Dell EMC SRDF/Metro

vWitness Configuration Guide

REVISION 04

DELL EMC

Copyright © 2016-2018 Dell Inc or its subsidiaries All rights reserved.

Published November 2018

Dell believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED “AS-IS.“ DELL MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. USE, COPYING, AND DISTRIBUTION OF ANY DELL SOFTWARE DESCRIBED IN THIS PUBLICATION REQUIRES AN APPLICABLE SOFTWARE LICENSE.

Dell, EMC, and other trademarks are trademarks of Dell Inc. or its subsidiaries. Other trademarks may be the property of their respective owners. Published in the USA.

Dell EMC
Hopkinton, Massachusetts 01748-9103
1-508-435-1000 In North America 1-866-464-7381
www.DellEMC.com

CONTENTS

FIGURES

CHAPTER 1

Product Overview

This chapter introduces SRDF/Metro and its resiliency features:

Introduction

SRDF/Metro changes SRDF behavior to better achieve the high availability requirements of today's applications. In traditional SRDF, only R1 (source) devices are Read/Write accessible to the host, while R2 (target) devices are Read Only/Write Disabled to the host. With SRDF/Metro:

- R2 devices are Read/Write accessible to the host.
- Hosts can write to both the R1 and R2 side of the device pair.
- R2 devices assume the same external device identity (such as, geometry and device WWN) as their R1 partners.

This shared identity means that the R1 and R2 devices appear to hosts as a single virtual device.

In the event that one or more device pairs become Not Ready (NR) or connectivity is lost between the arrays, SRDF/Metro must decide which side of the pair remains accessible to the hosts. There are two mechanisms that SRDF/Metro can use when making this decision: Device Bias and Witness.

Device Bias
Device pairs for SRDF/Metro are created with a *bias* attribute. By default, the create pair operation sets the bias to the R1 side of the pair. That is, if a device pair becomes Not Ready NR on the RDF link, the R1 (bias side) remains accessible to the hosts, and the R2 (non-bias side) becomes inaccessible. However, if there is a failure on the R1 side, the host loses all connectivity to the device pair. The Device Bias method cannot make the R2 device available to the host.

Witness
A third party mediates between the two arrays helping to:

- Decide which side remains available to the host
- Avoid a "split brain" scenario where both arrays attempt to remain accessible to the host despite the failure

There are two forms of the Witness mechanism.

- **Array Witness**: The operating environment on a third array is the mediator.
- **Virtual Witness**: A daemon running on a separate, virtual machine is the mediator. This method is available in PowerMaxOS 5978 and HYPERMAX OS 5977.945.890 or later.

The Array Witness method provides the highest availability. However, the added requirement of a third array may prevent its use in some environments. Virtual Witness, on the other hand, provides similar functionality and availability as Array Witness, without the need for a third array.

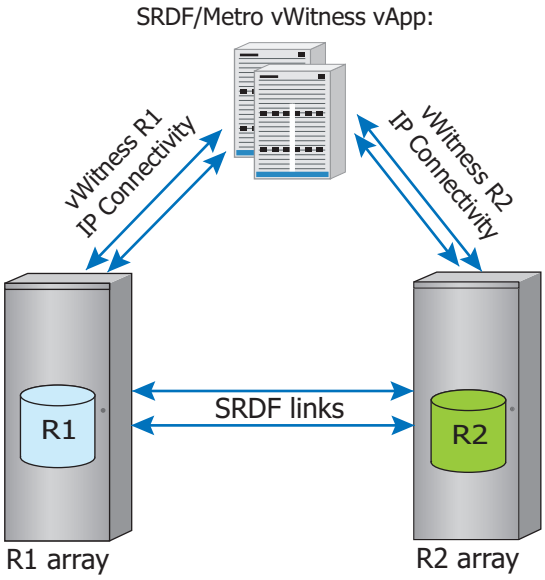
This guide shows how to configure and manage SRDF/Metro with the Virtual Witness option.

The *Dell EMC PowerMax Family Product Guide*, *Dell EMC VMAX All Flash Product Guide for VMAX 250F, 450F, 850F, 950F with HYPERMAX OS* , *EMC VMAX3 Family Product Guide for VMAX 100K, VMAX 200K, VMAX 400K with HYPERMAX OS* and the *Dell EMC Solutions Enabler SRDF Family CLI User Guide* provide more information on SRDF/Metro.

Virtual Witness (vWitness)

Virtual Witness (vWitness) is a resiliency option that is packaged to run in a virtual appliance (vApp) on a VMware ESX server. There can be up to 32 vApps, each providing a *vWitness instance*.

Figure 1 SRDF/Metro vWitness vApp and connections



The R1 and R2 arrays each contain a user-defined list of *vWitness definitions* that identify the vWitness instances that each array can use. A vWitness definition consists of a user-specified name and the location of the instance (either the IP address or the fully-qualified DNS name). The lists of vWitness definitions on each array do not have to be identical. However, they must have at least one instance in common. Initially, the R1 and R2 arrays negotiate which vWitness instance to use from the list of vWitness definitions that both arrays have in common.

Should the SRDF links between the R1 and R2 arrays fail, or one of the arrays has a serious problem, the vWitness instance helps to determine which array remains available to the host or hosts.

Unisphere for PowerMax, Unisphere for VMAX and SYMCLI provide facilities to manage a vWitness configuration. The user can add, modify, remove, enable, disable, and view vWitness definitions on the arrays. In addition, the user can add and remove vWitness instances. To remove an instance, however, it must not be actively protecting SRDF/Metro sessions.

Functional overview

A vWitness instance is a daemon process, known as the vWitness Lock Service (or vWLS), running in a vApp. On the R1 and R2 arrays, another daemon, known as the vWitness manager (or vWM), runs on both management guests (for redundancy) and acts as a proxy between the arrays and the vWitness instances (the vWLS instances).

Selecting a vWitness instance

Activity between a pair of SRDF/Metro groups is known as a SRDF/Metro session. When a session starts, the R1 and R2 arrays negotiate which of the available vWitness instances to use to protect the session. Thus, an individual array could be using several vWitness instances simultaneously. In the same way, an individual vWitness instance may be monitoring several SRDF/Metro sessions simultaneously.

Monitoring the connections to vWitness instances

vWM on each array polls all of the vWitness instances in its definition list every second. Each vWLS daemon sends a reply. This enables vWM to maintain the list of instances that are available and operational.

If an array detects that an instance has not responded for 10 seconds it checks whether the instance is in use by any SRDF/Metro session. If it is in use, the R1 and R2 arrays negotiate an alternative witness to use in its place. If there are no witnesses available, the session uses Device Bias as a fallback.

Acting on a systems failure

If either array detects that a session has failed (that is, the array has lost contact with the partner group either due to a failure of the SRDF link or in the partner array), it asks the vWM to request a lock from the vWitness instance allocated to the SRDF/Metro session.

On the R1 side, vWM sends the request to the vWitness instance for that session immediately. Typically, vWM waits 5 seconds before sending the request on the R2 side. This allows time for the R1 side to request the lock. That is, R1 has priority and gets the lock if it asks for it during this 5 second period.

The vWitness instance grants the lock in response to the first request it receives. The side that gains the lock remains available to the host while the other side becomes unavailable.

Determining the preferred winner

In addition to determining which vWitness instance to use, the arrays in each SRDF/Metro session also negotiate which of them is the preferred winner. In the event of a failure, the preferred winner is the side that has priority when requesting the lock from the vWitness instance; that is, the preferred winner is the R1 side.

When either side runs HYPERMAX OS 5977, SRDF/Metro uses the bias settings for the devices to determine the preferred winner. That is, the devices defined as the being on the bias side, if Device Bias were to be used, become the preferred winners.

When both sides run PowerMaxOS 5978, SRDF/Metro takes these, additional factors into account to determine the preferred winner (in priority order):

1. Which side has a SRDF/A DR leg
2. Whether the SRDF/A DR leg is synchronized
3. Which side has more than 50% of the RA or FA directors that are available

4. The side that is currently the bias side

The first of these criteria that one array has and the other does not stops the selection process. The side with the matched criteria is the preferred winner.

The two sides repeat this selection process regularly for each SRDF/Metro session to ensure that the winner remains the one that is most preferable. This means that the winning side may change during the course of a session.

vWitness benefits

vWitness provides the following benefits:

- Provides a similar level of high availability as the Array Witness option, without requiring a third array.
- Multiple vWitnesses can be configured for redundancy.
- IP connections between each vWitness and the arrays are secured using TLS/SSL.
- vWitness and Array Witness options can be used together.

Witness failure scenarios

This section depicts various single and multiple failure behaviors for SRDF/Metro when the Witness option (Array or vWitness) is used.

Figure 2 SRDF/Metro Witness single failure scenarios

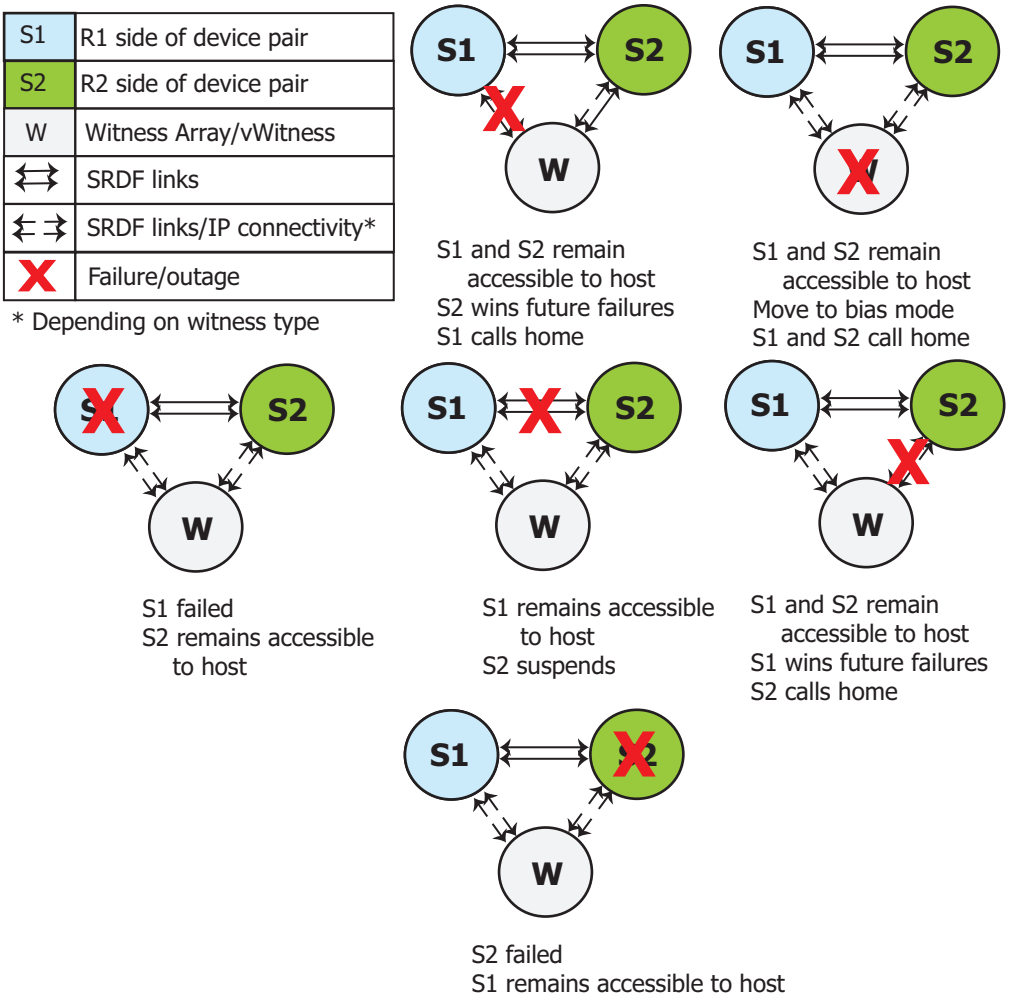
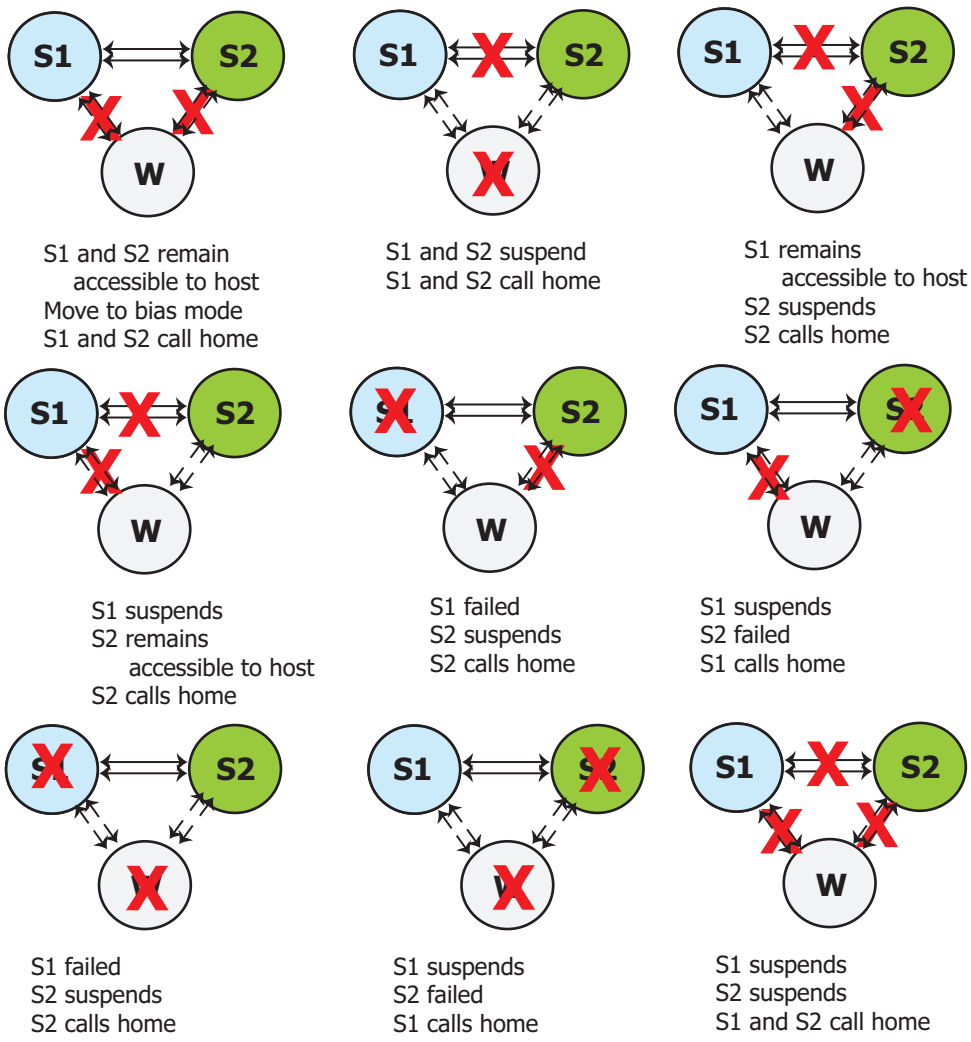


Figure 3 SRDF/Metro Witness multiple failure scenarios



CHAPTER 2

Install and configure

This chapter shows how to install and configure a vWitness instance:

Preparation

System architecture guidelines

This section provides guidance on deploying vWitness facilities with SRDF/Metro.

Preferred configuration

The minimum, preferred configuration is:

- The primary (R1) and secondary (R2) arrays are on separate sites with each array in its own fault domain (to include network and power domains).
- At least two vWitness instances for each SRDF/Metro configuration on separate sites:
  - Place each instance in a separate fault domain (to include network and power domains).
  - Avoid placing any instance in the same fault domain as the SRDF/Metro configuration it protects.
- The vWitness instances have independent network connectivity (latency < 40 ms) to both the primary and secondary sites.

This configuration can withstand most failures including communications failure between the primary and secondary sites. It also prevents a split brain scenario from occurring.

For additional resilience, consider enhancing this minimum configuration with additional vWitness installations (see Quantity of vWitness installations on page 17).

Alternative configurations

Instead of separate sites and fault domains for witness installations, you might consider multiple, local witness installations at both the primary and secondary sites. While this works in most situations, it has a major drawback that can result in both sides of a SRDF/Metro pair being unavailable to the host even though one side is fully functional.

For example, an SRDF/Metro session is using a witness located at the primary site. A failure at that site causes the witness and the SRDF links to become unavailable. This is a split brain situation. Since the SRDF links are unavailable the arrays cannot negotiate a witness to use in place of the one that failed. So, the arrays fall back to using Device Bias which results in the array at the secondary site becoming unavailable to the host (since it is not on the bias side). If the failure at the primary site has also affected the primary array, the result is that the host has no access to either array in the storage pair.

For this reason you are recommended to have each witness in its own fault domain separate from those used by the arrays in the SRDF/Metro configuration.

Quantity of vWitness installations

- Have at least two vWitness instances for every SRDF/Metro configuration.
- Spread vWitness instances over multiple servers, where possible, to avoid having them all run on one server (which creates a single point of failure).
- Where possible, have the vWitness installations at sites separate from the arrays that participate in SRDF/Metro sessions. This helps to ensure that a failure at one site affects only one of the arrays or a vWitness.
- Ensure that there are a sufficient number of witnesses to protect the number of SRDF/Metro groups that you intend to have active.
  The maximum number arrays, array pairs, and SRDF/Metro groups that a single vWitness instance can serve are:

Arrays:	32
Array pairs:	16
SRDF/Metro groups:	250

Hardware and software requirements

Gather the storage systems, VMware ESX servers, and software necessary to create a vWitness configuration.

Storage systems

The requirements for storage systems are:

- Two storage arrays running PowerMaxOS 5978 or HYPERMAX OS 5977.945.890 and later.
- SRDF/Metro license installed on each array.
- eManagement guest for Unisphere on each array. eManagment is standard on PowerMax and VMAX All Flash arrays, and can be added to VMAX3 arrays in the field. Contact your Dell EMC representative for more information.
- RA (Fibre/SAN) or RE (Ethernet/IP) connectivity between the paired arrays.
- Ethernet/IP connectivity between each array and each vWitness instance it uses.

VMware ESX servers

Ensure that each ESX server you want to use for vWitness operations meets these requirements:

- VMware ESX 4.0 or higher
- Depending on the vApp, the host must meet the following:
  - Solution Enabler Virtual Appliance: Single processor with 2 GB of memory; dual disks, with 16 GB of disk space and 5 GB of expandable disk space
  - Unisphere for PowerMax or Unisphere for VMAX: Dual core processor with 16 GB of memory and 120 GB of disk space

In addition, you require a client system, that you use to access the ESX servers, with the following:

- VMware vSphere Client
- Any of the following browsers with cookies and JavaScript enabled:
  - Internet Explorer 9.0 through 11.0 (Desktop only)
  - Firefox 30 or later
  - Chrome 21.0.1180 or later

  Browsers should have Flash Player 11.2 plug-in installed. If the browser has an older version of Flash Player, you are prompted to download the latest version when you start the web console.

Other Dell EMC software

To install and manage a vWitness configuration requires the following additional software:

- Solutions Enabler 8.3 or later
- Unisphere for PowerMax 9.0 (optional)
- Unisphere for VMAX 8.3 (optional)

vWitness instances

Decide on the number of instances of vWitness for your site. For each instance:

- Decide which ESX server the instance runs on.
- Gather the IP address or the fully-qualified DNS name of the vApp that runs the instance.
- Decide on a name for the instance.
  - The name has up to 12 characters and starts with an alphabetic character.
  - The remainder of the name can contain alphanumeric characters, underscores, and hyphens.
  - The name is not case sensitive, but the system preserves the case.

TCP ports

Ensure that the following TCP ports are open and available for use by vWitness instances and the SRDF/Metro storage systems:

System	Port number	Usage
VMware ESX server	10123	vWitness Lock Service
	5480	vApp Manager
Embedded Element Manager on a SRDF/Metro storage system	5480	vApp Manager

TLS certificates

Each vWitness instance is supplied with TLS security certificates. However, you can replace all of these with site-specific certificates if required. To apply custom certificates, gather the following files in Privacy Enhanced Mail (PEM) format:

- Certificate
- Key
- Trust certificate

Store the files at `/var/symapi/config/cert` on the client system.

Use the same trust certificate to generate all custom certificates.

Installation kit

Obtain the installation kit for the vWitness instances from Dell EMC Online Support. You need one of:

- The Solutions Enabler Virtual Appliance (vApp)
- The Unisphere for PowerMax vApp
- The Unisphere for VMAX vApp

The virtual appliance runs on the ESX server creating the vWitness instance.

Put the OVF archive file (`*.ova`) in a temporary directory on the system that runs the vSphere Client.

Install the vWitness instances

Note

This section shows just one way of installing the vWitness instances that use the Solutions Enabler Virtual Appliance. The *Dell EMC Virtual Appliance Manager Installation Guide* shows other ways of installing the instances packaged in either Virtual Appliance.

To install each of the vWitness instances that your site requires:

1. Import the Virtual Appliance.
2. Power on and configure the Virtual Appliance.

Import the Virtual Appliance

Procedure

1. Start the vSphere Client and log in to the ESX Server on which you are installing the appliance.
2. Click **Ignore** in the security warning message.
3. From the **File** menu, select **Deploy OVF Template**.
4. Browse to the OVF archive file, which is located in the temporary directory you created earlier. Select the OVF archive file with the suffix `*vappxxx_xxx_OVF10.ova`.
5. Click **Next**.
6. On the **Details** page, verify the details about the appliance and click **Next**.
7. On the **End User License Agreement** page, select **Accept all license agreements** and click **Next**.
8. On the **Name and Location** page, specify a name for the appliance and click **Next**.
9. If a resource pool is available, the **Resource Pool** page displays. Select the resource pool of the choice and click **Next**. Otherwise, the **Resource Pool** page is skipped.
10. On the **Datastore** page, select the data store of the choice and click **Next**.
11. On the **Disk Format** page, select the format in which to store the virtual machine's virtual disks and click **Next**.
12. On the **Network Mapping** page, map the source network to the appropriate destination network.
13. On the **Ready to Complete** page, verify the information and click **Finish**.
14. In the Completed Successfully message, click **Close**.

Power on and configure the Virtual Appliance

Procedure

1. On the **Summary** page of the Virtual Infrastructure Client, click **Power On**.
2. Click the **Console** tab and watch as the appliance starts up.
3. At the following prompts, type static IP configuration information:

```
Please select your static network configuration.
For IPv4: Enter 1
For IPv6: Enter 2
Enter your choice [1]/2:
```

```
Please enter static IP configuration:
```

- `IP Address [ ]:`
  Type the address that is assigned to the appliance, and then type **`y`** when asked to `Confirm [y]/n` and continue with the configuration.

  Note

  The virtual appliance uses this IP address to query the DNS Server and get its hostname. Therefore, you must ensure that the IP address has a hostname mapping in the DNS Server.

- `Netmask [ ]:`
  Type the mask of the network on which the appliance will be running, and then type **`y`** when asked to `Confirm [y]/n` and continue with the configuration.
- `Gateway [ ]:`
  Type the gateway address to the network on which the appliance will be running, and then type **`y`** when asked to `Confirm [y]/n` and continue with the configuration.
- `DNS1 [ ]:`
  Type the first DNS server address, and then type **`y`** when asked to `Confirm [y]/n` and continue with the configuration.
- `DNS2 [ ]:`
  Type the second DNS server address, and then type **`y`** when asked to `Confirm [y]/n` and continue with the configuration.
- `Is a proxy server necessary to reach the Internet? y/n [n]:`
  A **[`y`]**es response enables you to specify the IP address of the proxy server and the port.

The network is configured at this point.

4. At the following prompt, specify whether you want to set the time zone:

```
Do you want to set the time zone? y/[n] :
```

A **[`n`]**o response continues the configuration. If you select this option, you can use the appliance console to specify the time zone at a later time.

A [`y`]es response produces the following series of prompts that enable you to set the time zone:

- `Please select a continent or ocean`
  Type the number that corresponds to the time zone location and press **Enter**.
- `Please select a country`
  Type the number that corresponds to the country-specific time zone you want to set and press **Enter**.
- `Please select one of the following time zone regions`
  Type the number that corresponds to regional time zone you want to set and press **Enter**.

The time zone is now set.

5. At the following prompt, specify whether you want to type the host ESX Server information:

```
Do you want to set the host ESX Server y/[n]? :
```

   - A **n** response continues the configuration. If you select this option, you can use the Configuration Manager to type the host ESX Server details at a later time. For instructions, refer to the Configuration Manager's online help.
   - A **y** response prompts you for the ESX Server hostname. In which case, you should type the fully qualified hostname of the ESX Server and press **Enter**. When prompted for the root password, type the root password of the ESX Server and confirm it by typing it again.

   A Welcome screen displays. You have now finished installing the Solutions Enabler Virtual Appliance.

Enable Secure Shell (SSH)

Procedure

1. Launch the vApp Manager by typing the following URL in a browser:

   `https://`*`appliance`*`:5480`

   Replace *appliance* with the IP address or fully-qualified DNS name of the appliance.

   The vApp Manager main window appears.

2. Log in to vApp Manager using `seconfig` for both the user name and password.
3. When prompted, change the password.
4. Select **Command Execution** > **Host** and click **Enable SSH**.

Import TLS certificates (optional)

To use custom TLS certificates for any vWitness instance, import them. Complete this procedure for each vWitness instance that has custom certificates. Carry out the procedure on both the Virtual Appliance and the eManagement guests.

Start the certificate management utility

Procedure

1. Start and log in to vApp Manager on the vWitness instance.
2. Click **Appliance Info** and in the **Operations** panel click **Certificate management for Solutions Enabler**.

   The tool to import certificates starts and displays an introductory screen.
3. Click **Next**.
4. Select **Import certificate** and click **Next**.
5. Click **Yes** in the restart confirmation dialog.

   The **Import Alternate Private Key** window appears.

Import the certificate files

Procedure

1. Click **Import** to open a file browser.
2. Navigate to the location of the certificate files, select the file that contains the private key and click **Open**.

   vApp Manager validates the key file.
3. Click **Next**.

   The **Import Alternate Certificate** windows appears.
4. Click **Import** to open a file browser.
5. Navigate to the location of the certificate files, select the file that contains the alternate certificate, and click **Open**.

   vApp Manager validates the certificate file.
6. Click **Next**.

   The **Import Custom Trust Certificate** window opens.
7. Click **Import** to open a file browser.
8. Navigate to the location of the certificate files, select the file that contains the trust certificate and click **Open**.

   vApp manager validates the trust certificate.
9. Click **Next**.

   vApp Manager imports the certificate files and restarts the storsrvd and storvwlsd daemons.
10. Click **Finish**.

Define the vWitness instances on the storage systems

Follow the instructions in Manage and monitor vWitness definitions on a storage array on page 26 to create vWitness definitions on each storage array that runs SRDF/Metro. You can use Unisphere or SYMCLI.

UTC time synchronization

The UTC time of the storage arrays and the vWitness instances need to be synchronized. The vApp that contains a vWitness synchronizes its time with the VMware ESX server. So, the UTC time setting on the physical host of that server and on the storage arrays must be synchronized. Use a facility such as the Network Time Protocol (NTP) to achieve this.

On the server host, use an NTP product to connect to a NTP server that provides time synchronization. On the storage arrays, use the vApp Manager for eManagement web interface to enable NTP:

1. In a web browser, navigate to https://*emanage-ip-addr*:5480.
   Replace *emanage-ip-addr* with the IP address of the eManagement facility on the storage array.
2. Log in to the vApp Manager for eManagement.
3. Click **IP configuration** and then click the **NTP** tab.
4. In the **NTP** box, type the address of the NTP server and then click **Set Config**.

More information on using vApp Manager for eManagement is available from its online help.

Update a vWitness instance

The *Dell EMC Virtual Appliance Manager Installation Guide* shows how to install updates to a vWitness instance that uses the Solutions Enabler Virtual Appliance.

Note

In configurations that include storage arrays running HYPERMAX OS, the version of the Solutions Enabler Virtual Appliance that runs a vWitness instance must be the same or greater than the version of the eManagement Solutions Enabler that runs on the storage array. So, if you are performing an upgrade to HYPERMAX OS that includes an upgrade to the eManagement Solutions Enabler, ensure that you upgrade the Solutions Enabler Virtual Appliance beforehand. This requirement does not apply to storage arrays that run PowerMaxOS.

CHAPTER 3

Manage and monitor

This chapter shows how to manage and monitor a vWitness configuration:

Manage and monitor vWitness definitions on a storage array

This section shows how to set up, manage, and monitor a storage array's access to vWitness instances. You can use Unisphere or SYMCLI commands.

Note

When you create a vWitness definition the system does not check whether the final IP address of the instance is reachable from the array that holds that definition.

Unisphere for PowerMax

User roles

- To create, enable, modify, delete, or disable a vWitness definition you require the StorageAdmin or Administrator roles.
- To view vWitness definitions requires at least the PerformanceMonitor role.

Procedure

1. Select the storage array from the **System Selector** on the **Home Dashboard**.
2. Select **DATA PROTECTION** > **Virtual Witness**.
3. Follow the instructions for the vWitness definition task you want to complete:

Task	What to do
Create	a. Decide on a name for the definition. • The name has up to 12 characters and starts with an alphabetic character. • The remainder of the name can contain alphanumeric characters, underscores, and hyphens. • The name is not case sensitive, but the system preserves the case. b. Obtain the IP address or the fully-qualified DNS name of the system where the vWitness instance is installed. The address or name has a maximum of 128 characters. c. Click **Create**. d. Type the **Virtual Witness Name** and the **IP/DNS**. **Note** Create only one definition for each vWitness instance, specifying either the IP address or the fully-qualified DNS name of the instance. e. To simultaneously add this definition to other arrays, select the **Add Virtual Witness** checkbox and select the other arrays. f. Expand the list in the **Add to Job List** button and click **Run Now**. Unisphere creates the new definition and enables it.
Enable	a. Select the vWitness definition and then click **Set State**. b. Expand the list in the **Add to Job List** button and click **Run Now**. Unisphere enables the definition.

Task	What to do
Modify	a. Disable the definition. b. Select the vWitness definition and click **Delete**. c. Check that the confirmation dialog identifies the correct vWitness definition. d. Expand the list in the **Add to Job List** button and click **Run Now**. e. Click **Add**. f. Type the modified **Virtual Witness Name** and **IP/DNS**. **Note** Create only one definition for each vWitness instance, specifying either the IP address or the fully-qualified DNS name of the instance. g. Expand the list in the **Add to Job List** button and click **Run Now**.
Delete	a. Disable the definition. b. Select the vWitness definition and click **DELETE**. c. Check that the confirmation dialog identifies the correct vWitness definition. d. Expand the list in the **Add to Job List** button and click **Run Now**.
Disable	a. Select the vWitness definition and then click **Set State**. b. If necessary, click **Advanced Options** and set one of: • **Use Force** if the selected vWitness instance is in use and there is another witness (physical or virtual) available to take over. • **Use SymForce** if the selected vWitness instance is in use and there is no other witness (physical or virtual) to take over. c. Expand the list in the **Add to Job List** button and click **Run Now**.
View	When you select a vWitness definition, Unisphere displays the definition's properties : • Name • State • Indicators of whether the definition is in alive and in use Click ⓘ to view more detailed information: • Name • IP address or DNS name • Port • State • Indicators if whether the definition is alive and in use • The number of SRDF groups that are using the instance

Unisphere for VMAX

User roles

- To add, enable, modify, remove, or disable a vWitness definition you require the StorageAdmin or Administrator roles.
- To view vWitness definitions requires at least the PerformanceMonitor role.

Procedure

1. Select the storage array from the **System Selector** on the **Home Dashboard**.
2. Select **Data Protection** > **Replication Groups and Pools**.
3. On the **Replication Groups and Pools** page click **SRDF Virtual Witnesses**.
4. Follow the instructions for the vWitness definition task you want to complete:

Task	What to do
Add	a. Decide on a name for the definition. • The name has up to 12 characters and starts with an alphabetic character. • The remainder of the name can contain alphanumeric characters, underscores, and hyphens. • The name is not case sensitive, but the system preserves the case. b. Obtain the IP address or the fully-qualified DNS name of the system where the vWitness instance is installed. The address or name has a maximum of 128 characters. c. Click **Add**. d. Type the **Witness Name** and the **IP/DNS**. **Note** Create only one definition for each vWitness instance, specifying either the IP address or the fully-qualified DNS name of the instance. e. Click **Run now**. Unisphere creates the new definition and enables it.
Enable	a. Either: • Select the vWitness definition and then click **Set Status**. • Right click on the vWitness definition and select **Set Status** on the context menu. b. Click **OK**.
Modify	a. Disable the definition. b. Select the vWitness definition and click **Remove**. c. Check that the confirmation dialog identifies the correct vWitness definition, then click **OK**. d. Click **Add**. e. Type the modified **Witness Name** and **IP/DNS**.

Task	What to do
	Note Create only one definition for each vWitness instance, specifying either the IP address or the fully-qualified DNS name of the instance. f. Click **Run Now**.
Remove	a. Select the vWitness definition and click **Remove**. b. Check that the confirmation dialog identifies the correct vWitness definition, then click **OK**.
Disable	a. Either: • Select the vWitness definition and then click **Set Status**. • Right click on the vWitness definition and select **Set Status** on the context menu. b. If necessary, set one of: • **Use force** if the selected vWitness instance is in use and there is another witness (physical or virtual) available to take over. • **Use SymForce** if the selected vWitness instance is in use and there is no other witness (physical or virtual) to take over. c. Click **Run Now**.
View	When you click **SRDF Virtual Witnesses**, Unisphere displays a list of the vWitness definitions on the selected storage system. For each vWitness definition, the system displays its properties including: • Name • IP address or DNS name • State • Indicators of whether the definition is in alive and in use In addition you can view the details of a vWitness definition and the RDF groups associated with it: a. Select a vWitness definition. b. Click **View details**. The details of the vWitness definition are in the **Properties** panel, and the RDF groups associated with this vWitness definition are in the **Related Objects** panel.

SYMCLI commands

You can use SYMCLI commands to set up, manage, and view vWitness definitions.

Command syntax convention
The sections showing the syntax of the commands use square brackets [and] to enclose optional parts of a command.

Value of command options
The commands use a number of options and these sections use the following conventions to denote their values in syntax definitions:

SymmID
The local storage system.

WitnessName
A name for a vWitness definition.

- The name has up to 12 characters and starts with an alphabetic character.
- The remainder of the name can contain alphanumeric characters, underscores, and hyphens.
- The name is not case sensitive, but the system preserves the case.

IPorDNS
The IP address or the fully qualified DNS name of a vWitness instance. The address or name has a maximum of 128 characters.

Array access rights and user authorization
All the commands, except for list and show, require array access rights of SYMCFG and user authorization of Storage Admin.

Add a vWitness definition

To add a new vWitness definition to a storage array, use the syntax below. This also enables the definition automatically, but you can disable it using `symcfg disable` as described in Disable the use of a vWitness definition on page 31:

```
symcfg -sid SymmID add -witness WitnessName -location IPorDNS
```

Note

Create only one definition for each vWitness instance, specifying either the IP address or the fully-qualified DNS name of the instance.

Example
To add and enable a vWitness definition named metrovw1 that refers to a vWitness instance at IP address 198.51.100.24 on the storage array 1234:

```
symcfg -sid 1234 add -witness metrovw1 -location 198.51.100.24
```

Disable the use of a vWitness definition

To disable the use of a vWitness definition:

```
symcfg -sid SymmID disable -witness WitnessName [-force|-symforce]
```

Use the `-force` option when the definition is in use (protecting a Metro configuration), and there is another Witness (either an Array or a Virtual Witness) available to take over from this one.

Use the `-symforce` when the definition is in use and there is no other Witness available to take over from this one.

Example
To disable (suspend) the availability of the vWitness definition named metrovw1 on storage array 1234 when there is no other Witness available:

```
symcfg -sid 1234 disable -witness metrovw1 -symforce
```

Enable a vWitness definition

To enable a vWitness definition after it has been suspended:

```
symcfg -sid SymmID enable -witness WitnessName
```

Example
To enable the vWitness definition named metrovw1:

```
symcfg -sid 1234 enable -witness metrovw1
```

Modify a vWitness definition

To modify a vWitness definition:

1. Disable (Disable the use of a vWitness definition on page 31) and remove the existing definition (Remove a vWitness definition on page 32).
2. Add a new definition with the modified values (Add a vWitness definition on page 30).

Example
To change the IP address of a vWitness definition named metrovw1 on storage array 1234 to 198.51.100.32:

```
symcfg -sid 1234 disable -witness metrovw1 -force
symcfg -sid 1234 remove -witness metrovw1
symcfg -sid 1234 add -witness metrovw1 -loction 198.51.100.32
```

Remove a vWitness definition

First, disable the vWitness definition (Disable the use of a vWitness definition on page 31) and then remove it:

```
symcfg -sid SymmID remove -witness WitnessName
```

Example

To remove the vWitness definition named metrovw1 from storage array 1234:

```
symcfg -sid 1234 disable -witness metrovw1 -force
symcfg -sid 1234 remove -witness metrovw1
```

View vWitness definitions

View summary information on all vWitness definitions

```
symcfg -sid SymmID list -witness [-v] [-out xml] [-offline]
```

The `-v` option produces detailed information, similar to that produced by the `show` argument, but for all vWitness definitions.

Output is available in text or XML format. Use `-out xml` to generate XML.

Use the `-offline` option to display information from the data cached in the Solutions Enabler database file.

View detailed information on a single vWitness definition

```
symcfg -sid SymmID show -witness WitnessName [-out xml] [-offline]
```

Examples

Display information on all vWitness instances on the storage array 1234:

```
symcfg -sid 1234 list -witness
```

Display information on vWitness definition named metrovw1 on storage array 1234:

```
symcfg -sid 1234 show -witness metrovw1
```

Manage instances of vWitness

The following sections show how to create and remove instances of vWitness.

Create a vWitness

Follow the instructions in Install the vWitness instances on page 20 to add the new vWitness instance. Then add a definition of the instance to each storage array that may use that instance.

Remove a vWitness

To remove a vWitness, remove its definition from all storage arrays that use the instance. Then stop the storvwlsd daemon and prevent it from automatically starting.

Procedure

1. Make sure that the vWitness instance is not in use by any storage array.
2. Remove the definition of the instance from each storage array that has one (using Unisphere or SYMCLI command).
3. Launch and log in to vApp Manager on the system that runs the vWitness instance.
4. Click **Manage Daemons**.
5. In the **Action** column, click **Stop** next to the storvwlsd daemon.
6. In the **Autostart** column, click **Unset** next to the storvwlsd daemon.

Download vWLS log files

Each vWitness instance maintains a log file. Should problems arise, the log file can help locate the cause.

Procedure

1. Launch and log in to vApp Manager on the system running the vWitness instance.
2. Click **Appliance Data/Log**.
3. On the **Daemon/Log Files** panel, select **storvwlsd** from the **Select Daemon** list.
4. Click **Download storvwlsd Logs**.
5. In the file browser dialog, select a location for the downloaded file.
6. Click **Save**.