>configure
```

2. Modify the value of the `defaultLanguageCode` attribute:

```
(config-Announcements=1)>defaultLanguageCode=<value>
```

Note: The language code has the format of the RFC 5646 language tag.

3. Commit the changes:

```
(config-Announcements=1)>commit
```

3.1.6.2

Create a Basic Announcement

This procedure describes how to create a *BasicAnnouncement* MO, that represents a single audio announcement file.

Prerequisites

- An Ericsson Command-Line Interface (ECLI) session in Exec mode is open.
- The announcement files are available on the announcement storage server or in the local storage.

Steps

1. In the MOM, navigate to **ManagedElement=1,MediaResourceFunction=1,Announcements=1,BasicAnnouncements=1** and create a *BasicAnnouncement* MO for each audio announcement file.
2. For each created *BasicAnnouncement* MO, define values for the following attributes:

- `announcementId`

An identity for the basic announcement. The combination of the `announcementId` and `languageCode` attributes form a unique identity for the basic announcement.

- `basicAnnouncementId`



The value component of the RDN.

- `fileName`

The name of the announcement file.

- `filePath`

The relative file path to the directory where the announcement file is stored. The value of this attribute is relative to the `announcement_storage_server_path` attribute in the deployment template. In case of local storage the value of the attribute is relative to `/cluster/storage/announcements`.

Example: `basic/en-GB/`

- `languageCode`

Language code of the basic announcement. The combination of the `announcementId` and `languageCode` attributes form a unique identity for the basic announcement.

Language code has the format of an RFC 5646 language tag.

Example: `en-GB`

The following attributes are optional:

- `duration`

The amount of time the announcement is to be played in milliseconds. If the specified duration is greater than the overall length of the announcement, the announcement is repeated until the duration is elapsed. The playing time is also dependent on the value of the iteration attribute. The selected playing time is always the shortest possible alternative.

- `iteration`

The number of times the announcement is repeated when played to the user.

- `userLabel`

Label for free use.

See table *Basic Announcement Attributes* for details.

3.1.6.3 Create a Variable Announcement

This procedure describes how to create a *VariableAnnouncement* MO, that represents a single variable announcement type.



Prerequisites

- An Ericsson Command-Line Interface (ECLI) session in Exec mode is open.
- The announcement files are available on the announcement storage server or in the local storage.

Steps

1. In the MOM, navigate to **ManagedElement=1,MediaResourceFunction=1,Announcements=1,VariableAnnouncements=1** and create a *VariableAnnouncement* MO for each variable announcement type.
2. For each created *VariableAnnouncement* MO, define values for the following attributes:
 - **variableAnnouncementId**
The value component of the RDN.
 - **logicFileName**
The name of the variable announcement logic file.
 - **logicFilePath**
The relative file path to the directory where the variable announcement logic file is stored. The value of this attribute is relative to the `announcement_storage_server_path` in the deployment template. In case of local storage the value of attribute is relative to `/cluster/storage/announcements`.

Recommended value: **variable/<languageCode>/**

Example: **variable/en-GB/**
 - **languageCode**
The language code of the variable announcement.

The combination of the `variableAnnouncementType` and `languageCode` attributes form a unique identity for the variable announcement.

The language code has the format of an RFC 5646 language tag.

Example: **en-GB**
 - **variableAnnouncementType**



Type of the variable announcement.

The combination of the `variableAnnouncementType` and `languageCode` attributes form a unique identity for the variable announcement.

The following attribute is optional:

- `userLabel`

Label for free use.

See table *Basic Announcement Attributes* for details.

3.2 Importing Configuration Data from NETCONF Template

An example initial configuration file is part of the vMRF software delivery package, with the name `example_config.tar.gz`.

The `tar.gz` file contains the following:

- An XML file that contains MO configuration values. The example `tar.gz` file contains an example XML file with the name `config.xml`. You must edit the XML file with values that match your environment, as described in the procedure below. As an optional service, Ericsson can provide a file that is already edited to match your environment.
- Node credentials: You must add the node credentials to the `tar.gz` file, as described in the procedure below.
- Trusted certificates: You must add the trusted certificates to the `tar.gz` file, as described in the procedure below.
- `manifest.yaml`: The file contains configuration metadata for the import and the contents of `/usr/share/image/package_build_data` for troubleshooting.

Note: You must manage the content of the `tar.gz` file using a Python script that is also delivered in the vMRF software delivery package. To run the script, you need Python installed on a Linux computer.

3.2.1 Create and Import Configuration Data

This procedure describes how to create and import the initial configuration data.



Steps

1. Extract the XML file from the tar file for editing using the `vmrs_config_update.py` script:

```
vmrs_config_update.py -c example_config.tar.gz -gxf
config.xml
```

2. Edit the XML file with specific values for your environment. For a list of MOs and attributes—that is, XML lines to check or edit—to configure, see [Initial Configuration Data for the vMRF Application](#) on page 14.

Note: The XML document is case-sensitive, pay attention when editing MO names and attributes.

3. Pack the XML file back into the `tar.gz` file, using the `vmrs_config_update.py` script:

```
.vmrs_config_update.py -c example_config.tar.gz -sxf
config.xml
```

4. Add the node credentials and trusted certificates to the tar file, and set the LDAP password using the `vmrs_config_update.py` script:

```
vmrs_config_update.py -c example_config.tar.gz -anc
key.pem cert.pem -atc <trusted_certificate>.pem ldap
<password>
```

5. Open an SSH connection to the O&M IP address of the vMRF VNF using the following command:

```
ssh <user ID>@<O&M IP address>
```

Emergency user credentials are needed to perform this step.

6. Copy the modified `tar.gz` file to the VNF using, for example, the `scp` command:

```
scp example_config.tar.gz <user ID>@<O&M IP address>:/
home/<user ID>
```

Result

The configuration file is copied from the current directory to the `/home/<user ID>` folder in the file system of the vMRF VNF.

7. Run the following command:

```
/opt/mrf_director/mrf_import_conf.py /home/<user ID>/
example_config.tar.gz
```

Result

The configuration data is imported and vMRF connects to the user plane and signaling networks.



3.2.1.1 Additional Security Configuration

Configure security as described in *vMRF Security Management*. Skip the installing node credentials and trusted certificates configuration steps, as they were already performed during the configuration data import procedure.

3.3 Initial Configuration Data for the vMRF Application

Table 2 vMRF Network Attributes

ManagedElement=1,MediaResourceFunction=1	
mediaIpVersion	Possible values: <ul style="list-style-type: none">• IPV4 (default)• IPV4_IPV6• IPV6

Table 3 vMRF H.248 Control Attributes

ManagedElement=1,MediaResourceFunction=1,MrfH248Control=1	
localPortNumber	

Table 4 vMRF Controlling Server Attributes

ManagedElement=1,MediaResourceFunction=1,MrfH248Control=1,MrfH248Interface=<interface ID>	
remoteIpAddress	IP address of the controlling server
administrativeState	State of H.248 link
remotePortNumber	Remote port of the controlling server

Table 5 Default Announcement Language Attributes

ManagedElement=1,MediaResourceFunction=1,Announcements=1	
defaultLanguageCode	Default language code for basic and variable announcements



Table 6 Basic Announcement Attributes

ManagedElement=1,MediaResourceFunction=1,Announcements=1,BasicAnnouncements=1,BasicAnnouncement=1	
announcementId	The identity of the basic announcement, combined with the <code>languageCode</code> attribute it forms a unique identity for the basic announcement.
basicAnnouncementId	The value component of the RDN.
duration	The amount of time the announcement is to be played
fileName	Announcement file name
filePath	Announcement file storage path
iteration	The number of times the announcement is to be repeated
languageCode	Basic announcement language code

Table 7 Variable Announcement Attributes

ManagedElement=1,MediaResourceFunction=1,Announcements=1,VariableAnnouncements=1,VariableAnnouncement=1	
languageCode	Variable announcement language code
logicFileName	The name of the variable announcement logic file
logicFilePath	Logic file storage path
variableAnnouncementId	The value component of the RDN.
variableAnnouncementType	Type of variable announcement

Table 8 SCTP Profile Attributes

ManagedElement=1,Transport=1,SctpProfile=1	
dscp	
heartbeatInterval	
enablePathMtuDiscovery	
initRto	



ManagedElement=1,Transport=1,SctpProfile=1
maxBurst
maxRto
maxSctpPduSize
maxInitRt
minRto
sackTimer
assocMaxRtx
pathMaxRtx
primaryPathMaxRtx
alphaIndex
betaIndex
cookieLife



4 Configure SNMP for Fault Management

The Simple Network Management Protocol (SNMP) provides alarm notifications.

The used SNMP version must be consistent with the SNMP targets used in the network fault management system. The recommended SNMP version is SNMPv3.

Do the following:

Steps

1. Create SNMP targets using one of the following instructions:

- *Create SNMPv1 Target*
- *Create SNMPv2C Target*
- *Create SNMPv3 Target*

Note: To prevent modification of the management information base (MIB) files through SNMP, the `isMibWritable` attribute of the SNMP target MOs is recommended to be set to `false` during the creation of the target.

2. SNMP views can be created to restrict read/write access to SNMP MIBs. It is recommended to grant SNMP access only to the ERICSSON-ALARM-MIB, that is needed for fault management purposes. This can be done by creating an SNMP view with `readOids` attribute set to 1.3.6.1.4.1.193.183 using the instruction *Create SNMP View*.

For the default SNMP settings, refer to the *Snmp* MO in the MOM.

For more information on fault management, refer to *Fault Management*.



5 Configure the networkManagedElementId Attribute

This procedure describes how to configure the `networkManagedElementId` attribute of the *ManagedElement* MO. The attribute changes the RDN visible in, for example, NETCONF transactions and PM files. This task is required for the OSS-RC interworking.

Prerequisites

An Ericsson Command-Line Interface (ECLI) session in Exec mode is in progress.

Steps

1. In the MOM, navigate to the *ManagedElement* MO and enter configure mode:

```
>ManagedElement=1
```

```
(ManagedElement=1)>configure
```

2. Modify the value of the `networkManagedElementId` attribute:

```
(config-ManagedElement=1)>networkManagedElementId=<value>
```

Note: The value must be unique within the network namespace.

3. Commit the changes:

```
(config-ManagedElement=1)>commit
```

Result

The job is completed.