

Deployment Guide for VMware vSphere

Virtual Multimedia Resource Function

Installation Instructions

Copyright

© Ericsson AB 2016–2019. 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.

Trademark List

All trademarks mentioned herein are the property of their respective owners. These are shown in the document *Trademark Information*.



Contents

1	About This Document	1
2	vMRF Deployment Principles for VMware	2
3	vMRF Deployment Process for VMware	3
4	Prerequisites for vMRF Deployment	6
4.1	Download and Extract vMRF Software Delivery Package	6
5	vMRF Deployment Preparations for the Cloud Administrator	7
5.1	Cloud Hardware and Software Preparation and Configuration	7
5.2	Create Network Topology	7
6	vMRF Deployment for the End User	9
6.1	Initial VNF Configuration Data for Deployment	9
6.2	Configure Shared Storage	9
6.3	Deploy the OVF Template	11
6.4	Power On vMRF vApp	14
6.5	Check vMRF Status	14





1 About This Document

This document describes vMRF deployment on a VMware cloud service. VMware service means VMware vSphere® including VMware ESXi® and VMware vCenter Server®.

The following user roles are distinguished in this document:

Cloud Administrator

The cloud administrator is the cloud service provider who delivers the cloud service to the end user. The cloud administrator must fulfill certain prerequisites before the end user can start deploying vMRF.

End User

The end user is the vMRF operator and deployment responsible, who is assumed to be a cloud service consumer on a vSphere cloud service. The end user is also referred to as a tenant.

2 vMRF Deployment Principles for VMware

If the hardware and software requirements are met, and after the needed configurations in VMware are done, vMRF is instantiated.

vMRF can contain one or more Virtual Network Functions (VNF).

A single VNF contains multiple Virtual Machines (VMs). See [Figure 1](#) for an example overview of deployment with two VNFs.

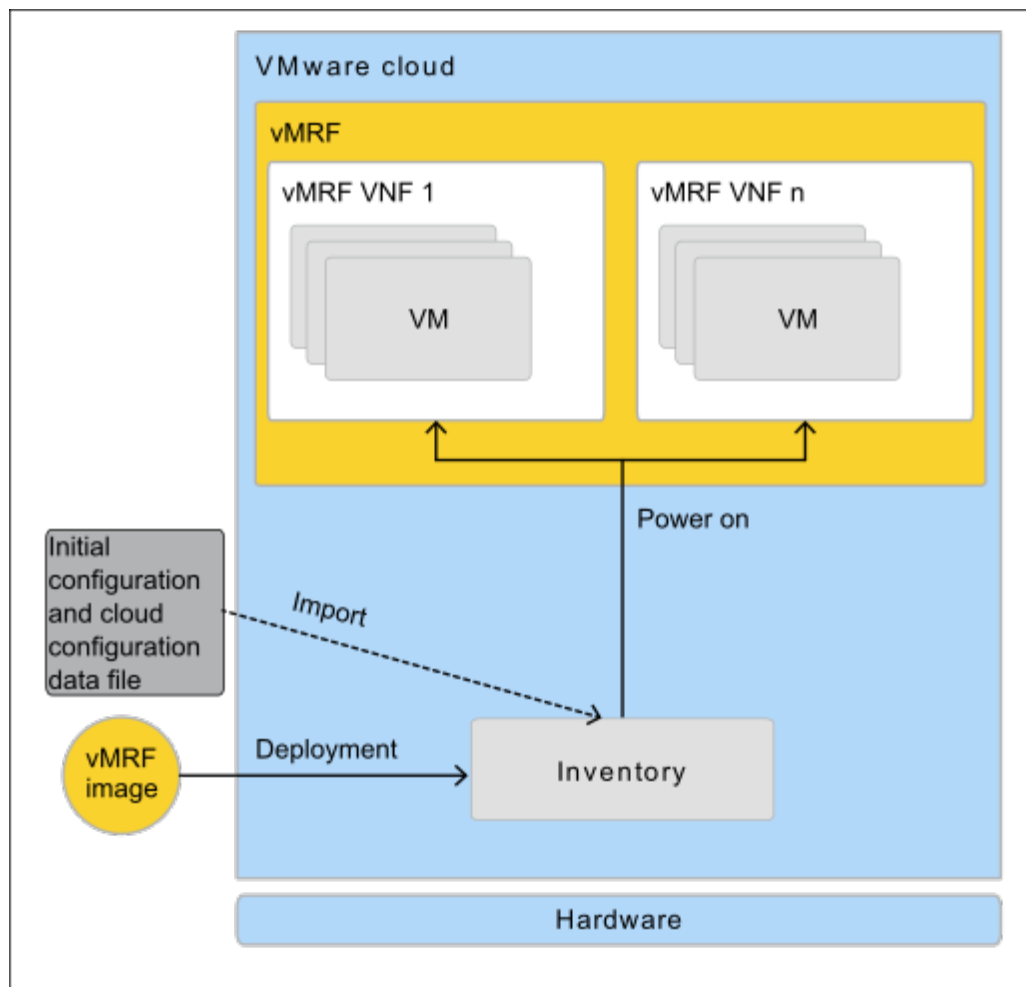


Figure 1 vMRF Deployment



3 vMRF Deployment Process for VMware

The vMRF deployment process consists of preparations and basic configuration of the cloud environment, and the actual instantiation of one or more vMRF VNF instances.

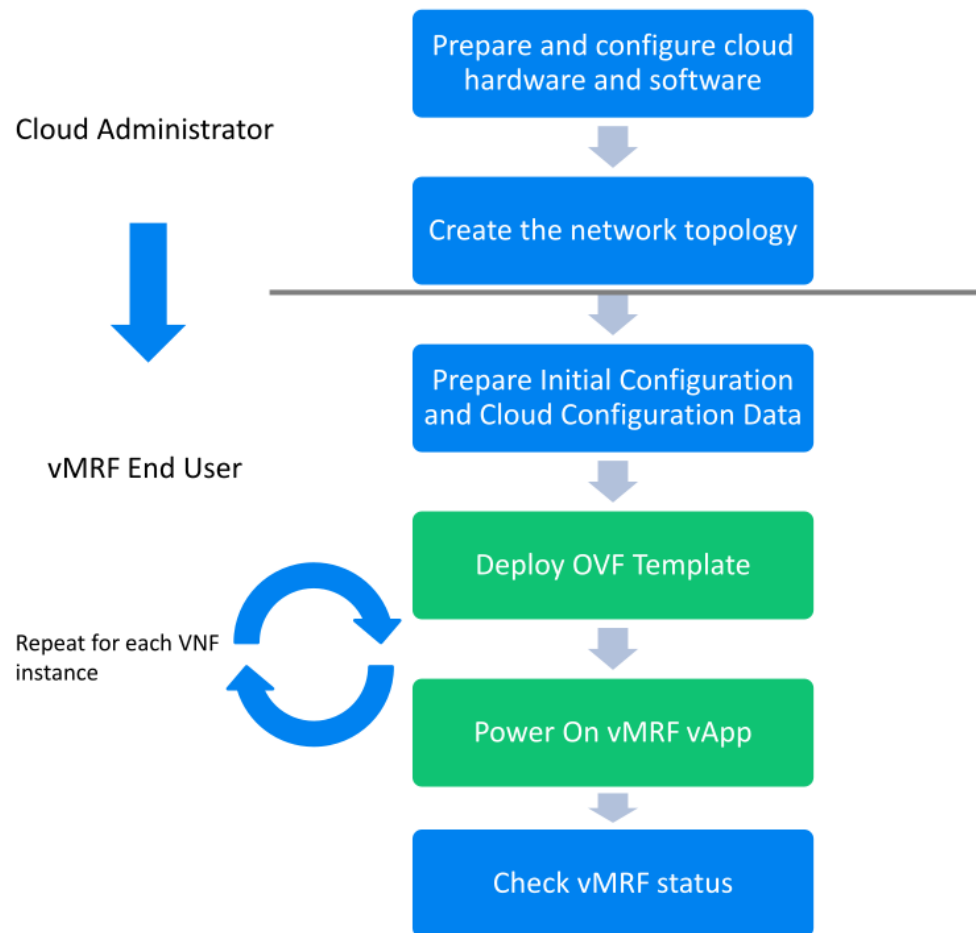


Figure 2 vMRF Deployment Process

1. Prepare the cloud environment to run vMRF

This set of steps is done by the cloud administrator.

a. Prepare and configure cloud hardware and software

This step involves checking that the necessary hardware exists, and making hardware-related configuration in VMware and in the host Operating System so that the requirements listed in [Prerequisites for vMRF Deployment](#) on page 6 are fulfilled.



b. Create the network topology

This step involves ensuring that the required networks to which the VNF connects are in place.

2. Deploy and check vMRF

This set of steps is done by the end user.

a. Download and extract the vMRF software delivery package

The vMRF software delivery package contains the Open Virtualization Format (OVF) template and the related files. The vMRF software delivery package must be extracted to a place where the files can be accessed by vSphere. vSphere offers the following options:

- A local directory that is accessible by the vSphere client
- The files can be uploaded to an HTTP server connected to the vCenter

b. Deploy OVF Template

The extracted OVF template must be deployed in VMware, which means that the VMs and related resources in the OVF template become visible in the inventory. This step is repeated for each VNF that needs to be created.

During OVF template deployment, cloud configuration data must be provided for the VNF. At the same time, initial configuration data can also be provided for the VNF. This makes it possible for the VNF to have all necessary configuration for processing traffic right after being created.

Note: If initial configuration data is not imported during OVF template deployment, it must be performed after power-on. For more information, see the [Initial Configuration Guide](#).

c. Provide Initial Configuration and Cloud Configuration Data

Initial configuration data means the data needed for vMRF to start processing traffic. It can either be provided during deployment in Base64 encoding, or imported in an iso file.

If you are importing this initial configuration data during deployment, you must prepare it so that it matches your environment.

For other options, see [Initial Configuration Guide](#).

d. Power On vMRF vApp

After powering on the vMRF vApp, vMRF starts running traffic. This step is repeated for each VNF instance that needs to be created.



e. Check vMRF status

It is recommended to run a status check on the newly deployed vMRF.



4 Prerequisites for vMRF Deployment

Before the end user can deploy and use vMRF, the cloud administrator must ensure that the environment fulfills hardware, software, and network requirements. The main requirements are listed in [vMRF Infrastructure Requirements](#).

4.1 Download and Extract vMRF Software Delivery Package

Before the deployment, the end user must download and extract the vMRF software delivery package. Both the end user and the cloud administrator must have access to the proper example files in the package.

Steps

1. Download the vMRF software delivery package to a computer from which the cloud service clients are reachable.
2. Extract the vMRF software delivery package.
3. Check that the following files exist after extracting the vMRF software delivery package:

- Deployment files (ovf files)

OVF File Name	Description
<code>vmrf.ovf</code>	OVF file for deployment with automatic IP address allocation from IP pools.
<code>vmrf_man_ip.ovf</code>	OVF file for deployment with manual IP address allocation.

- vMRF image (vmdk file)



5 vMRF Deployment Preparations for the Cloud Administrator

The procedures for vMRF deployment preparation must be performed by the cloud administrator to prepare the cloud environment for running vMRF. The procedures described in this section serve as examples only to demonstrate how to fulfill the vMRF requirements.

5.1 Cloud Hardware and Software Preparation and Configuration

Preparation for vMRF deployment starts by checking that the necessary hardware exists, and making hardware-related configurations in the VIM, in the hypervisor, and in the host Operating System.

5.1.1 Group Compute Nodes for vMRF into DRS Cluster

Perform this procedure only if you want to specify exactly which hosts can run vMRF due to, for example hardware considerations.

Steps

1. Create a Distributed Resource Scheduler (DRS) cluster for the hosts selected to run vMRF. For the details, see [Using DRS Clusters to Manage Resources](#) in the VMware documentation.
2. Use VM-Host affinity rules to define a relationship between vMRF VMs and the group of hosts selected to run vMRF. For the details, see [VM-Host Affinity Rules](#) in the VMware documentation.

5.2 Create Network Topology

The vMRF VNF instance connects to networks. The networks in [Table 1](#) must be created already before the VNF instance can be deployed, since the OVF template uses them as input parameters.

Table 1 vMRF Networks

Network Type
VNF-internal ⁽¹⁾
H.248 signaling towards MTAS
Management network



Network Type
User plane towards media networks

(1) Each VNF instance version requires a dedicated VNF-internal network.

Steps

1. Using the network plan, create the required networks listed in [Table 1](#), if they do not exist, and configure distributed port group parameter **VLAN type** in the vSphere Web Client.
 - For access vNIC setup, select **VLAN** and configure **VLAN ID**.

Note: The corresponding **VLAN ID** values must also be given in vMRF network level configuration data.

2. Create network protocol profiles and associate them with the networks created in [Step 1](#).

Depending on the IP allocation alternative, the following additional configuration options apply:

- `vmif.ovf`: network protocol profiles must be configured with IP pool, subnet, subnet mask length, and gateway, and associated to the internal and management networks in [Table 1](#).
- `vmif_man_ip.ovf`: network protocol profiles can be configured with subnet, subnet mask length, and gateway, and associated to the internal and management networks in [Table 1](#), or the information is provided during deployment.

In the case of manual IP address allocation, the use of network protocol profiles is mandatory if vSphere 6.5 is used, since vCenter 6.5 does not render prompting of subnet information properly during deployment.

3. Inform the personnel who are doing the vMRF deployment.



6 vMRF Deployment for the End User

After the deployment preparations are completed by the cloud administrator, the end user can start vMRF deployment.

6.1 Initial VNF Configuration Data for Deployment

Initial configuration data means the data needed for vMRF to start processing traffic. This procedure describes how to prepare initial configuration data if you are importing it during deployment. For other options, see [Initial Configuration Guide](#).

While providing cloud configuration data during deployment is mandatory, importing initial configuration is optional. It can be provided during deployment in Base64 encoding. The input can be generated with the following command:

```
base64 -w 0 mrsvconfig.tar.gz
```

If the size of the initial configuration data file `mrsvconfig.tar.gz` exceeds the 23 kB limit, it can be imported in an `iso` file. For more information, see [Create VNF Configuration ISO File](#) on page 9.

6.1.1 Create VNF Configuration ISO File

Initial configuration data must be imported to the VNF in an `iso` if it exceeds the limit imposed by vSphere and cannot be provided during deployment with cloud configuration data.

Steps

1. Rename the `exported_config.tar.gz` file to `mrsvconfig.tar.gz`.
2. Create an `iso` file which includes the `mrsvconfig.tar.gz` file. For example, on a Linux computer, use the following command:

```
genisoimage -l -iso-level 4 -o mrs-init.iso mrsvconfig.tar.gz
```

6.2 Configure Shared Storage

Note: This procedure is optional. Perform the steps only if shared storage is used.

The external shared storage allows for the storage of the following files from each VM on a remote server:



- Log files from the `/var/log` directory, including journal log files
- Crash dump files
- Configuration backup files created by the automatic backup and restore function

vMRF connects to the server with SSHFS, mounts a specified directory path, and creates a subfolder for the cluster, and subfolders for the files for each VM in the cluster. This ensures that logs and other shared files of different VMs and VNF instances do not get mixed up.

For authentication, an SSH key pair has to be created. This key pair can be cluster-specific, or common for all clusters.

The following parameters must be prepared and included in the `example_environment.yaml` file parameter list or the vApp property list during deployment:

- Storage server username
- Storage server address (IP address and port number)
- Storage server path
- Storage server SSH private key and fingerprint

Prerequisites

- The private and public SSH keys are generated.

Steps

1. Generate the private SSH key parameter value by replacing the end of line characters with the string `"\n"` and including the key data string between single quotation mark (') characters using the following command:

```
echo "'$(awk 'BEGIN{ORS="\n"} {print $0}' .ssh/id_rsa)'"
```

Note: The resulting string must be added in OpenStack-based deployments as the `shared_storage_ssh_private_key` value in the `example_environment.yaml` file or, in VMware-based deployments as vApp property.

2. Generate the SSH fingerprint parameter value from the public SSH key by replacing the end of line characters with the string `"\n"` using the following command:

```
echo "'$(ssh-keyscan -p <server_port> <server_host> |awk 'BEGIN{ORS="\n"} {print $0}')
```



Note: The resulting string must be added in OpenStack-based deployments as the `shared_storage_server_fingerprint_key` value in the `example_environment.yaml` file or, in VMware-based deployments as vApp property.

3. On the shared storage server, append the public key to the authorized keys file using the following command:

```
cat .ssh/<public_key_file_name> >> .ssh/<authorized_keys_file_name>
```

— RELATED INFORMATION —

[6.3 Deploy the OVF Template on page 11](#)

6.3 Deploy the OVF Template

This procedure describes how to upload the vMRF VMs to the VMware vSphere inventory by deploying the vMRF OVF template.

Steps

1. Deploy the OVF template based on the instructions in the VMware documentation for the [vSphere web client](#). For the deployment options, use the following information:

- a. For IP allocation, depending on the IP address allocation method, select one of the following OVF files:

OVF File Name	Description
<code>vmrf.ovf</code>	OVF file for deployment with automatic IP address allocation from IP pools.
<code>vmrf_man_ip.ovf</code>	OVF file for deployment with manual IP address allocation.

- b. For destination network mapping, use the information from [Create Network Topology](#) on page 7.
- c. Based on the IP address allocation method chosen for the deployment, provide the following information as a minimum in the interactive window:

For manual IP address allocation, provide the following vApp properties:

vApp Property Type	vApp Property
Emergency user credentials	Username



vApp Property Type	vApp Property
	Password hash. The default password must be changed by supplying a new password hash in this parameter.
	Ssh public key
O&M IP address of the VNF	Movable IP address, an IPv4 address in dot-decimal notation
NTP server IP address	NTP IPs, that is, the list of NTP server addresses separated by a space character
Shared storage configuration ⁽¹⁾	Shared storage server username
	Shared storage server path ⁽²⁾
	Shared storage server IP
	Shared storage server port
	Shared storage server fingerprint ⁽³⁾
	Shared storage ssh private key ⁽⁴⁾
Announcement storage configuration ⁽¹⁾	Announcement storage server username
	Announcement storage server path ⁽⁵⁾
	Announcement storage server IP
	Announcement storage server port
	Announcement storage server fingerprint ⁽³⁾
	Announcement storage ssh private key ⁽⁴⁾
PM data monitoring configuration ⁽¹⁾	Pm data monitoring hosts IP address
	Pm data monitoring hosts port
Initial configuration ⁽¹⁾	Initial configuration
Gateways and subnet network properties to all networks, if network profiles are not associated ⁽⁶⁾	Netmask
	Gateway



- (1) Optional property type.
- (2) The external shared storage POSIX path pointing to a directory on the shared storage server.

Note: vMRF creates VM specific folders under this path. In order to separate VMs of different vApps, include vApp identifier in this path.

- (3) The input can be generated by replacing the end of line characters with the string "\n" and including the key data string between single quotation mark (') characters with the following command: `echo "$(ssh-keyscan -p <server_port> <server_ip_address> | awk 'BEGIN{ORS="\n"} {print $0}')'".`
- (4) The input can be generated by replacing the end of line characters with the string "\n" and including the key data string between single quotation mark (') characters using the following command: `echo "$(awk 'BEGIN{ORS="\n"} {print $0}' .ssh/id_rsa)'".`
- (5) The announcement shared storage POSIX path pointing to a directory on the shared storage server.
- (6) The vCenter 6.5 deployment wizard shows property fields for **Gateway** and **Netmask** under section **Networking Properties** on the **Customize template** step. These are related to the vCenter 6.5 rendering problem, and can be left empty.

Provide the following VM properties:

VM Property Type	VM Property
IPv4 or IPv6, or both type of addresses for media networks	Media IPv4 address
	Media IPv6 address
	Trusted IPv4 address
	Trusted IPv6 address
IPv4 addresses for the following networks: VNF-internal, O&M, signaling	Internal IP address
	Management IP address
	Signaling IP address
Media network subnet information	Media IPv4 Subnet mask length
	Media IPv6 Subnet mask length
	Trusted IPv6 Subnet mask length

Result: The vMRF vApp with one VM becomes visible in the inventory.

2. For possible future vMRF VNF deployments, create a vApp template from the vMRF vApp.

If initial configuration is going to be provided during deployment and it has not already been provided in Base64 encoding, perform the following step, otherwise continue with [Step 5](#).

3. Upload the iso file created in [Create VNF Configuration ISO File](#) on page 9.



4. Mount the `iso` file on the VM as a CD/DVD image in the vApp.
5. Clone the VM to a template.

6.4 Power On vMRF vApp

This procedure describes how to power on the vMRF vApp that is now in the inventory. After powering on the vMRF vApp, vMRF starts running traffic.

Steps

1. Navigate to the VM in the vMRF vApp.
2. Right-click the VM and select **Power On**.

If a delay is set in the startup settings of the VM, the VM is powered on only after the set length of time.
3. Check that the VM works correctly as described in [Check vMRF Status](#) on page 14.
4. Use the VM template created in [Step 5 in Section 6.3](#) to deploy the required number of VMs.
5. If manual IP address allocation is chosen, modify VM properties so that each VM has unique IP addresses.
6. Power on all the newly created VMs..
7. Modify the **Shutdown Action** vApp setting to **Guest Shutdown** for each VM in **vApp > Edit Settings > Authoring Session > Start Order**.
8. Continue with [Check vMRF Status](#) on page 14 to verify the whole VNF

6.5 Check vMRF Status

This procedure describes how to verify the vMRF deployment. The status check involves running a vMRF command.

Steps

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

```
ssh -A <user_ID>@<O&M_IP_address>
```

2. Run the following command to check the status of VMs in the cluster:

```
verify_vmrf_cluster_status.py
```



- If the VMs are operating correctly, the OK status is displayed in the command printout. You can exit this procedure.
 - If there are VMs with faulty components, a list of faulty VMs and detailed component information of the cluster is displayed. Continue with the next step.
3. Check all components with erroneous state. For specific trouble cases and remedies, refer to the [vMRF Troubleshooting Guideline](#).

Note: The `MrfDirector` and `COM` components are in the OK state only for the VM whose `SC` role is `ACTIVE`, that is, the active System Controller (SC) VM. In all other VMs, these components are in the `OK, NOT RUNNING` state, which is normal behavior.