Copyright © 2018, Zerto Ltd. All rights reserved.
Information in this document is confidential and subject to change without notice and does not represent a commitment on the part of Zerto Ltd. Zerto Ltd. does not assume responsibility for any printing errors that may appear in this document. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or information storage and retrieval systems, for any purpose other than the purchaser’s personal use, without the prior written permission of Zerto Ltd.
All other marks and names mentioned herein may be trademarks of their respective companies.

The scripts are provided by example only and are not supported under any Zerto support program or service. All examples and scripts are provided “as-is” without warranty of any kind. The author and Zerto further disclaim all implied warranties including, without limitation, any implied warranties of merchantability or of fitness for a particular purpose.
In no event shall Zerto, its authors, or anyone else involved in the creation, production, or delivery of the scripts be liable for any damages whatsoever (including, without limitation, damages for loss of business profits, business interruption, loss of business information, or other pecuniary loss) arising out of the use of or inability to use the sample scripts or documentation, even if the author or Zerto has been advised of the possibility of such damages. The entire risk arising out of the use or performance of the sample scripts and documentation remains with you.

ZVR-API-5.5U4 Rev 01 Apr2018
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP Method</td>
<td>60</td>
</tr>
<tr>
<td>Security</td>
<td>60</td>
</tr>
<tr>
<td>See Also</td>
<td>60</td>
</tr>
<tr>
<td>Format</td>
<td>60</td>
</tr>
<tr>
<td>Code Examples</td>
<td>62</td>
</tr>
<tr>
<td>Peer Sites API</td>
<td>62</td>
</tr>
<tr>
<td>HTTP Method</td>
<td>62</td>
</tr>
<tr>
<td>Security</td>
<td>62</td>
</tr>
<tr>
<td>See Also</td>
<td>62</td>
</tr>
<tr>
<td>Format</td>
<td>63</td>
</tr>
<tr>
<td>Peersites - GET</td>
<td>63</td>
</tr>
<tr>
<td>URL</td>
<td>63</td>
</tr>
<tr>
<td>Code Examples</td>
<td>65</td>
</tr>
<tr>
<td>Peersites - POST</td>
<td>65</td>
</tr>
<tr>
<td>URL</td>
<td>65</td>
</tr>
<tr>
<td>Code Examples</td>
<td>66</td>
</tr>
<tr>
<td>Peersites - DELETE</td>
<td>66</td>
</tr>
<tr>
<td>URL</td>
<td>67</td>
</tr>
<tr>
<td>Code Examples</td>
<td>67</td>
</tr>
<tr>
<td>Service Profiles API</td>
<td>68</td>
</tr>
<tr>
<td>URL</td>
<td>68</td>
</tr>
<tr>
<td>HTTP Method</td>
<td>68</td>
</tr>
<tr>
<td>Security</td>
<td>68</td>
</tr>
<tr>
<td>See Also</td>
<td>68</td>
</tr>
<tr>
<td>Format</td>
<td>68</td>
</tr>
<tr>
<td>Code Examples</td>
<td>69</td>
</tr>
<tr>
<td>Session API</td>
<td>70</td>
</tr>
<tr>
<td>HTTP Methods</td>
<td>70</td>
</tr>
<tr>
<td>Security</td>
<td>70</td>
</tr>
<tr>
<td>See Also</td>
<td>70</td>
</tr>
<tr>
<td>Format</td>
<td>70</td>
</tr>
<tr>
<td>Session: POST</td>
<td>70</td>
</tr>
<tr>
<td>Session: DELETE</td>
<td>71</td>
</tr>
<tr>
<td>Code Examples</td>
<td>71</td>
</tr>
<tr>
<td>Tasks API</td>
<td>74</td>
</tr>
<tr>
<td>URL</td>
<td>74</td>
</tr>
<tr>
<td>HTTP Method</td>
<td>74</td>
</tr>
<tr>
<td>Security</td>
<td>76</td>
</tr>
<tr>
<td>See Also</td>
<td>76</td>
</tr>
<tr>
<td>Format</td>
<td>76</td>
</tr>
<tr>
<td>Code Examples</td>
<td>77</td>
</tr>
<tr>
<td>Virtualization Sites API</td>
<td>83</td>
</tr>
<tr>
<td>URL</td>
<td>83</td>
</tr>
<tr>
<td>HTTP Method</td>
<td>83</td>
</tr>
<tr>
<td>Security</td>
<td>84</td>
</tr>
<tr>
<td>See Also</td>
<td>84</td>
</tr>
<tr>
<td>Format</td>
<td>84</td>
</tr>
<tr>
<td>Code Examples</td>
<td>87</td>
</tr>
<tr>
<td>Protected VMs API</td>
<td>88</td>
</tr>
<tr>
<td>URL</td>
<td>88</td>
</tr>
<tr>
<td>HTTP Method</td>
<td>88</td>
</tr>
<tr>
<td>Security</td>
<td>90</td>
</tr>
<tr>
<td>See Also</td>
<td>90</td>
</tr>
<tr>
<td>Format</td>
<td>90</td>
</tr>
<tr>
<td>Code Examples</td>
<td>94</td>
</tr>
<tr>
<td>VPGs API</td>
<td>95</td>
</tr>
</tbody>
</table>
CHAPTER 5: XML REQUEST AND RESPONSE FORMATS

V1/ API XML Response Format ........................................... 233
Alerts API XML Response Format ........................................ 233
Events API XML Response Format ....................................... 234
Local Site API XML Response Format ................................. 237
Peer Sites API XML Request and Response Formats ............... 238
   Peer Sites API GET Method Request and Response Formats .... 238
   Peer Sites API POST Method Request and Response Formats ... 238
CHAPTER 1: INTRODUCTION TO THE ZERTO VIRTUAL REPLICATION RESTFUL APIs

The Zerto Virtual Replication RESTful API enables you to manage Zerto Virtual Replication programmatically. The REST APIs provide a way to automate many of the tasks required to manage DR, without having to use the Zerto User Interface.

The following topics are described in this:

- “Using the APIs”, below
- “A First Time Tutorial”, on page 10
- “Managing VPGs”, on page 12

APIs are available both to return status information and to perform actions, such as a failover. The REST APIs are used for the following:

- Start and end a session. See below, “Using the APIs”, on page 7 and “Session API”, on page 70.
- Return a list of available Zerto Virtual Replication v1 REST APIs. See “Listing the Available APIs”, on page 8 and “/v1/ API”, on page 33.
- Return information about Zerto Virtual Replication alerts. See “Alerts API”, on page 35.
- Dismiss or undismiss an alert. See “Alerts API”, on page 35.
- Return information about Zerto Virtual Replication events. See “Events API”, on page 50.
- Return information about the local site where the API is run. See “Local Site API”, on page 60.
- Return information about the peer sites for the site where the API is run. See “Peer Sites API”, on page 62.
- Perform pairing the current site to a peer site. See “Peer Sites API”, on page 62
- Perform unpairing the current site to a peer site. See “Peer Sites API”, on page 62
- Return information about service profiles. See “Service Profiles API”, on page 68.
- Return information about tasks run at a site. See “Tasks API”, on page 74.
- Return information about the hypervisor site where the API is run and the paired hypervisor sites and as well as information about the resources at a specified site. See “Virtualization Sites API”, on page 83.
- Return information about protected virtual machines. See “Protected VMs API”, on page 88.
- Return information about VPGs. See “VPGs API”, on page 95.
- Manage VPGs. See “VPG Management API”, on page 117.
- Return information about VRAs. See “VRAs API”, on page 176.
- Generate a resource report about the virtual machines being protected to a recovery site. See “Resources Report API”, on page 187.

The information returned by these APIs refers to what is managed by the Zerto Virtual Manager where the API is run.

Using the APIs

All APIs are exposed over HTTPS.

IMPORTANT API Update:

The parameter used for defining the response type format was changed from Content-Type to Accept.

In order to control the content-type of the response, the user can specify either “application/xml” or “application/json” in the Accept header. If the Accept header is not specified, the response content type will be the default - “application/json”. Before this update, the ZVM used the content-type value of the API request to determine the response content-type.

Please take this into consideration for your current and future automation scripts.
Most of the Zerto Virtual Replication RESTful APIs require a session running with basic authorization. The username and password authorization used must be a valid username and password either for the Windows machine where the Zerto Virtual Manager is installed or for the hypervisor manager, VMware vCenter Server or Microsoft SCVMM, accessed by the Zerto Virtual Manager. In both cases the Zerto Virtual Manager is the Zerto Virtual Manager where the APIs will run.

The following APIs can be called without any authentication:

- Listing the RESTful APIs, described in “Listing the Available APIs”, on page 8 and “/v1/ API”, on page 33.
- Getting help for an API, described in “ Getting Help for an API”, on page 9.
- Generating a resource report about the virtual machines being protected to a recovery site, using the ResourcesReport API, described in “Resources Report API”, on page 187.

When passing a URL in a browser, you require a security certificate. In Microsoft Internet Explorer you have to be in Compatibility mode.

To test the APIs, Zerto recommends using cURL or a REST client, such as the following:


Starting a Session

Using the username and password either for the Windows machine where the Zerto Virtual Manager is installed or for the hypervisor manager, VMware vCenter Server or Microsoft SCVMM, accessed by the Zerto Virtual Manager, you can establish a session by posting the following URL:

https://zvm_ip:port/v1/session/add

A session identifier, x-zerto-session, is returned as part of the response header and the session is established. The session identifier is used in the client code with every API call for the duration of the session.

The APIs can be consumed by applications implemented in different technologies in a stateless manner.

Data returned is formatted either as JSON or as XML as set by the consumer. By default, data that is returned for the v1 APIs is formatted as JSON.

Listing the Available APIs

Running the following API returns a list of available APIs under /v1:

https://zvm_ip:port/v1/

The following output is generated:

```json
[{
    "href": "https://10.100.0.51:9669/v1/events",
    "rel": "down",
    "type": "IEventsService"
}, {
    "href": "https://10.100.0.51:9669/v1/vpgs",
    "rel": "down",
    "type": "IVpgService"
}, {
    "href": "https://10.100.0.51:9669/v1/vms",
    "rel": "down",
    "type": "IVmService"
}, {
    "href": "https://10.100.0.51:9669/v1/vras",
    "rel": "down",
    "type": "IVraService"
}, {
    "href": "https://10.100.0.51:9669/v1/peersites",
    "rel": "down",
    "type": "IPeerSitesService"
}, {
    "href": "https://10.100.0.51:9669/v1/session",
    "rel": "down",
    "type": "ISessionService"
}, {
    "href": "https://10.100.0.51:9669/v1/tasks",
    "rel": "down",
    "type": "ITasksService"
}, {
    "href": "https://10.100.0.51:9669/v1/serviceprofiles",
    "rel": "down",
    "type": "IServiceProfilesService"
}, {
    "href": "https://10.100.0.51:9669/v1/virtualizationsites",
    "rel": "down",
    "type": "IVirtualizationSitesService"
}, {
    "href": "https://10.100.0.51:9669/v1/zorgs",
    "rel": "down",
    "type": "IZorgsService"
}, {
    "href": "https://10.100.0.51:9669/v1/localsite",
    "rel": "down",
    "type": "ILocalSiteService"
}, {
    "href": "https://10.100.0.51:9669/v1/alerts",
    "rel": "down",
    "type": "IAlertsService"
}]
```

Where:

- **zvm_ip** The IP address of the Zerto Virtual Manager where the API is run.
- **port** The port to access the Zerto Virtual Manager. The default port is 9669.
- **href** The URL used.
Filtering Information Retrieved By an API

Information retrieved by many of the APIs can be filtered. The filter parameters are optional and any combination of these parameters is valid. When more than one filter is applied, the AND operand is used. The Help page lists the filters for a specific API. For details, refer to "Getting Help for an API", below.

Getting Help for an API

Help about each v1 API is available, using the following URL:

https://zvm_ip:port/v1/API/help

Where:

- **zvm_ip**  The IP address of the Zerto Virtual Manager where the report is run.
- **port**  The port to access the Zerto Virtual Manager. The default port is 9669.
- **API**  The name of the API for which help is required, for example vpgs or vras.

The available methods are returned. For example, the vpgs API has the following response:
Drilling-down on a method returns an example response in both XML and JSON formats and the complete XML schema. For example, the `/v1/` GET method returns the following response:

### Ending a Session

End a session with the following URL and the HTTP DELETE request:

https://zvm_ip:port/v1/session

**Note:** If a session is dormant for thirty minutes, the session is automatically terminated.

### A First Time Tutorial

This walk-through uses cURL installed with SSL. The username and password supplied for authentication to start a session is the hypervisor manager, vCenter Server or Microsoft SCVMM, credentials accessed by the Zerto Virtual Manager.
To start a session and run some queries, do the following:

1. Enter the following code to start a session using cURL, where the response is headers.

   ```
   ``

   where `zvm_ip` is the IP of the Zerto Virtual Manager where the API will run and `user:password` are the username and password that access the vCenter Server or the Microsoft SCVMM hypervisor management tool.

   The `1` in `-d "{"AuthenticationMethod":1}"`, indicates that vCenter Server or the Microsoft SCVMM hypervisor management tool credentials are being used.

   The `-k` option at the end of the command indicates that certificate verification has been turned off.

   The `responseHeader` file that is returned is written to the folder in which the cURL executable is located. It is similar to the following:

   ```
HTTP/1.1 200 OK
Content-Length: 0
Server: Microsoft-HTTPAPI/2.0
x-zerto-session: 9UDQD6RG7YF33QJLWQXGJVB5G7C453N277NA22P7FSNWZCJTWCBRHQ
Date: Mon, 29 Jun 2015 08:50:58 GMT
   ```

   The fourth line contains the value of `x-zerto-session`, which is used in all subsequent calls.

2. To retrieve information about all VPGs, run:

   ```
   ``

   where `zvm_ip` is the IP of the Zerto Virtual Manager where the API will run.

   The response to this command is a list of VPGs with information about each VPG, in JSON format.

   The following example shows the result when one VPG, named VPGA, is returned.
Managing VPGs

The Zerto Virtual Replication RESTful APIs includes an API that enables you to manage the creation and editing of VPGs programmatically.

The following topics are described in this section:

- “An Introduction to the vpgSettings API”, below
- “How Does the vpgSettings API Work”, on page 13
- “Using the APIs”, on page 7
- “Short Outlines of How to Manage a VPG”, on page 14
- “A First Time Tutorial Using the vpgSettings API”, on page 17
- “Getting Help for an API”, on page 9

An Introduction to the vpgSettings API

You can manage VPGs using both the /v1/vpgs API or the /v1/vpgSettings API. Use the /v1/vpgs API to perform actions on a VPG, such as failing over a VPG, cloning a VPG or testing a VPG. Use the /v1/vpgSettings API to manage the definition of a VPG, including editing the VPG definition and adding or removing virtual machines from a VPG.

Although both APIs can be used to create a VPG, the /v1/vpgs API can only be used to create a VPG in a VMware vSphere environment and relies on Zerto Virtual Manager defaults for many of the VPG definitions. Using the /v1/vpgSettings API, you have full control to specify every field in the VPG definition in both a VMware vSphere and Microsoft Hyper-V environment.

Use the vpgSettings API to do the following:

- Create a new VPG.
- Display values in an existing VPG.
- Edit existing values in a VPG.

What is Supported

The vpgSettings API works in the following environments:

- vCenter Server
- Hyper-V
- vCenter Server to Hyper-V and Hyper-V to vCenter Server
- Running on the protected site

For additional cURL examples, see “cURL Code”, on page 22.
After 30 minutes of inactivity, the session automatically ends.
Managing VPGs

Running on the recovery site

What is Not Supported

The vpgSettings API does not support the following:

- Preseeding
- Virtual machines with RDM disks
- Setting backup
- AWS
- Azure

How Does the vpgSettings API Work

The definition of a VPG can be described by the set of values. The vpgSettings API enables managing this set of values. All the VPG settings are managed in a VPG settings object, which is saved in memory. This object is managed during a session and used to either create a VPG or update an existing VPG.

Managing a VPG using the vpgSettings API involves the following steps:

1. Create a VPG settings object.
2. Manipulate the VPG settings object to include the values you want.
3. Commit the VPG settings object to update the VPG definition in the Zerto Virtual Manager using the values from the VPG settings object in the memory.

The VPG settings object uses an identifier, the vpgSettingsIdentifier. This identifier is not the same as the VPG identifier used to identify an existing VPG.

To ensure uniqueness, you supply identifiers to the vpgSettings API and not names. To retrieve identifiers to use in the vpgSettings API, use other Zerto Virtual Replication RESTful APIs. For example, to retrieve the recovery network identifiers for failover and failover test networks, use the virtualizationsites/{SITEIDENTIFIER}/networks API to return both the network names and identifiers. In the vpgSettings API, you use the network identifier and not the name. In the same way, use the virtualizationsites/{SITEIDENTIFIER}/vms API to return the identifiers of all unprotected virtual machines in a site, and use these identifiers in the vpgSettings API to specify the virtual machines to add to a VPG or to protect in a new VPG.

Note: For details about using all the Zerto Virtual Replication APIs, refer to the Zerto Virtual Replication RESTful API Reference Guide documentation.

Using the vpgSettings API

The vpgSettings API is exposed over HTTPS and requires a session running with basic authorization. The username and password authorization used must be a valid username and password either for the Windows machine where the Zerto Virtual Manager is installed or for the hypervisor manager, VMware vCenter Server or Microsoft SCVMM, accessed by the Zerto Virtual Manager. In both cases the Zerto Virtual Manager is the Zerto Virtual Manager where the APIs will run.

Using the username and password either for the Windows machine where the Zerto Virtual Manager is installed or for the hypervisor manager, VMware vCenter Server or Microsoft SCVMM, accessed by the Zerto Virtual Manager, you can establish a session by posting the following URL:

https://zvm_ip:port/v1/session/add

A session identifier, x-zerto-session, is returned as part of the response header and the session is established. The session identifier is used in the client code with every API call for the duration of the session.

The APIs can be consumed by applications implemented in different technologies in a stateless manner.

Data returned is formatted either as JSON or as XML as set by the consumer. By default, data that is returned for the v1 APIs is formatted as JSON.
An API session times out after 30 minutes of inactivity. A VPG settings object is automatically deleted from memory in the following situations:

- When a session times out.
- When the VPG settings object is not used in a session for 30 minutes, even if the session is still active.
- When the VPG settings object is committed.

**Short Outlines of How to Manage a VPG**

You manage a VPG using the vpgSettings API by using multiple methods. This section outlines the basic use of the vpgSettings API to perform different VPG functions:

- Creating a VPG
- Updating a VPG
- Adding a Virtual Machine to a VPG
- Resetting Values in a VPG Settings Object
- Deleting a Virtual Machine from a VPG
- Deleting the VPG Settings Object

**Creating a VPG**

When creating a VPG the set of values are specified and saved as a VPG settings object in the API session. To update the Zerto Virtual Manager with the settings, you commit the object.

You create the VPG settings object by executing the vpgSettings API with the POST method and the necessary request body.

```plaintext
https://zvm_ip:port/v1/vpgsettings
```

The request body specifies all the details required for the VPG, details of which are in “VPGs: POST”, on page 109.

You can create a skeleton VPG settings object by using the following request body:

```json
{}
```

You can then edit the response body with the required values and pass the completed JSON as a new request body.

Executing this API creates the object settings and returns the vpgSettingsIdentifier in the response body. To create the VPG in the Zerto Virtual Manager, you have to commit the changes using the following API, with the POST method:

```plaintext
https://zvm_ip:port/v1/vpgsettings/vpgSettingsIdentifier/commit
```

For more details, refer to “To create a VPG;”, on page 17.

**Updating a VPG**

When editing a VPG, you first get the VPG settings as an object using the vpgIdentifier to create a new vpgSettingsIdentifier. The settings for the VPG are then modified as required and then the VPG settings object with the changes is committed to update the Zerto Virtual Manager with the changed settings. For example, the following URL with a GET method retrieves the VPG identifier that can then be used to generate a vpgSettingsIdentifier for the settings object for the VPG:

```plaintext
https://zvm_ip:port/v1/vpgs
```

Details of the VPG are returned including the VPG identifier, which is then used in the request body of the vpgSettings URL, using a POST method:

```plaintext
https://zvm_ip:port/v1/vpgsettings
```

The request body is similar to the following, in JSON format:

```json
{
  "VpgIdentifier": "49b189ff-dcd7-4544-a25d-356bea6c6676"
}
```
Managing VPGs

The vpgSettingsIdentifier is returned in the response body. You can use this identifier to retrieve the settings object for the VPG, using a GET method:

https://zvm_ip:port/v1/vpgsettings/ac942f3f-8e40-4d5f-a782-8cfebe80ed33

You update the VPG settings object by executing the relevant API. For example to update any of the basic settings, you use the vpgsettings/vpgSettingsIdentifier/basic API with the PUT method and the necessary request body:

https://zvm_ip:port/v1/vpgsettings/ac942f3f-8e40-4d5f-a782-8cfebe80ed33/basic

In the following example, the request body specifies that the priority is changed to high and the journal history is changed to 10 hours:

```json
{
  "JournalHistoryInHours": 10,
  "Priority": "High"
}
```

Executing this API changes the object settings. To update the Zerto Virtual Manager with the changed settings, you have to commit the changes using the following API, with the POST method:

https://zvm_ip:port/v1/vpgsettings/ac942f3f-8e40-4d5f-a782-8cfebe80ed33/commit

The task identifier for the update is returned in the response body and the settings object is deleted.

**Note:** Changes to all the VPG settings, such as the expected RPO, are performed in the same way as described here.

For more details, refer to “To update an existing VPG:”, on page 20.

**Adding a Virtual Machine to a VPG**

The procedure to add a virtual machine to an existing VPG is the same as updating a VPG, described above, but a POST method is used instead of a PUT method to update the settings object. After the settings object is updated with the new information, the object must be committed, also using the POST method. For more details, refer to “To add a virtual machine to the VPG:”, on page 21.

**Resetting Values in a VPG Settings Object**

To reset a value in a VPG settings object, get the object and save the response. You can then delete the section of the VPG settings object, using the DELETE method. For example to reset the priority to the default value:

First retrieve the basic settings in the VPG settings object using the following URL with the GET method:

https://zvm_ip:port/v1/vpgsettings/vpgSettingsIdentifier/basic

Save the response body and then delete the settings in the current VPG settings object, using the same URL but with the DELETE object:

https://zvm_ip:port/v1/vpgsettings/vpgSettingsIdentifier/basic

You can now update the basic settings with the reset priority by changing the value of priority setting in the new request body, as follows:

```json
{
  "JournalHistoryInHours": 4,
  "Name": "test-using-API1",
  "Priority": null,
  "ProtectedSiteIdentifier": "6c36720e-b32d-44de-9600-042ce5268d0d",
  "RecoverySiteIdentifier": "02159615-16d8-40e0-87f8-2fe669bf414f",
  "RpoInSeconds": 300,
  "ServiceProfileIdentifier": null,
  "TestIntervalInMinutes": 262080,
  "UseWanCompression": true,
  "ZorgIdentifier": null
}
```
Use the following URL with the PUT method to update the VPG settings object:

https://zvm_ip:port/v1/vpgsettings/vpgSettingsIdentifier/basic

Use the following URL with the POST method to commit the settings to update the VPG:

https://zvm_ip:port/v1/vpgsettings/vpgSettingsIdentifier/commit

Example: Resetting the Journal Values

The same procedure is used for all the settings in a VPG. For example to change the journal settings, first retrieve the journal settings object using the following URL with the GET method:

https://zvm_ip:port/v1/vpgsettings/vpgSettingsIdentifier/journal

Save the response body and then delete the settings, using the same URL but with the DELETE object:

https://zvm_ip:port/v1/vpgsettings/vpgSettingsIdentifier/journal

You can now update the journal settings by changing the relevant values in the new request body for example, the storage to use and the hard limit and warning threshold are unlimited, as follows:

```json
{
   "DatastoreIdentifier":"841c3a57-e4bb-4e53-b045-47e95da4ece9.datastore-201",
   "Limitation":{
      "HardLimitInMB":0,
      "HardLimitInPercent":0,
      "WarningThresholdInMB":0,
      "WarningThresholdInPercent": 0
   }
}
```

Use the following URL with the PUT method to update the settings:

https://zvm_ip:port/v1/vpgsettings/vpgSettingsIdentifier/journal

Use the following URL with the POST method to commit the settings:

https://zvm_ip:port/v1/vpgsettings/vpgSettingsIdentifier/commit

Deleting a Virtual Machine from a VPG

The procedure to delete a virtual machine from an existing VPG requires that you first get the VPG settings as an object using the vpgIdentifier to create a new vpgSettingsIdentifier. The virtual machine identifier to remove from the VPG is then sent with the DELETE method to delete the virtual machine from the VPG settings:

https://zvm_ip:port/v1/vpgsettings/ac942f3f-8e40-4d5f-a782-8cfebe80ed33/vms/13d550a4-24af-4914-9ca4-09f8619eb703.vm-153

Executing this API changes the object settings. To update the Zerto Virtual Manager with the changed settings, you have to commit the changes using the following API, with the POST method:

https://zvm_ip:port/v1/vpgsettings/ac942f3f-8e40-4d5f-a782-8cfebe80ed33/commit

The task identifier for the update is returned in the response body and the settings object is deleted.

**Note:** The virtual machine is deleted from the VPG but the target disks in the recovery site are kept.

Deleting the VPG Settings Object

A VPG settings object is destroyed in the following cases:

- When a session times out.
- When the object is not used in a session for 30 minutes, even if the session is still active.
- When the object is committed.
By executing the vpgSettings API for the object, with a DELETE method:

```
https://zvm_ip:port/v1/vpgsettings/ac942f3f-8e40-4d5f-a782-8cfebe80ed33
```

**Note:** To delete a VPG you use the vpgs API. For details, refer to the *Zerto Virtual Replication RESTful API Reference Guide* documentation.

### A First Time Tutorial Using the vpgSettings API

This walk-through uses the URLs that are passed to create a session and then edit an existing VPG. The examples were tested using a REST client, such as Postman, used in Google Chrome, and available from [http://www.getpostman.com/](http://www.getpostman.com/).

In all the example code, the `zvm_ip` is the IP of the Zerto Virtual Manager where the API will run and both the content type and accept type is application/json.

**Note:** For a brief tutorial using cURL, refer to the *Zerto Virtual Replication RESTful API Reference Guide* documentation.

#### To start a session:

- Use the following URL with a POST method and basic authentication to access the hypervisor management tool, vCenter Server or the Microsoft SCVMM.

```
https://zvm_ip:9669/v1/session/add
```

The Request Body contains the following to set the authentication method to use the hypervisor management tool authentication:

```
{
    "AuthenticationMethod":1
}
```

The Response Header that is returned is similar to the following:

```
Content-Length: 0
Date: Mon, 29 Jun 2015 08:50:58 GMT
Server: Microsoft-HTTPAPI/2.0
x-zerto-session: 9UDQD6RG7YF33QJLWQXGJV8C453N277NA22P7FSNWVZCJTWCBRHQ
```

The fourth line contains the value of `x-zerto-session`, which is used in all subsequent calls.

#### To create a VPG:

1. Create a skeleton VPG settings object, using the following URL with a POST method:

```
https://zvm_ip:9669/v1/vpgsettings
```

and the following request body:

```
{}
```

The API must be run on the protected site.

The vpgSettingsIdentifier is returned.

2. Get the skeleton structure for the VPG settings object using the following URL with a GET method:

```
https://zvm_ip:9669/v1/vpgsettings/3864850e-1580-4518-95d1-f88a4d97c677
```

Where `3864850e-1580-4518-95d1-f88a4d97c677` is the vpgSettingsIdentifier returned in step 1.

3. Set in the basic parameters as required, at least the following fields:

- JournalHistoryInHours
- Name
- Priority
- ProtectedSiteIdentifier
- RecoverySiteIdentifier
- RpoInSeconds
- TestIntervalInMinutes.
Managing VPGs

Get the RecoverySiteIdentifier by running the virtualizationsites API, using the following URL with the GET method:

https://zvm_ip:9669/v1/virtualizationsites

**Note:** If required, the ServiceProfileIdentifier can be retrieved using the /v1/serviceprofiles API with the GET method and the ZorgIdentifier can be retrieved using the /v1/zorgs API with the GET method.

4. Get the structures for the Journal, Networks and Recovery sections in the skeleton by copying the Json request bodies from the examples in the help, accessed by running the following URL:

https://zvm_ip:9669/v1/vpgsettings/help

For more details, refer to “Getting Help for an API”, on page 9.

5. Use the virtualizationsites API to get the following values from the recovery site to add to the skeleton settings:

   - DatastoreIdentifier for both the journal and recovery storage
   - DefaultNetworkIdentifier, for both the failover and move network and for the test failover network
   - DefaultFolderIdentifier
   - DefaultHostClusterIdentifier or DefaultHostIdentifier or ResourcePoolIdentifier

6. Set the journal limitations as required. A zero, 0, value means unlimited.

7. Get the identifiers for the virtual machines that are not protected, with the virtualizationsites/vms API and add the list of virtual machines you want protected in the VPG, as in the following example

   ```
   [{"VmIdentifier": "13d550a4-24af-4914-9ca4-09f8619eb703.vm-147"}]
   ```

8. Remove the vpgIdentifier and vpgSettingsIdentifier lines from the skeleton.

9. Create a new VPG settings object by using the following URL with the POST method:

   https://zvm_ip:9669/v1/vpgsettings

and the updated skeleton for the request body.

10. Create the VPG by committing the updated skeleton, using the following URL with a POST method:

    https://zvm_ip:9669/v1/vpgsettings/a965180f-1375-6545-9d21-556a4a41c871/commit

The task identifier is returned which can then be used with the /v1/tasks/{taskIdentifier} API to monitor the task progress. The settings object for this VPG is also destroyed.

**Note:** For details about using virtualizationsites APIs, refer to “Virtualization Sites API”, on page 83.
The following is an example VPG settings object skeleton after it has been edited:

```json
{
  "Backup": null,
  "Basic": {
    "JournalHistoryInHours": 4,
    "Name": "MyFirstVpg",
    "Priority": "Medium",
    "ProtectedSiteIdentifier": "6c36720e-b32d-44de-9600-042ce5268d0d",
    "RecoverySiteIdentifier": "02159615-16d8-40e0-87f8-2fe669bf414f",
    "RpoInSeconds": 300,
    "ServiceProfileIdentifier": null,
    "TestIntervalInMinutes": 262080,
    "UseWanCompression": true,
    "ZorgIdentifier": null
  },
  "BootGroups": {
    "BootGroups": [
      {
        "BootDelayInSeconds": 0,
        "BootGroupIdentifier": "00000000-0000-0000-0000-000000000000",
        "Name": "Default"
      }
    ]
  },
  "Journal": {
    "DatastoreIdentifier": "841c3a57-e4bb-4e53-b045-47e95da4ece9.datastore-201",
    "Limitation": {
      "HardLimitInMB": 0,
      "HardLimitInPercent": 0,
      "WarningThresholdInMB": 0,
      "WarningThresholdInPercent": 0
    }
  },
  "Networks": {
    "Failover": {
      "Hypervisor": {
        "DefaultNetworkIdentifier": "841c3a57-e4bb-4e53-b045-47e95da4ece9.network-34"
      }
    },
    "FailoverTest": {
      "Hypervisor": {
        "DefaultNetworkIdentifier": "841c3a57-e4bb-4e53-b045-47e95da4ece9.network-34"
      }
    }
  },
  "Recovery": {
    "DefaultDatastoreIdentifier": "841c3a57-e4bb-4e53-b045-47e95da4ece9.datastore-201",
    "DefaultFolderIdentifier": "841c3a57-e4bb-4e53-b045-47e95da4ece9.group-v22",
    "DefaultHostClusterIdentifier": null,
    "DefaultHostIdentifier": "841c3a57-e4bb-4e53-b045-47e95da4ece9.host-30",
    "ResourcePoolIdentifier": null
  }
}
```
To update an existing VPG:

1. Use the following URL with a GET method and the x-zerto-session added to the header to retrieve the VPG identifiers for existing VPGs. A VPG identifier is then used to create the VPG settings object:

   ```
   https://zvm_ip:9669/v1/vpgs
   ``

   The response to this command is a list of VPGs with information about each VPG.

2. Copy the relevant VpgIdentifier from the response to use in the request body of another API, and create a VPG settings object for this VPG. The request body contains code similar to the following:

   ```
   {  
     "VpgIdentifier": "b030cbc3-3cd1-4a3b-9378-afd2a6e0ee88"
   }
   ```

   **Note:** The content type for this example is application/json.

   Use the following URL with a POST method to create the VPG settings object for the VPG:

   ```
   https://zvm_ip:9669/v1/vpgsettings
   ``

   The response to this command is the vpgSettingsIdentifier and the VPG settings object is created.

3. Get the VPG settings object for review, using the following URL with a GET method:

   ```
   https://zvm_ip:9669/v1/vpgsettings/3864850e-1580-4518-95d1-f88a4d97c677
   ``

   Where 3864850e-1580-4518-95d1-f88a4d97c677 is the vpgSettingsIdentifier returned in step 2.

4. Updating the VPG is done using the following steps:

   a) Get the current settings for the part of the VPG you want to update. For example, to update the basic values, use the following URL with a GET method:

      ```
      https://zvm_ip:9669/v1/vpgsettings/3864850e-1580-4518-95d1-f88a4d97c677/basic
      ``

      Where 3864850e-1580-4518-95d1-f88a4d97c677 is the vpgSettingsIdentifier returned in step 2.

      The following example response body is returned:

      ```
      {
        "JournalHistoryInHours": 4,
        "Name": "Test-Using-API",
        "Priority": "Medium",
        "ProtectedSiteIdentifier": "6c36720e-b32d-44de-9600-042ce5268d0d",
        "RecoverySiteIdentifier": "02159615-16d8-40e0-87f8-2fe669bf414f",
        "RpoInSeconds": 540,
        "ServiceProfileIdentifier": null,
        "TestIntervalInMinutes": 131040,
        "UseWanCompression": true,
        "ZorgIdentifier": null
      }
      ```
b) Use the response body from step 4 to create the request body for the update with the required changes and then use this request body with the following URL with a PUT method:

```
https://zvm_ip:9669/v1/vpgsettings/3864850e-1580-4518-95d1-f88a4d97c677/basic
```

The following example request body changes the priority, journal history and compression settings for the VPG:

```json
{
  "JournalHistoryInHours": 10,
  "Priority": "High",
  "UseWanCompression": false,
}
```

c) You can verify that the VPG settings object has been updated by rerunning the following URL with a GET method:

```
https://zvm_ip:9669/v1/vpgsettings/3864850e-1580-4518-95d1-f88a4d97c677/basic
```

d) Update the VPG with these new settings by committing the change, using the following URL with a POST method:

```
https://zvm_ip:9669/v1/vpgsettings/3864850e-1580-4518-95d1-f88a4d97c677/commit
```

The task identifier is returned which can then be used with the /v1/tasks/{taskIdentifier} API to monitor the task progress. The settings object for this VPG is also destroyed.

For details of the /v1/tasks API, refer to the Zerto Virtual Replication RESTful API Reference Guide documentation.

To add a virtual machine to the VPG:

1. Get the identifier for the protected site, for the virtual machines that are not protected, with the /v1/virtualizationsites or /v1/localsite API. For example, using the following URL with a GET method:

```
https://zvm_ip:9669/v1/localsite
```

The API must be run on the protected site.

2. Get the identifiers for the virtual machines that are not protected, with the /v1/virtualizationsites/vms API, using the following URL with a GET method:

```
https://zvm_ip:9669/v1/virtualizationsites/6c36720e-b32d-44de-9600-042ce5268d0d/vms
```

Where 6c36720e-b32d-44de-9600-042ce5268d0d is the local site identifier, returned in step 1. The following example response body is returned:

```json
[
  {
    "VmIdentifier": "13d550a4-24af-4914-9ca4-09f8619eb703.vm-147",
    "VmName": "Operations"
  },
  {
    "VmIdentifier": "13d550a4-24af-4914-9ca4-09f8619eb703.vm-148",
    "VmName": "HR"
  }
]
```

3. Updating the VPG to include the virtual machine you want to add, by first creating the VPG settings object as described in steps 1 to 3 in "To update an existing VPG"; above and then using the following URL with a POST method:

```
https://zvm_ip:9669/v1/vpgsettings/3864850e-1580-4518-95d1-f88a4d97c677/vms
```

Where 3864850e-1580-4518-95d1-f88a4d97c677 is the vpgSettingsIdentifier. The Request Body contains the following:

```json
{
  "VmIdentifier": "13d550a4-24af-4914-9ca4-09f8619eb703.vm-147"
}
```

Where 13d550a4-24af-4914-9ca4-09f8619eb703.vm-147 is the virtual machine identifier to add to the VPG.

4. Update the VPG with the virtual machine by committing the change, using the following URL with a POST method:

```
https://zvm_ip:9669/v1/vpgsettings/3864850e-1580-4518-95d1-f88a4d97c677/commit
```

The task identifier is returned which can then be used with the /v1/tasks/{taskIdentifier} API to monitor the task progress. The settings object for this VPG is also destroyed.
CHAPTER 2: CODE SAMPLES

The APIs can be consumed by applications implemented in different technologies in a stateless manner. Data returned is formatted either as JSON or as XML as set by the consumer. By default, data that is returned for the v1 APIs is formatted as JSON.

Code must include the necessary authentication to run, including the username and password fields that are used by the Zerto Virtual Manager to access the vCenter Server and establishing a session.

The following code samples are described in this:

- "cURL Code", below
- "Python Code", on page 23
- "stringPowerShell Scripts", on page 26

Starting an API session for any of the following code samples requires authentication. The following basic authentication methods are available:

**Windows-based authentication (default)** - The username and password are authenticated by Windows on the host on which the Zerto Virtual Manager is installed.

**Hypervisor manager authentication** - The username and password are authenticated by the VMware vCenter Server or Microsoft SCVMM to which the Zerto Virtual Manager is connected.

### cURL Code

Zerto recommends using cURL for scripting and testing before using another language, such as Python, for production. To use cURL with the Zerto Virtual Replication APIs, you must install cURL with SSL, for example from http://curl.haxx.se/download.html.

**Note:**

- You can turn off verification of the certificate by using the -k option in each cURL command.
- Any content type included as part of the request is JSON format, -H "Content-Type: application/json".
- The response returned is in JSON format, -H "Accept: application/json".

The following code shows how to start a session using cURL, where the response is headers.

```bash
```

In this example, the VMware vCenter Server or Microsoft SCVMM hypervisor management tool credentials are used for authentication, -d "\"AuthenticationMethod\":1\"". To use Windows authentication, use -d "\"AuthenticationMethod\":0\"".

The `responseHeader` file that is returned is similar to the following:

```
HTTP/1.1 200 OK
Content-Length: 0
Server: Microsoft-HTTPAPI/2.0
x-zerto-session: 9UDQD6RG7YF33QJLWQXGJ8C453N277NA22P7FSNWVZCJTWCBRHQ
Date: Mon, 29 Jun 2015 07:57:49 GMT
```

The `x-zerto-session` value is used in all subsequent calls, such as in the following example the retrieve information about all the VPGs.

```bash
curl -D responseHeader -H "Content-Type: application/json" -H "Accept: application/json" -H "x-zerto-session: 9UDQD6RG7YF33QJLWQXGJ8C453N277NA22P7FSNWVZCJTWCBRHQ"
https://zvm_ip:9669/v1/vpgs
```

Zerto Virtual Replication RESTful API Reference Guide - Version 5.5
Code Samples
The response to this command is the list of VPGs with information about each VPG, in JSON format.

You can use filters with the VPG API, as in the following example, where only information about VPGs with a low priority (priority=0) are returned.

```
```

To perform an action on a VPG, such as a failover test using a specific checkpoint, use a cURL command similar to the following.

```
```

The response is the task identifier or an error message if the action fails.

## Python Code

The Zerto Orchestrator uses Python code to enable automatically scheduled failover tests. The following code is available in full by downloading and deploying the Zerto Orchestrator in the /Zerto directory.

The following code uses the use the Requests library for making the REST calls: [http://docs.python-requests.org/en/latest/](http://docs.python-requests.org/en/latest/).

```python
import logging
import os.path
import sys
import subprocess
import os
import datetime
import time
import requests
import json
import base64

def GenerateZertoRestSession_WindowsAuthentication(zvmIp, user, password):
    credStr = user + "":" + password
    encodedCredStr = "Basic " + base64.b64encode(credStr)
    print encodedCredStr
    headers = {"Authorization": encodedCredStr}
    url = "https://" + zvmIp + "":9669/v1/session/add"
    r = requests.post(url, headers=headers, verify=False)
    print r.status_code
    if(r.status_code != requests.codes.ok):
        raise Exception("Failed authenticating to the Zerto Virtual Manager")
    res = r.headers.get('x-zerto-session')
    print "session = " + res
    return res
```
```python
def GenerateZertoRestSession_VcenterAuthentication(zvmIp, user, password):
    credStr = user + "\":" + password
    encodedCredStr = "Basic " + base64.b64encode(credStr)
    print encodedCredStr

    payload = {"AuthenticationMethod": 1}
dataval = json.dumps(payload)
print (dataval)

    headers = {'Authorization': encodedCredStr, 'content-type': 'application/json'}
url = "https://" + zvmIp + ":9669/v1/session/add"
r = requests.post(url, data=dataval, headers=headers, verify=False)
print r.status_code
if(r.status_code != requests.codes.ok):
    raise Exception("Failed authenticating to the Zerto Virtual Manager")
res = r.headers.get('x-zerto-session')
print "session = " + res
return res

def GetVpgIdFromName(zvmIp, sessionId, vpgName):
    url = "https://" + zvmIp + ":9669/v1/vpgs?name=" + vpgName
    headers = {'x-zerto-session': sessionId, 'content-type': 'application/json'}
r = requests.get(url, headers=headers, verify=False)
if(r.status_code != requests.codes.ok):
    raise Exception("Call to ZVM failed")
print r.status_code
print r.text
vpgData = json.loads(r.text)
if(len(vpgData) == 0):
    raise Exception("VPG by name " + vpgName + " not found")
if(len(vpgData) > 1):
    raise Exception("multiple vpgs named " + vpgName + " found")
thisVpg = vpgData[0]
res = thisVpg["VpgIdentifier"]
print thisVpg
print res
return res

def GetLatestCheckpointForVpg(zvmIp, sessionId, vpgId):
    url = "https://" + zvmIp + ":9669/v1/vpgs/" + vpgId + "/checkpoints"
    headers = {'x-zerto-session': sessionId, 'content-type': 'application/json'}
r = requests.get(url, headers=headers, verify=False)
print r.status_code
if(r.status_code != requests.codes.ok):
    raise Exception("Failed GetLatestCheckpointForVpg")
allcpData = json.loads(r.text)
print str(len(allcpData))
if(len(allcpData) == 0):
    raise Exception("no checkpoints found for vpg " + vpgId)
sortedCps = sorted(allcpData, key=lambda x: x["TimeStamp"], reverse=True)
latestCp = sortedCps[0]
print latestCp
return latestCp["CheckpointIdentifier"]
```

```python
def StartFailoverTest(zvmIp, sessionId, vpgId, cpId):
    url = "https://" + zvmIp + ":9669/v1/vpgs/" + vpgId + "/failovertest"
    payload = {"CheckpointIdentifier": cpId}
    dataval = json.dumps(payload)
    headers = {'x-zerto-session': sessionId, 'content-type': 'application/json'}
    r = requests.post(url, data=dataval, headers=headers, verify=False)
    print r.status_code
    if(r.status_code != requests.codes.ok):
        raise Exception("Failed StartFailoverTest")
    taskId = json.loads(r.text)
    print "task id = " + taskId
    return taskId

def StopFailoverTest(zvmIp, sessionId, vpgId, status):
    url = "https://" + zvmIp + ":9669/v1/vpgs/" + vpgId + "/failoverteststop"
    payload = {"FailoverTestSuccess": status, "FailoverTestSummary": "Automatically"}
    dataval = json.dumps(payload)
    headers = {'x-zerto-session': sessionId, 'content-type': 'application/json'}
    r = requests.post(url, data=dataval, headers=headers, verify=False)
    print r.status_code
    if(r.status_code != requests.codes.ok):
        raise Exception("Failed StopFailoverTest")
    taskId = json.loads(r.text)
    print "task id = " + taskId
    return taskId

def GetVmNamesForVpg(zvmIp, sessionId, vpgName):
    url = "https://" + zvmIp + ":9669/v1/vms?vpgName=" + vpgName
    headers = {'x-zerto-session': sessionId, 'content-type': 'application/json'}
    r = requests.get(url, headers=headers, verify=False)
    print r.status_code
    if(r.status_code != requests.codes.ok):
        raise Exception("Failed GetVmNamesForVpg")
    allvms = json.loads(r.text)
    print allvms
    res = []
    for vm in allvms:
        print "cur"
        print vm
        res.append(vm["VmName"])
    print res
    return res

def IsTaskComplete(zvmIp, sessionId, stopFotTaskId):
    url = "https://" + zvmIp + ":9669/v1/tasks/" + stopFotTaskId
    headers = {'x-zerto-session': sessionId, 'content-type': 'application/json'}
    r = requests.get(url, headers=headers, verify=False)
    print r.status_code
    if(r.status_code != requests.codes.ok):
        raise Exception("Failed getting tasks")
    tasksRes = json.loads(r.text)
    print tasksRes
    taskStatus = tasksRes["Status"]
    state = taskStatus["State"]
    print state
    if(state == 4 or state == 5):
        raise Exception("task failed")
    res = state == 6
    print res
    return res
```

stringPowerShell Scripts

**Note:** To invoke Zerto Virtual Replication APIs from PowerShell, you must install PowerShell version 4.0 or higher.

The following code sample gets a list of VPGs from a specific Zerto Virtual Manager, and gets a VPG identifier based on a VPG name.

```powershell
$strZVMIP = "{{ZVM IP}}"
$strZVMPort = "{{ZVM HTTPS port}}"
$strZVMUser = "{{ZVM user}}"
$strZVMPwd = "{{ZVM user password}}"

## Perform authentication so that Zerto APIs can run. Return a session identifier that needs to be inserted in the header for subsequent requests.

```powershell
definition getxZertoSession ($userName, $password)
    $baseURL = "https://" + $strZVMIP + ":" + $strZVMPort
    $xZertoSessionURL = $baseURL + "/v1/session/add"
    $authInfo = "{{AuthenticationMethod": "1"}
    $authInfo = [System.Convert]::ToBase64String($authInfo)
    $headers = @{Authorization=((Basic $authInfo))}
    $body = ["\"AuthenticationMethod\"": "1"]
    $contentType = "application/json"
    $xZertoSessionResponse = Invoke-WebRequest -Uri $xZertoSessionURL -Headers $headers -Method POST -Body $body -Content-Type $contentType
    return $xZertoSessionResponse.headers.get_item("x-zerto-session")

# Extract x-zerto-session from the response, and add it to the actual API:
$xZertoSession = getxZertoSession $strZVMUser $strZVMPwd
$zertoSessionHeader = @"{x-zerto-session}=$xZertoSession"
$zertoSessionHeader_xml = @"{Accept}="application/xml"\"x-zerto-session\"=$xZertoSession"

# Invoke the Zerto API:
$vpgListApiUrl = "https://" + $strZVMIP + ":" + $strZVMPort + "/v1/vpgs"
$VPGNAME = "MyApp"

# Iterate with XML:
$vpgListXML = Invoke-RestMethod -Uri $vpgListApiUrl -Headers $zertoSessionHeader_xml -Content-Type "application/xml" -Body $body
foreach ($vpg in $vpgListXML.ArrayOfVpgApi.VpgApi){
    if ($vpg.VpgName -eq $VPGNAME){
        $tmpVpgIdentifier = $vpg.VpgIdentifier
        break
    }
}

# Iterate with JSON:
$vpgListJSON = Invoke-RestMethod -Uri $vpgListApiUrl -Headers $zertoSessionHeader -Content-Type "application/json" -Body $body
foreach ($vpg in $vpgListJSON){
    if ($vpg.VpgName -eq $VPGNAME){
        $tmpVpgIdentifier = $vpg.VpgIdentifier
        break
    }
}
write-host $tmpVpgIdentifier
```

## End of script
CHAPTER 3: INVOKING RESTFUL
APIS FROM POWERSHELL SCRIPTS

This chapter describes how to invoke Zerto Virtual Replication APIs using Windows PowerShell. It shows how to authenticate with the service and shows the different ways to send data to, and receive data from, the Zerto Virtual Replication APIs.

If you intend to invoke Zerto Virtual Replication APIs from PowerShell, you must install PowerShell version 4.0 or higher.

All references to Zerto Virtual Replication RESTful APIs assume the following base URL:

https://zvm-ip:zvm-port/v1

Where:

- **zvm-ip**: The IP address of the Zerto Virtual Manager where the API is run.
- **zvm-port**: The port to access the Zerto Virtual Manager. The default port is 9669.

### Authentication Methods

The following basic authentication methods are available:

**Windows-based authentication (default)** - The username and password are authenticated by Windows on the host on which the Zerto Virtual Manager is installed.

**Hypervisor manager authentication** - The username and password are authenticated by the VMware vCenter Server or Microsoft SCVMM to which the Zerto Virtual Manager is connected.

To authenticate, invoke the following API:

https://zvm-ip:zvm-port/v1/session/add

The response to this request contains a variable called **x-zerto-session** that contains a unique session ID that is used in subsequent requests to APIs.

### Authentication Sample

The following code snippet shows an example of authentication.

```powershell
$baseURL = "https://" + $strZVMIP + ":"+$strZVMPort
$xZertoSessionURL = $baseURL +"/v1/session/add"
#$authInfo = @('{0}:{1}' -f $userName, $password)
$authInfo = [System.Convert]::ToBase64String($authInfo)
$headers = @{Authorization=("Basic {0}" -f $authInfo)
$body = @"{AuthenticationMethod": "1")'
$contentType = "application/json"
$xZertoSessionResponse = Invoke-WebRequest -Uri $xZertoSessionURL -Headers $headers -Method POST -Body $body -ContentType $contentType
return $xZertoSessionResponse.headers.get_item("x-zerto-session")
```
where the code is created using the following steps, with each steps identified by the line number in the above table:

1. Create an authentication object array from the username and password.
   ```powershell
   $authInfo = "({0}:{1})" -f $userName,$password
   ```
2. Convert the authentication object to UTF8.
   ```powershell
   ```
3. Convert the information to base64 format.
   ```powershell
   $authInfo = [System.Convert]::ToBase64String($authInfo)
   ```
4. Build the basic authentication format into a header variable.
   ```powershell
   $headers = @{Authorization="Basic {0}" -f $authInfo}
   ```
5. Build the session request URL.
   ```powershell
   $xZertoSessionURL = $baseURL +"/v1/session/add"
   ```
6. Authenticate the user’s credentials. The fields `body` and `contentType` authenticate a user’s credentials. By default, Windows authentication is used, and these fields are optional. To authenticate with hypervisor manager credentials, add the following:
   ```powershell
   $body = '{"AuthenticationMethod": "1"}'
   $contentType = "application/json"
   ```
7. Invoke the HTTP request to the specified URL, given the authentication header. The response will contain a header with the `x-zerto-session` variable.
   ```powershell
   $xZertoSessionResponse = Invoke-WebRequest -Uri $xZertoSessionURL -Headers $headers
   -Method POST -Body $body -ContentType $contentType
   ```
   If you are authenticating with the default, Windows credentials, remove `"-Body $body -ContentType $contentType"`.
8. Extract the `x-zerto-session` value from the response headers.
   ```powershell
   $xZertoSessionResponse.headers.get_item("x-zerto-session")
   ```

**Using the x-zerto-session Variable**

Once you have obtained the `x-zerto-session` unique session identifier, build a header for subsequent HTTP requests containing the session identifier:

```powershell
$zertSessionHeader = @{"x-zerto-session"=$xZertoSession}
```

where `$xZertoSession` holds the unique session identifier.

**Basic Invocation**

Running Zerto Virtual Replication APIs from PowerShell is based on the `Invoke-RestMethod` command.

A basic invocation is similar to the following:

```powershell
$vpgListApiUrl = "https://" + $strZVMIP + ":"+$strZVMPort+"/v1/vpgs"
Invoke-RestMethod -Uri $vpgListApiUrl -Headers $zertSessionHeader
```

**Input Format**

If the request requires parameters that do not reside in the URL, such as POST requests, specify the input format type using the `-ContentType` argument and switch between XML and JSON.
## Output Format

The output formats of the API requests can be parsed as XML or JSON. PowerShell offers additional formats to which the data can be converted. You also have the ability to save the response to a file.

For example, running the `baseUrl/v1/VPGs` request returns a list of VPGs configured for a specific site.

### Example: Perform an Action

The following code sample creates a VPG with a list of VMs specified in a file.

Both pre-recovery and post-recovery scripts are run by the ZVM service on the ZVM machine. The account running the ZVM service is the account that will run the scripts when they are executed.

```powershell
# Parameters Section
$strZVMIP = "Zerto Virtual Manager IP"
$strZVMPort = "Zerto Virtual Manager HTTPS port"
$strZVMUser = "Zerto Virtual Manager user"
$strZVMPw = "Zerto Virtual Manager user password"
$sourceSiteName = "protected site name"
$targetSiteName = "recovery site name"
$targetDataStoreName = "recovery storage name in the recovery site for the VPG"
$vpgName = "name of the VPG you want to create"
$unProtectedVMsCSVFile = "name of the file that has the names of the VMs to add to the VPG. The file must not have headers, and the VM names must be separated with commas, without spaces between the names. For example, the first row in the file would look like this: vm1,vm2,vm3"
$BASEURL = "https://" + $strZVMIP + ":" + $strZVMPort + "/v1/" # base URL for all APIs

$zertoSessionHeader_xml = @ { "Accept" = "application/xml" "x-zerto-session" = $xZertoSession }

## Function Definitions

### Get a site identifier by invoking Zerto APIs, given a Zerto API session and a site name:

```powershell
function getSiteIdentifierByName ($sessionHeader, $siteName) {
    $url = $BASEURL + "virtualizationsites"
    $response = Invoke-RestMethod -Uri $url -Headers $zertoSessionHeader_xml -ContentType "application/xml"
    ForEach ($site in $response<ArrayOfVirtualizationSiteApi.VirtualizationSiteApi) {
        if ($site.VirtualizationSiteName -eq $siteName) {
            return $site.SiteIdentifier
        }
    }
}
```

### Get a storage identifier by invoking Zerto APIs, given a Zerto Virtual Replication API session and a storage name:

```powershell
function getDatastoreIdentifierByName ($sessionHeader, $siteIdentifier, $datastoreName) {
    $url = $BASEURL + "virtualizationsites" + $siteIdentifier + "/datastores"
    $response = Invoke-RestMethod -Uri $url -Headers $zertoSessionHeader_xml -ContentType "application/xml"
    ForEach ($datastore in $response<ArrayOfDatastoreNativeApi.DatastoreNativeApi) {
        if ($datastore.DatastoreName -eq $datastoreName) {
            return $datastore.DatastoreIdentifier
        }
    }
}
```
Invoking RESTful APIs from PowerShell Scripts

```powershell
#Get unprotected VM identifiers by invoking Zerto APIs, given a Zerto API session, a site identifier, and a list of VMs to add to the VPG:
function getUnprotectedVMsIdentifiers($sessionHeader, $siteIdentifier, $VMNames)
{
    $url = $BASEURL + "virtualizationsites/"+$siteIdentifier + "/vms"
    $unprotectedVMsIdentifiers = @()
    $response = Invoke-RestMethod -Uri $url -Headers $zertoSessionHeader_xml -ContentType "application/xml"
    ForEach ($vm in $response.ArrayOfVmNativeApi.VmNativeApi) {
        if ($VMNames.IndexOf($vm.VmName) -gt -1){
            $unprotectedVMsIdentifiers+=($vm.VmIdentifier)
        }
    }
    return $unprotectedVMsIdentifiers
}

#Authenticate with Zerto APIs: create a Zerto API session and return it, to be used in other APIs
function getZertoXSession()
{
    #Authenticate with Zerto APIs:
    $xZertoSessionURL = "https://" + $strZVMIP + ":"+$strZVMPort+"/v1/session/add"
    $authInfo = ("{0}:{1}" -f $strZVMUser,$strZVMPw)
    $authInfo = [System.Convert]::ToBase64String($authInfo)
    $headers = @{Authorization=("Basic {0}" -f $authInfo))}
    $xZertoSessionResponse = Invoke-WebRequest -Uri $xZertoSessionURL -Headers $headers -Method POST
    #Extract x-zerto-session from the response and add it to the actual API:
    $xZertoSession = $xZertoSessionResponse.headers.get_item("x-zerto-session")
    return $xZertoSession
}

#Build VM elements to be added to the VPGs API, based on a list of VM identifiers
function buildVMsElement ($VMs)
{
    $response = "<VmsIdentifiers>
    ForEach ($vm in $VMs) {
        $response+="<string xmlns="'+"http://schemas.microsoft.com/2003/10/Serialization/Arrays"'+">"+$vm+"</string>"
    }
    $response += "</VmsIdentifiers>"
    return $response
}
```
#Script starts here:
$zertoSessionHeader = @{"x-zerto-session"=$zertoSession}
$sourceSiteIdentifier = getSiteIdentifierBySiteName $zertoSessionHeader $sourceSiteName
$targetSiteIdentifier = getSiteIdentifierBySiteName $zertoSessionHeader $targetSiteName
$dataStoreIdentifier = getDatastoreIdentifierBySiteName $zertoSessionHeader $dataStoreName
$unprotectedVMNames = Get-Content $unProtectedVMsCSVFile | %{$_.Split(",")}
$vmsIdentifiers = getUnprotectedVMsIdentifiers $zertoSessionHeader $sourceSiteIdentifier $unprotectedVMNames
$vmsIdentifiersElement = buildVMsElement $vmsIdentifiers
#Create the URL and body of the VPGs request:
$createVPGUrl = $BASEURL+"vpgs"
$vpgsRequestBody = "<VpgCreateDataApi xmlns="http://schemas.zerto.com/zvm/api">" +"<DatastoreIdentifier">+$dataStoreIdentifier +"</DatastoreIdentifier>" +"<SourceSiteIdentifier">+$sourceSiteIdentifier+"</SourceSiteIdentifier>" +"<TargetSiteIdentifier">+$targetSiteIdentifier+"</TargetSiteIdentifier>" +"$vmsIdentifiersElement+"<VpgName>$vpgName</VpgName> </VpgCreateDataApi>"
#Invoke the Zerto API:
Invoke-RestMethod -Uri $createVPGUrl -Headers $zertoSessionHeader -Body $vpgsRequestBody -ContentType "application/xml" -method POST
##End of script
CHAPTER 4: ZERTO VIRTUAL REPLICATION APIS

This chapter provides reference material about the Zerto Virtual Replication RESTful APIs.

All APIs

The following APIs are available:

<table>
<thead>
<tr>
<th>API</th>
<th>SUBJECT OF API</th>
<th>METHOD</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>/v1/</td>
<td>List available APIs</td>
<td>GET</td>
<td>Retrieves the list of available Zerto Virtual Replication v1 REST APIs. See “/v1/ API”, on page 33.</td>
</tr>
<tr>
<td>/v1/alerts</td>
<td>Alert information</td>
<td>GET</td>
<td>Retrieves information about Zerto Virtual Replication alerts. See “Alerts API” and “Alerts: GET”, on page 35.</td>
</tr>
<tr>
<td>/v1/alerts</td>
<td>Alert actions</td>
<td>POST</td>
<td>Dismisses or undismisses an alert. See “Alerts API” and “Alerts: POST”, on page 49.</td>
</tr>
<tr>
<td>/v1/events</td>
<td>Events</td>
<td>GET</td>
<td>Retrieves information about Zerto Virtual Replication events. See “Events API”, on page 50.</td>
</tr>
<tr>
<td>/v1/localsite</td>
<td>Local site information</td>
<td>GET</td>
<td>Retrieves information about the local site where the API is run. See “Local Site API”, on page 60.</td>
</tr>
<tr>
<td>/v1/peersites</td>
<td>Peer site information</td>
<td>GET</td>
<td>Retrieves information about the peer sites for the site where the API is run. See “Peer Sites API” and “Peersites - GET”, on page 63.</td>
</tr>
<tr>
<td>/v1/peersites</td>
<td>Peer site action</td>
<td>POST</td>
<td>Adds a peer site. See “Peer Sites API” and “Peersites - POST”, on page 65.</td>
</tr>
<tr>
<td>/v1/peersites</td>
<td>Peer site action</td>
<td>DELETE</td>
<td>Unpair a peer site. See “Peer Sites API” and “Peersites - POST”, on page 65.</td>
</tr>
<tr>
<td>/v1/serviceprofiles</td>
<td>Service profiles</td>
<td>GET</td>
<td>Retrieves service profile information. See “Service Profiles API”, on page 68.</td>
</tr>
<tr>
<td>/v1/session</td>
<td>Session management</td>
<td>POST</td>
<td>Starts a session. See “Session API” and “Session: POST”, on page 70.</td>
</tr>
<tr>
<td>/v1/session</td>
<td>Session management</td>
<td>DELETE</td>
<td>Ends a session. See “Session API” and “Session: DELETE”, on page 71.</td>
</tr>
<tr>
<td>/v1/tasks</td>
<td>Task information</td>
<td>GET</td>
<td>Retrieves information about tasks run at a site. See “Tasks API”, on page 74.</td>
</tr>
<tr>
<td>/v1/virtualizationsites</td>
<td>Site information</td>
<td>GET</td>
<td>Retrieves information about the hypervisor site where the API is run and the paired hypervisor sites as well as information about the resources at a specified site. See “Virtualization Sites API”, on page 83.</td>
</tr>
<tr>
<td>/v1/vms</td>
<td>Protected virtual machine information</td>
<td>GET</td>
<td>Retrieves information about protected virtual machines. See “Protected VMs API”, on page 88.</td>
</tr>
<tr>
<td>/v1/vpgs</td>
<td>VPG information</td>
<td>GET</td>
<td>Retrieves information about VPGs. See “VPGs API” and “VPGs: GET”, on page 97.</td>
</tr>
<tr>
<td>/v1/vpgs</td>
<td>VPG actions</td>
<td>POST</td>
<td>Performs actions on a VPG. See “VPGs API” and “VPGs: POST”, on page 109.</td>
</tr>
<tr>
<td>/v1/vpgs</td>
<td>VPG actions</td>
<td>DELETE</td>
<td>Deletes a VPG. See “VPGs API” and “VPGs: DELETE”, on page 114.</td>
</tr>
<tr>
<td>/v1/vpgSettings</td>
<td>Manage a VPG</td>
<td>GET</td>
<td>Manages a vpgSettings object. See “VPG Management API” and “VPG Settings: GET”, on page 119.</td>
</tr>
</tbody>
</table>
Zerto Virtual Replication RESTful API Reference Guide - Version 5.5
Zerto Virtual Replication APIs

<table>
<thead>
<tr>
<th>API</th>
<th>SUBJECT OF API</th>
<th>METHOD</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/v1/vpgSettings</code></td>
<td>Manage a VPG</td>
<td>POST</td>
<td>Performs actions on a vpgSettings object. See “VPG Management API” and “VPG Settings: POST”, on page 142.</td>
</tr>
<tr>
<td><code>/v1/vpgSettings</code></td>
<td>Manage a VPG</td>
<td>PUT</td>
<td>Enables editing a vpgSettings object. See “VPG Management API” and “VPG Settings: PUT”, on page 155.</td>
</tr>
<tr>
<td><code>/v1/vpgSettings</code></td>
<td>Manage a VPG</td>
<td>DELETE</td>
<td>Deletes a vpgSettings object. See “VPG Management API” and “VPG Settings: DELETE”, on page 175.</td>
</tr>
<tr>
<td><code>/v1/vras</code></td>
<td>VRA information</td>
<td>GET</td>
<td>Retrieves VRA information. See “VRAs API” and “VRAs: GET”, on page 177.</td>
</tr>
<tr>
<td><code>/v1/vras</code></td>
<td>VRA actions</td>
<td>POST</td>
<td>Performs actions on a VRA. See “VRAs API” and “VRAs: POST”, on page 181.</td>
</tr>
<tr>
<td><code>/v1/vras</code></td>
<td>VRA actions</td>
<td>PUT</td>
<td>Enables editing a VRA. See “VRAs API” and “VRAs: PUT”, on page 183.</td>
</tr>
<tr>
<td><code>/v1/vras</code></td>
<td>VRA actions</td>
<td>DELETE</td>
<td>Deletes a VRA. See “VRAs API” and “VRAs: DELETE”, on page 185.</td>
</tr>
<tr>
<td><code>/v1/zorgs</code></td>
<td>ZORG information</td>
<td>GET</td>
<td>Retrieves ZORG information. See in “ZORGs API”, on page 186.</td>
</tr>
<tr>
<td><code>/ZvmService/ResourcesReport</code></td>
<td>Run the resource report</td>
<td>GET</td>
<td>Generates a resource report about the virtual machines being protected to a recovery site. See “Resources Report API”, on page 187.</td>
</tr>
<tr>
<td>vCD APIs</td>
<td></td>
<td></td>
<td>View, create, update or delete vCD VPGs. See “Managing vCD APIs”, on page 196.</td>
</tr>
</tbody>
</table>

/v1/ API

/v1/ returns the list of all Zerto Virtual Replication RESTful v1 APIs except for the Zerto Virtual Replication ResourcesReport RESTful API, under /ZvmService.

URL

/v1/ APIs  https://zvm_ip:port/v1/

Where:

<table>
<thead>
<tr>
<th>zvm_ip</th>
<th>The IP address of the Zerto Virtual Manager where the API is run.</th>
</tr>
</thead>
<tbody>
<tr>
<td>port</td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
</tbody>
</table>

HTTP Method

GET

Security

The API is exposed over HTTPS. This API does not require logging in with authentication to run the API.

See Also

The Zerto Virtual Replication Resources Report: Resources Report API
Zerto Virtual Replication RESTful API Reference Guide - Version 5.5
Zerto Virtual Replication APIs

**Format**

Json, XML

**Request Format**

The request body is empty.

**Json Response Format**

The following is an example response Json body for `https://zvm_ip:port/v1/`.

```
[
  {
    "href": "String content",
    "rel": "String content",
    "type": "String content"
  }
]
```

**XML Response Format**

For the XML response format, see “V1/ API XML Response Format”, on page 233.

**Response Values**


<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>href</td>
<td>The URL used.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level for the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
</tbody>
</table>

**Code Examples**

The `/v1/` API does not require a session.

**/v1/ cURL Code Example**

Retrieve the list of APIs.

```
$ curl https://zvm_ip:9669/v1/
```

Returns the list of APIs.

```
[{
  "href": "https://10.100.0.51:9669/v1/events","rel":"down","type":"IEventsService"},
  {
    "href": "https://10.100.0.51:9669/v1/vpgs","rel":"down","type":"IVpgService"},
  {
    "href": "https://10.100.0.51:9669/v1/vms","rel":"down","type":"IVmService"},
  {
    "href": "https://10.100.0.51:9669/v1/vras","rel":"down","type":"IVraService"},
  {
    "href": "https://10.100.0.51:9669/v1/peersites",
    "rel": "down",
    "type": "IPeerSitesService"},
  {
    "href": "https://10.100.0.51:9669/v1/session",
    "rel": "down",
    "type": "ISessionService"},
  {
    "href": "https://10.100.0.51:9669/v1/tasks",
    "rel": "down",
    "type": "ITasksService"},
  {
    "href": "https://10.100.0.51:9669/v1/serviceprofiles",
    "rel": "down",
    "type": "IServiceProfilesService"},
  {
    "href": "https://10.100.0.51:9669/v1/virtualizationsites",
    "rel": "down",
    "type": "IVirtualizationSitesService"},
  {
    "href": "https://10.100.0.51:9669/v1/zorgs",
    "rel": "down",
    "type": "IZorgsService"},
  {
    "href": "https://10.100.0.51:9669/v1/localsite",
    "rel": "down",
    "type": "ILocalSiteService"},
  {
    "href": "https://10.100.0.51:9669/v1/alerts",
    "rel": "down",
    "type": "IAlertsService"}
]
```

For more code examples, see “cURL Code”, on page 22.
Alerts API

/v1/alerts returns information about alerts and dismisses or und dismisses a specific alert. The following API are available:

- “Alerts: GET”, below
- “Alerts: POST”, on page 49

<table>
<thead>
<tr>
<th>PURPOSE</th>
<th>METHOD</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Information for all alerts</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/alerts</td>
</tr>
<tr>
<td>Information for one alert</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/alerts/{alertId}</td>
</tr>
<tr>
<td>Valid values for alert entities</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/alerts/entities</td>
</tr>
<tr>
<td>Valid values for alert help identifiers</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/alerts/helpidentifiers</td>
</tr>
<tr>
<td>Valid values for alert levels</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/alerts/levels</td>
</tr>
<tr>
<td>Dismisses an alert</td>
<td>POST</td>
<td>https://zvm_ip:port/v1/alerts/{alertId}/dismiss</td>
</tr>
<tr>
<td>Undismisses an alert</td>
<td>POST</td>
<td>https://zvm_ip:port/v1/alerts/{alertId}/undismiss</td>
</tr>
</tbody>
</table>

HTTP Methods

GET, POST

Security

The API is exposed over HTTPS. Client code must use the x-zerto-session HTTP authorization header.

See Also

Starting a session: Session: POST

Format

Json, XML

Alerts: GET

Returns information about alerts.
### Alerts API

#### URL

- **All alerts**
  ```
  https://zvm_ip:port/v1/alerts
  ```

- **Filtered alerts**
  ```
  https://zvm_ip:port/v1/alerts?startDate={STARTDATE}&endDate={ENDDATE}&vpgIdentifier={VPGIDENTIFIER}&zorgIdentifier={ZORGIDENTIFIER}&siteIdentifier={SITEIDENTIFIER}&level={LEVEL}&entity={ENTITY}&helpIdentifier={HELPIDENTIFIER}&isDismissed={ISDISMISSED}
  ```

- **Single alert**
  ```
  https://zvm_ip:port/v1/alerts/{alertId}
  ```

- **Valid alert entities**
  ```
  https://zvm_ip:port/v1/alerts/entities
  ```

- **Valid alert help identifiers**
  ```
  https://zvm_ip:port/v1/alerts/helpidentifiers
  ```

- **Valid alert levels**
  ```
  https://zvm_ip:port/v1/alerts/levels
  ```

#### Where:

- **zvm_ip**
  The IP address of the Zerto Virtual Manager where the API is run.

- **port**
  The port to access the Zerto Virtual Manager. The default port is 9669.

- **alertId**
  The identifier of the alert for which information is returned or which is dismissed or undismissed.

- **Filters**
  Filters are optional and any combination of filters is valid. When no filter is specified, all alerts are returned. Filters are not case-sensitive.

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>startDate</strong></td>
<td>The starting date for the list of events, supplied as a date with the format of the Zerto Virtual Manager where the API runs, for example, yyyy-mm-dd. You can also specify a local time with the following format: yyyy-mm-ddThh:mm:ss. Adding Z to the end of the time sets the time to UTC.</td>
</tr>
<tr>
<td><strong>endDate</strong></td>
<td>The end date for the list, supplied as a date with the format of the Zerto Virtual Manager where the API runs, for example, yyyy-mm-dd. You can also specify a local time with the following format: yyyy-mm-ddThh:mm:ss. Adding Z to the end of the time sets the time to UTC.</td>
</tr>
<tr>
<td><strong>vpgIdentifier</strong></td>
<td>The identifier of the VPG for which you want to return alerts.</td>
</tr>
<tr>
<td><strong>zorgIdentifier</strong></td>
<td>The identifier of the ZORG, Zerto organization, defined in the Zerto Cloud Manager</td>
</tr>
<tr>
<td><strong>siteIdentifier</strong></td>
<td>The internal site identifier.</td>
</tr>
<tr>
<td><strong>level</strong></td>
<td>The alert level: warning or error</td>
</tr>
<tr>
<td><strong>entity</strong></td>
<td>The type of entity to return. Possible values are:</td>
</tr>
<tr>
<td></td>
<td><strong>Zvm</strong>: Alerts that are triggered when there is a problem with a Zerto Virtual Manager.</td>
</tr>
<tr>
<td></td>
<td><strong>Vra</strong>: Alerts that are triggered when there is a problem with a VRA.</td>
</tr>
<tr>
<td></td>
<td><strong>Vpg</strong>: Alerts that are triggered when there is a problem with a VPG.</td>
</tr>
<tr>
<td></td>
<td><strong>CloudConnector</strong>: Alerts that are triggered when there is a problem with a Zerto Cloud Connector.</td>
</tr>
<tr>
<td></td>
<td><strong>Storage</strong>: Alerts that are triggered when there is a problem with a datastore specified in a VPG.</td>
</tr>
<tr>
<td></td>
<td><strong>License</strong>: Alerts that are triggered when there is a problem with a Zerto Virtual Manager license.</td>
</tr>
<tr>
<td></td>
<td><strong>Zcm</strong>: Alerts that are triggered when there is a problem with a Zerto Cloud Manager.</td>
</tr>
<tr>
<td></td>
<td><strong>FileRecoveryComponent</strong>: Alerts that are triggered when there is a problem with file level recover.</td>
</tr>
<tr>
<td><strong>helpIdentifier</strong></td>
<td>The help identifier associated with the alert. The following alert identifiers are available:</td>
</tr>
<tr>
<td></td>
<td><strong>AWS0001</strong>: AWS S3 bucket removed</td>
</tr>
<tr>
<td></td>
<td><strong>BCK0001</strong>: Offsite backup fails</td>
</tr>
<tr>
<td></td>
<td><strong>BCK0005</strong>: Offsite backup repository disconnected</td>
</tr>
<tr>
<td></td>
<td><strong>BCK0006</strong>: Offsite backup repository disconnected</td>
</tr>
<tr>
<td>LIC0001</td>
<td>License exceeded</td>
</tr>
<tr>
<td>LIC0003</td>
<td>License about to expire</td>
</tr>
<tr>
<td>LIC0004</td>
<td>License expired and exceeded</td>
</tr>
<tr>
<td>LIC0006</td>
<td>License expired</td>
</tr>
<tr>
<td>LIC0007</td>
<td>License exceeded</td>
</tr>
<tr>
<td>LIC0009</td>
<td>Public cloud replication violation</td>
</tr>
<tr>
<td>LIC0010</td>
<td>VCD violation</td>
</tr>
<tr>
<td>LIC0011</td>
<td>Cross replication violation</td>
</tr>
<tr>
<td>STR0001</td>
<td>Datastore not accessible</td>
</tr>
<tr>
<td>STR0002</td>
<td>Datastore full</td>
</tr>
<tr>
<td>STR0004</td>
<td>Datastore low in space</td>
</tr>
<tr>
<td>VCD0001</td>
<td>vCenter Server for the Org vDC is not found</td>
</tr>
<tr>
<td>VCD0002</td>
<td>Org vDC is defined in multiple vCenter Servers</td>
</tr>
<tr>
<td>VCD0003</td>
<td>Org vDC storage profile not found in vCenter Server</td>
</tr>
<tr>
<td>VCD0004</td>
<td>Provider vDC storage profile not found in vCenter Server</td>
</tr>
<tr>
<td>VCD0005</td>
<td>Org vDC network not retrieved</td>
</tr>
<tr>
<td>VCD0006</td>
<td>Provider vDC metadata not found</td>
</tr>
<tr>
<td>VCD0007</td>
<td>Org vDC resource pool not retrieved</td>
</tr>
<tr>
<td>VCD0010</td>
<td>OrgNetwork not retrieved</td>
</tr>
<tr>
<td>VCD0014</td>
<td>vCD disconnection</td>
</tr>
<tr>
<td>VCD0015</td>
<td>AMQP-server disconnection</td>
</tr>
<tr>
<td>VCD0016</td>
<td>Provider vDC datastore not found</td>
</tr>
<tr>
<td>VCD0017</td>
<td>Metadata not accessible</td>
</tr>
<tr>
<td>VCD0018</td>
<td>Duplicated MAC addresses</td>
</tr>
<tr>
<td>VCD0020</td>
<td>VM inconsistency in vApp</td>
</tr>
<tr>
<td>VCD0021</td>
<td>VM inconsistency in vApp</td>
</tr>
<tr>
<td>Code</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>VPG0003</td>
<td>VPG has low journal history</td>
</tr>
<tr>
<td>VPG0004</td>
<td>VPG has low journal history</td>
</tr>
<tr>
<td>VPG0005</td>
<td>VPG in error state</td>
</tr>
<tr>
<td>VPG0006</td>
<td>VPG missing configuration details</td>
</tr>
<tr>
<td>VPG0007</td>
<td>VPG replication paused</td>
</tr>
<tr>
<td>VPG0008</td>
<td>VPG rollback failed</td>
</tr>
<tr>
<td>VPG0009</td>
<td>VPG target RPO exceeded</td>
</tr>
<tr>
<td>VPG0010</td>
<td>VPG target RPO exceeded</td>
</tr>
<tr>
<td>VPG0011</td>
<td>VPG test overdue</td>
</tr>
<tr>
<td>VPG0012</td>
<td>VPG test overdue</td>
</tr>
<tr>
<td>VPG0014</td>
<td>VPG waiting for commit or rollback</td>
</tr>
<tr>
<td>VPG0015</td>
<td>Resources not enough to support VPG</td>
</tr>
<tr>
<td>VPG0016</td>
<td>Resources pool not found</td>
</tr>
<tr>
<td>VPG0017</td>
<td>VPG protection paused</td>
</tr>
<tr>
<td>VPG0018</td>
<td>VMs in VPG not configured with a storage profile</td>
</tr>
<tr>
<td>VPG0019</td>
<td>VPG recovery storage profile disabled</td>
</tr>
<tr>
<td>VPG0020</td>
<td>VPG recovery storage profile not found</td>
</tr>
<tr>
<td>VPG0021</td>
<td>VPG recovery storage profile not found</td>
</tr>
<tr>
<td>VPG0022</td>
<td>VPG recovery storage profile disabled</td>
</tr>
<tr>
<td>VPG0023</td>
<td>VPG recovery storage profile not found</td>
</tr>
<tr>
<td>VPG0024</td>
<td>VPG recovery storage profile does not include active datastores</td>
</tr>
<tr>
<td>VPG0025</td>
<td>vApp network mapping not defined</td>
</tr>
<tr>
<td>VPG0026</td>
<td>VPG recovery storage profile changed</td>
</tr>
<tr>
<td>VPG0027</td>
<td>VPG includes VMs that are no longer protected</td>
</tr>
<tr>
<td>VPG0028</td>
<td>Corrupted Org vDC network mapping</td>
</tr>
<tr>
<td>VPG0035</td>
<td>VPG protected resources not in ZORG</td>
</tr>
<tr>
<td>VPG0036</td>
<td>VPG recovery resources not in ZORG</td>
</tr>
<tr>
<td>VPG0037</td>
<td>Journal history is compromised</td>
</tr>
<tr>
<td>VPG0038</td>
<td>Journal history is compromised</td>
</tr>
<tr>
<td>VPG0039</td>
<td>RDM has an odd number of blocks</td>
</tr>
<tr>
<td>VPG0040</td>
<td>Virtual machine hardware mismatch with recovery site</td>
</tr>
<tr>
<td>VPG0041</td>
<td>Virtual machine running Windows 2003</td>
</tr>
<tr>
<td>VPG0042</td>
<td>Recovery network not found</td>
</tr>
<tr>
<td>VPG0043</td>
<td>SAN policy might cause some volumes to become offline upon recovery</td>
</tr>
<tr>
<td>VPG0044</td>
<td>One or more of the virtual machines in the VPG has a disk with size 0</td>
</tr>
<tr>
<td>Alert Code</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>VRA0001</td>
<td>Host without VRA</td>
</tr>
<tr>
<td>VRA0002</td>
<td>VRA without IP</td>
</tr>
<tr>
<td>VRA0003</td>
<td>Host IP changes</td>
</tr>
<tr>
<td>VRA0004</td>
<td>VRA lost IP</td>
</tr>
<tr>
<td>VRA0005</td>
<td>VRAs not connected</td>
</tr>
<tr>
<td>VRA0006</td>
<td>Datastore for journal disk is full</td>
</tr>
<tr>
<td>VRA0007</td>
<td>I/O error to journal</td>
</tr>
<tr>
<td>VRA0008</td>
<td>Recovery disk and VMs missing</td>
</tr>
<tr>
<td>VRA0009</td>
<td>Recovery disk missing</td>
</tr>
<tr>
<td>VRA0010</td>
<td>Recovery disks turned off</td>
</tr>
<tr>
<td>VRA0011</td>
<td>Recovery disk inaccessible</td>
</tr>
<tr>
<td>VRA0012</td>
<td>Cannot write to recovery disk</td>
</tr>
<tr>
<td>VRA0013</td>
<td>I/O error to recovery disk</td>
</tr>
<tr>
<td>VRA0014</td>
<td>Cloned disks turned off</td>
</tr>
<tr>
<td>VRA0015</td>
<td>Cloned disk inaccessible</td>
</tr>
<tr>
<td>VRA0016</td>
<td>Datastore for clone disk is full</td>
</tr>
<tr>
<td>VRA0017</td>
<td>I/O error to clone</td>
</tr>
<tr>
<td>VRA0018</td>
<td>Protected disk and VM missing</td>
</tr>
<tr>
<td>VRA0019</td>
<td>Protected disk missing</td>
</tr>
<tr>
<td>VRA0020</td>
<td>VM powered off</td>
</tr>
<tr>
<td>VRA0021</td>
<td>VM disk inaccessible</td>
</tr>
<tr>
<td>VRA0022</td>
<td>VM disk incompatible</td>
</tr>
<tr>
<td>VRA0023</td>
<td>VRA cannot be registered.</td>
</tr>
<tr>
<td>VRA0024</td>
<td>VRA removed</td>
</tr>
<tr>
<td>VRA0025</td>
<td>I/O synchronization</td>
</tr>
<tr>
<td>VRA0026</td>
<td>Recovery disk removed</td>
</tr>
<tr>
<td>VRA0027</td>
<td>Journal disk removed</td>
</tr>
<tr>
<td>VRA0028</td>
<td>VRA powered off</td>
</tr>
<tr>
<td>VRA0029</td>
<td>VRA memory low</td>
</tr>
<tr>
<td>VRA0030</td>
<td>Journal size mismatch</td>
</tr>
<tr>
<td>VRA0032</td>
<td>VRA out-of-date</td>
</tr>
<tr>
<td>VRA0035</td>
<td>VRA reconciliation</td>
</tr>
<tr>
<td>VRA0036</td>
<td>For internal use only</td>
</tr>
<tr>
<td>VRA0037</td>
<td>Local MAC Address Conflict</td>
</tr>
<tr>
<td>VRA0038</td>
<td>MAC Address Conflict</td>
</tr>
<tr>
<td>VRA0039</td>
<td>Journal reached configured limit</td>
</tr>
<tr>
<td>VRA0040</td>
<td>Journal space low</td>
</tr>
<tr>
<td>VRA0049</td>
<td>Host rollback failed</td>
</tr>
<tr>
<td>VRA0050</td>
<td>Wrong host password</td>
</tr>
<tr>
<td>VRA0051</td>
<td>For internal use only</td>
</tr>
<tr>
<td>VRA0052</td>
<td>Disk visible but not recognized</td>
</tr>
<tr>
<td>VRA0053</td>
<td>System disk removed</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Alert Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ZCC0001</td>
<td>Zerto Cloud Connector removed</td>
</tr>
<tr>
<td>ZCC0002</td>
<td>Zerto Cloud Connector powered off</td>
</tr>
<tr>
<td>ZCC0003</td>
<td>Orphaned Zerto Cloud Connector</td>
</tr>
<tr>
<td>ZCM0001</td>
<td>No connection to Zerto Virtual Manager</td>
</tr>
<tr>
<td>ZCM0002</td>
<td>Zerto Cloud Manager not support</td>
</tr>
</tbody>
</table>
### Zerto Virtual Replication RESTful API Reference Guide - Version 5.5
### Zerto Virtual Replication APIs

#### Request Format

The request body is empty.

<table>
<thead>
<tr>
<th>Alert Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ZVM0001</td>
<td>No connection to hypervisor manager, such as VMware vCenter Server and Microsoft SCVMM, or to public cloud.</td>
</tr>
<tr>
<td>ZVM0002</td>
<td>No connection to VRA</td>
</tr>
<tr>
<td>ZVM0003</td>
<td>No connection to site</td>
</tr>
<tr>
<td>ZVM0004</td>
<td>Peer site out-of-date</td>
</tr>
<tr>
<td>ZVM0005</td>
<td>Zerto Virtual Manager space low</td>
</tr>
<tr>
<td>ZVM0006</td>
<td>Upgrade available</td>
</tr>
<tr>
<td>ZVM0007</td>
<td>Cannot upgrade</td>
</tr>
<tr>
<td>ZVM0008</td>
<td>Version mismatch</td>
</tr>
<tr>
<td>ZVM0009</td>
<td>Internal error</td>
</tr>
<tr>
<td>ZVM0010</td>
<td>Synchronization between Zerto Virtual Managers</td>
</tr>
<tr>
<td>ZVM0012</td>
<td>Metadata Collection</td>
</tr>
<tr>
<td>ZVM0013</td>
<td>Metadata Collection</td>
</tr>
<tr>
<td>ZVM0014</td>
<td>VRA/Diskbox SCSI GUID mismatch</td>
</tr>
<tr>
<td>ZVM0015</td>
<td>Hyper-V host state</td>
</tr>
<tr>
<td>FLR0001</td>
<td>Files cannot be restored</td>
</tr>
<tr>
<td>Unknown</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>isDismissed</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>True</td>
<td>Returns the list of alerts that have been dismissed.</td>
</tr>
<tr>
<td>False</td>
<td>Returns the list of alerts that have not been dismissed.</td>
</tr>
</tbody>
</table>
Json Response Format

The following is an example response Json body for https://zvm_ip:port/v1/alerts and for https://zvm_ip:port/v1/alerts/{alertId}.

```
[{
   "AffectedVpgs":[
      "href": "String content",
      "identifier": "String content",
      "rel": "String content",
      "type": "String content"
   ],
   "AffectedZorgs":[
      "href": "String content",
      "identifier": "String content",
      "rel": "String content",
      "type": "String content"
   ],
   "Description": "String content",
   "Entity": "String content",
   "HelpIdentifier": "String content",
   "IsDismissed": Boolean,
   "Level": "String content",
   "Link": {
      "href": "String content",
      "identifier": "String content",
      "rel": "String content",
      "type": "String content"
   },
   "Site": {
      "href": "String content",
      "identifier": "String content",
      "rel": "String content",
      "type": "String content"
   },
   "TurnedOn": "/Date(928142400000+0300)/"
}]
```


```
["String content"]
```

XML Response Format

For the XML response format, see “Alerts API XML Response Format”, on page 233.

Response Values


<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>AffectedVpgs</td>
<td>The VPGs affected by the alert.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used to retrieve VPG information for each VPG affected by the event.</td>
</tr>
<tr>
<td>identifier</td>
<td>The unique internal identifier of the VPG.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level for the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>AffectedZorgs</td>
<td>The ZORGs affected by the alert.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used to retrieve ZORG information for each ZORG affected by the event</td>
</tr>
</tbody>
</table>
## Alerts API

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>identifier</td>
<td>The unique internal identifier of the ZORG.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level for the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>Description</td>
<td>The alert description.</td>
</tr>
</tbody>
</table>
| Entity | The entity for which the alert applies. The following alert entities are available:  
  - **Zvm**: Alerts that are triggered when there is a problem with a Zerto Virtual Manager.  
  - **Vra**: Alerts that are triggered when there is a problem with a VRA.  
  - **Vpg**: Alerts that are triggered when there is a problem with a VPG.  
  - **CloudConnector**: Alerts that are triggered when there is a problem with a Zerto Cloud Connector.  
  - **Storage**: Alerts that are triggered when there is a problem with a datastore specified in a VPG.  
  - **License**: Alerts that are triggered when there is a problem with a Zerto Virtual Manager license.  
  - **Zcm**: Alerts that are triggered when there is a problem with a Zerto Cloud Manager.  
  - **FileRecoveryComponent**: Alerts that are triggered when there is a problem with file level recovery. |
| HelpIdentifier | The help identifier associated with the alert. The following alert identifiers are available:  
  - **AWS0001**: AWS S3 bucket removed  
  - **BCK0001**: Offsite backup fails  
  - **BCK0002**: Offsite backup fails  
  - **BCK0005**: Offsite backup repository disconnected.  
  - **BCK0006**: Offsite backup repository disconnected.  
  - **BCK0007**: Offsite backup repository not defined.  
  - **LIC0001**: License exceeded  
  - **LIC0003**: License about to expire  
  - **LIC0004**: License expired and exceeded  
  - **LIC0006**: License expired  
  - **LIC0007**: License exceeded  
  - **LIC0009**: Public cloud replication not supported  
  - **LIC0010**: VCD not supported  
  - **LIC0011**: Multi hypervisor replication not supported  
  - **STR0001**: Datastore not accessible  
  - **STR0002**: Datastore full  
  - **STR0004**: Datastore low in space  
  - **VCD0001**: vCenter Server for the Org vDC is not found  
  - **VCD0002**: Org vDC is defined in multiple vCenter Servers  
  - **VCD0003**: Org vDC storage profile not found in vCenter Server  
  - **VCD0004**: Provider vDC storage profile not found in vCenter Server  
  - **VCD0005**: Org vDC network not retrieved  
  - **VCD0006**: Provider vDC metadata not found  
  - **VCD0007**: Org vDC resource pool not retrieved  
  - **VCD0010**: OrgNetwork not retrieved  
  - **VCD0014**: vCD disconnection  
  - **VCD0015**: AMQP-server disconnection  
  - **VCD0016**: Provider vDC datastore not found  
  - **VCD0017**: Metadata not accessible  
  - **VCD0018**: Duplicated MAC addresses  
  - **VCD0020**: VM inconsistency in vApp  
  - **VCD0021**: VM inconsistency in vApp |
<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>VPG0003</td>
<td>VPG has low journal history</td>
</tr>
<tr>
<td>VPG0004</td>
<td>VPG has low journal history</td>
</tr>
<tr>
<td>VPG0005</td>
<td>VPG in error state</td>
</tr>
<tr>
<td>VPG0006</td>
<td>VPG missing configuration details</td>
</tr>
<tr>
<td>VPG0007</td>
<td>VPG replication paused</td>
</tr>
<tr>
<td>VPG0008</td>
<td>VPG rollback failed</td>
</tr>
<tr>
<td>VPG0009</td>
<td>VPG target RPO exceeded</td>
</tr>
<tr>
<td>VPG0010</td>
<td>VPG target RPO exceeded</td>
</tr>
<tr>
<td>VPG0011</td>
<td>VPG test overdue</td>
</tr>
<tr>
<td>VPG0012</td>
<td>VPG test overdue</td>
</tr>
<tr>
<td>VPG0014</td>
<td>VPG waiting for commit or rollback</td>
</tr>
<tr>
<td>VPG0015</td>
<td>Resources not enough to support VPG</td>
</tr>
<tr>
<td>VPG0016</td>
<td>Resources pool not found</td>
</tr>
<tr>
<td>VPG0017</td>
<td>VPG protection paused</td>
</tr>
<tr>
<td>VPG0018</td>
<td>VMs in VPG not configured with a storage profile</td>
</tr>
<tr>
<td>VPG0019</td>
<td>VPG recovery storage profile disabled</td>
</tr>
<tr>
<td>VPG0020</td>
<td>VPG recovery storage profile not found</td>
</tr>
<tr>
<td>VPG0021</td>
<td>VPG recovery storage profile not found</td>
</tr>
<tr>
<td>VPG0022</td>
<td>VPG recovery storage profile disabled</td>
</tr>
<tr>
<td>VPG0023</td>
<td>VPG recovery storage profile not found</td>
</tr>
<tr>
<td>VPG0024</td>
<td>VPG recovery storage profile does not include active datastores</td>
</tr>
<tr>
<td>VPG0025</td>
<td>vApp network mapping not defined</td>
</tr>
<tr>
<td>VPG0026</td>
<td>VPG recovery storage profile changed</td>
</tr>
<tr>
<td>VPG0027</td>
<td>VPG includes VMs that are no longer protected</td>
</tr>
<tr>
<td>VPG0028</td>
<td>Corrupted Org vDC network mapping</td>
</tr>
<tr>
<td>VPG0035</td>
<td>VPG protected resources not in ZORG</td>
</tr>
<tr>
<td>VPG0036</td>
<td>VPG recovery resources not in ZORG</td>
</tr>
<tr>
<td>VPG0037</td>
<td>Journal history is compromised</td>
</tr>
<tr>
<td>VPG0038</td>
<td>Journal history is compromised</td>
</tr>
<tr>
<td>VPG0039</td>
<td>RDM has an odd number of blocks</td>
</tr>
<tr>
<td>VPG0040</td>
<td>Virtual machine hardware mismatch with recovery site</td>
</tr>
<tr>
<td>VPG0041</td>
<td>Virtual machine running Windows 2003</td>
</tr>
<tr>
<td>VPG0042</td>
<td>Recovery network not found</td>
</tr>
<tr>
<td>VPG0043</td>
<td>SAN policy might cause some volumes to become offline upon recovery</td>
</tr>
<tr>
<td>VPG0044</td>
<td>One or more of the virtual machines in the VPG has a disk with size 0</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>VRA0001:</td>
<td>Host without VRA</td>
</tr>
<tr>
<td>VRA0002:</td>
<td>VRA without IP</td>
</tr>
<tr>
<td>VRA0003:</td>
<td>Host IP changes</td>
</tr>
<tr>
<td>VRA0004:</td>
<td>VRA lost IP</td>
</tr>
<tr>
<td>VRA0005:</td>
<td>VRAs not connected</td>
</tr>
<tr>
<td>VRA0006:</td>
<td>Datastore for journal disk is full</td>
</tr>
<tr>
<td>VRA0007:</td>
<td>I/O error to journal</td>
</tr>
<tr>
<td>VRA0008:</td>
<td>Recovery disk and VMs missing</td>
</tr>
<tr>
<td>VRA0009:</td>
<td>Recovery disk missing</td>
</tr>
<tr>
<td>VRA0010:</td>
<td>Recovery disks turned off</td>
</tr>
<tr>
<td>VRA0011:</td>
<td>Recovery disk inaccessible</td>
</tr>
<tr>
<td>VRA0012:</td>
<td>Cannot write to recovery disk</td>
</tr>
<tr>
<td>VRA0013:</td>
<td>I/O error to recovery disk</td>
</tr>
<tr>
<td>VRA0014:</td>
<td>Cloned disks turned off</td>
</tr>
<tr>
<td>VRA0015:</td>
<td>Cloned disk inaccessible</td>
</tr>
<tr>
<td>VRA0016:</td>
<td>Datastore for clone disk is full</td>
</tr>
<tr>
<td>VRA0017:</td>
<td>I/O error to clone</td>
</tr>
<tr>
<td>VRA0018:</td>
<td>Protected disk and VM missing</td>
</tr>
<tr>
<td>VRA0019:</td>
<td>Protected disk missing</td>
</tr>
<tr>
<td>VRA0020:</td>
<td>VM powered off</td>
</tr>
<tr>
<td>VRA0021:</td>
<td>VM disk inaccessible</td>
</tr>
<tr>
<td>VRA0022:</td>
<td>VM disk incompatible</td>
</tr>
<tr>
<td>VRA0023:</td>
<td>VRA cannot be registered.</td>
</tr>
<tr>
<td>VRA0024:</td>
<td>VRA removed</td>
</tr>
<tr>
<td>VRA0025:</td>
<td>I/O synchronization</td>
</tr>
<tr>
<td>VRA0026:</td>
<td>Recovery disk removed</td>
</tr>
<tr>
<td>VRA0027:</td>
<td>Journal disk removed</td>
</tr>
<tr>
<td>VRA0028:</td>
<td>VRA powered off</td>
</tr>
<tr>
<td>VRA0029:</td>
<td>VRA memory low</td>
</tr>
<tr>
<td>VRA0030:</td>
<td>Journal size mismatch</td>
</tr>
<tr>
<td>VRA0032:</td>
<td>VRA out-of-date</td>
</tr>
<tr>
<td>VRA0035:</td>
<td>VRA reconciliation</td>
</tr>
<tr>
<td>VRA0036:</td>
<td>For internal use only</td>
</tr>
<tr>
<td>VRA0037:</td>
<td>Local MAC Address Conflict</td>
</tr>
<tr>
<td>VRA0038:</td>
<td>MAC Address Conflict</td>
</tr>
<tr>
<td>VRA0039:</td>
<td>Journal reached configured limit</td>
</tr>
<tr>
<td>VRA0040:</td>
<td>Journal space low</td>
</tr>
<tr>
<td>VRA0049:</td>
<td>Host rollback failed</td>
</tr>
<tr>
<td>VRA0050:</td>
<td>Wrong host password</td>
</tr>
<tr>
<td>VRA0051:</td>
<td>For internal use only</td>
</tr>
<tr>
<td>VRA0052:</td>
<td>Disk visible but not recognized</td>
</tr>
<tr>
<td>VRA0053:</td>
<td>System disk removed</td>
</tr>
<tr>
<td>ZCC0001:</td>
<td>Zerto Cloud Connector removed</td>
</tr>
<tr>
<td>ZCC0002:</td>
<td>Zerto Cloud Connector powered off</td>
</tr>
<tr>
<td>ZCC0003:</td>
<td>Orphaned Zerto Cloud Connector</td>
</tr>
<tr>
<td>ZCM0001:</td>
<td>No connection to Zerto Virtual Manager</td>
</tr>
<tr>
<td>ZCM0002:</td>
<td>Zerto Cloud Manager not support</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
</tr>
<tr>
<td>ZVM0001</td>
<td>No connection to hypervisor manager, such as VMware vCenter Server and Microsoft SCVMM, or to public cloud.</td>
</tr>
<tr>
<td>ZVM0002</td>
<td>No connection to VRA</td>
</tr>
<tr>
<td>ZVM0003</td>
<td>No connection to site</td>
</tr>
<tr>
<td>ZVM0004</td>
<td>Peer site out-of-date</td>
</tr>
<tr>
<td>ZVM0005</td>
<td>Zerto Virtual Manager space low</td>
</tr>
<tr>
<td>ZVM0006</td>
<td>Upgrade available</td>
</tr>
<tr>
<td>ZVM0007</td>
<td>Cannot upgrade</td>
</tr>
<tr>
<td>ZVM0008</td>
<td>Version mismatch</td>
</tr>
<tr>
<td>ZVM0009</td>
<td>Internal error</td>
</tr>
<tr>
<td>ZVM0010</td>
<td>Synchronization between Zerto Virtual Managers</td>
</tr>
<tr>
<td>ZVM0011</td>
<td>Metadata Collection</td>
</tr>
<tr>
<td>ZVM0012</td>
<td>Metadata Collection</td>
</tr>
<tr>
<td>ZVM0013</td>
<td>Metadata Collection</td>
</tr>
<tr>
<td>ZVM0014</td>
<td>VRA/Diskbox SCSI GUID mismatch</td>
</tr>
<tr>
<td>ZVM0015</td>
<td>Hyper-V host state</td>
</tr>
<tr>
<td>FLR0001</td>
<td>Files cannot be restored</td>
</tr>
</tbody>
</table>

**IsDismissed**

- **True**: Returns the list of alerts that have been dismissed.
- **False**: Returns the list of alerts that have not been dismissed.

**Level**

- **Warning**: The alert is a warning.
- **Error**: The indicates an error.

**Link**

- **href**: The URL used.
- **identifier**: The unique internal identifier of the alert.
- **rel**: The next path level for the API relative to the current path.
- **type**: The API interface service.

**Site**

- **href**: The URL used.
- **identifier**: The unique internal identifier of the local site.
- **rel**: The next path level for the API relative to the current path.
- **type**: The API interface service.

**TurnedOn**

The date the alerts was issued. The value can be converted to an understandable date using code similar to the following:

```javascript
var date = new Date(jsonDate);
```

or code similar to the Perl code example, `jsonDateToString()`, in “Perl Code Example: Retrieving the First 100 Records”, on page 193.

<table>
<thead>
<tr>
<th>Entity Types</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zvm</td>
<td>Alerts that are triggered when there is a problem with a Zerto Virtual Manager.</td>
</tr>
<tr>
<td>Vra</td>
<td>Alerts that are triggered when there is a problem with a VRA.</td>
</tr>
<tr>
<td>Vpg</td>
<td>Alerts that are triggered when there is a problem with a VPG.</td>
</tr>
<tr>
<td>CloudConnector</td>
<td>Alerts that are triggered when there is a problem with a Zerto Cloud Connector.</td>
</tr>
<tr>
<td>Storage</td>
<td>Alerts that are triggered when there is a problem with a datastore specified in a VPG.</td>
</tr>
<tr>
<td>License</td>
<td>Alerts that are triggered when there is a problem with a Zerto Virtual Manager license.</td>
</tr>
<tr>
<td>Zcm</td>
<td>Alerts that are triggered when there is a problem with a Zerto Cloud Manager.</td>
</tr>
<tr>
<td>FileRecoveryComponent</td>
<td>Alerts that are triggered when there is a problem with file level recovery.</td>
</tr>
</tbody>
</table>


<table>
<thead>
<tr>
<th>Help Identifier Types</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS0001</td>
<td>AWS S3 bucket removed</td>
</tr>
<tr>
<td>BCK0001</td>
<td>Offsite backup fails</td>
</tr>
<tr>
<td>BCK0002</td>
<td>Offsite backup fails</td>
</tr>
<tr>
<td>BCK0005</td>
<td>Offsite backup repository disconnected.</td>
</tr>
<tr>
<td>BCK0006</td>
<td>Offsite backup repository disconnected.</td>
</tr>
<tr>
<td>BCK0007</td>
<td>Offsite backup repository not defined.</td>
</tr>
<tr>
<td>LIC0001</td>
<td>License exceeded</td>
</tr>
<tr>
<td>LIC0003</td>
<td>License about to expire</td>
</tr>
<tr>
<td>LIC0004</td>
<td>License expired and exceeded</td>
</tr>
<tr>
<td>LIC0006</td>
<td>License expired</td>
</tr>
<tr>
<td>LIC0007</td>
<td>License exceeded</td>
</tr>
<tr>
<td>LIC0009</td>
<td>Public cloud replication not supported</td>
</tr>
<tr>
<td>LIC0010</td>
<td>VCD not supported</td>
</tr>
<tr>
<td>LIC0011</td>
<td>Multi hypervisor replication not supported</td>
</tr>
<tr>
<td>STR0001</td>
<td>Datastore not accessible</td>
</tr>
<tr>
<td>STR0002</td>
<td>Datastore full</td>
</tr>
<tr>
<td>STR0004</td>
<td>Datastore low in space</td>
</tr>
<tr>
<td>VCD0001</td>
<td>vCenter Server for the Org vDC is not found</td>
</tr>
<tr>
<td>VCD0002</td>
<td>Org vDC is defined in multiple vCenter Servers</td>
</tr>
<tr>
<td>VCD0003</td>
<td>Org vDC storage profile not found in vCenter Server</td>
</tr>
<tr>
<td>VCD0004</td>
<td>Provider vDC storage profile not found in vCenter Server</td>
</tr>
<tr>
<td>VCD0005</td>
<td>Org vDC network not retrieved</td>
</tr>
<tr>
<td>VCD0006</td>
<td>Provider vDC metadata not found</td>
</tr>
<tr>
<td>VCD0007</td>
<td>Org vDC resource pool not retrieved</td>
</tr>
<tr>
<td>VCD0010</td>
<td>OrgNetwork not retrieved</td>
</tr>
<tr>
<td>VCD0014</td>
<td>vCD disconnection</td>
</tr>
<tr>
<td>VCD0015</td>
<td>AMQP-server disconnection</td>
</tr>
<tr>
<td>VCD0016</td>
<td>Provider vDC datastore not found</td>
</tr>
<tr>
<td>VCD0017</td>
<td>Metadata not accessible</td>
</tr>
<tr>
<td>VCD0018</td>
<td>Duplicated MAC addresses</td>
</tr>
<tr>
<td>VCD0020</td>
<td>VM inconsistency in vApp</td>
</tr>
<tr>
<td>VCD0021</td>
<td>VM inconsistency in vApp</td>
</tr>
<tr>
<td>RESPONSE: DESCRIPTION</td>
<td></td>
</tr>
<tr>
<td>-----------------------</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0003</strong>: VPG has low journal history</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0004</strong>: VPG has low journal history</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0005</strong>: VPG in error state</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0006</strong>: VPG missing configuration details</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0007</strong>: VPG replication paused</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0008</strong>: VPG rollback failed</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0009</strong>: VPG target RPO exceeded</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0010</strong>: VPG target RPO exceeded</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0011</strong>: VPG test overdue</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0012</strong>: VPG test overdue</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0014</strong>: VPG waiting for commit or rollback</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0015</strong>: Resources not enough to support VPG</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0016</strong>: Resources pool not found</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0017</strong>: VPG protection paused</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0018</strong>: VMs in VPG not configured with a storage profile</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0019</strong>: VPG recovery storage profile disabled</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0020</strong>: VPG recovery storage profile not found</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0021</strong>: VPG recovery storage profile not found</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0022</strong>: VPG recovery storage profile disabled</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0023</strong>: VPG recovery storage profile not found</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0024</strong>: VPG recovery storage profile does not include active datastores</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0025</strong>: vApp network mapping not defined</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0026</strong>: VPG recovery storage profile changed</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0027</strong>: VPG includes VMs that are no longer protected</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0028</strong>: Corrupted Org vDC network mapping</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0035</strong>: VPG protected resources not in ZORG</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0036</strong>: VPG recovery resources not in ZORG</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0037</strong>: Journal history is compromised</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0038</strong>: Journal history is compromised</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0039</strong>: RDM has an odd number of blocks</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0040</strong>: Virtual machine hardware mismatch with recovery site</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0041</strong>: Virtual machine running Windows 2003</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0042</strong>: Recovery network not found</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0043</strong>: SAN policy might cause some volumes to become offline upon recovery</td>
<td></td>
</tr>
<tr>
<td><strong>VPG0044</strong>: One or more of the virtual machines in the VPG has a disk with size 0</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RESPONSE: DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>VRA0001</strong>: Host without VRA</td>
</tr>
<tr>
<td><strong>VRA0002</strong>: VRA without IP</td>
</tr>
<tr>
<td><strong>VRA0003</strong>: Host IP changes</td>
</tr>
<tr>
<td><strong>VRA0004</strong>: VRA lost IP</td>
</tr>
<tr>
<td><strong>VRA0005</strong>: VRAs not connected</td>
</tr>
<tr>
<td><strong>VRA0006</strong>: Datastore for journal disk is full</td>
</tr>
<tr>
<td><strong>VRA0007</strong>: I/O error to journal</td>
</tr>
<tr>
<td><strong>VRA0008</strong>: Recovery disk and VMs missing</td>
</tr>
<tr>
<td><strong>VRA0009</strong>: Recovery disk missing</td>
</tr>
<tr>
<td>RESPONSE: DESCRIPTION</td>
</tr>
<tr>
<td>-------------------------</td>
</tr>
<tr>
<td>VRA0010: Recovery disks turned off</td>
</tr>
<tr>
<td>VRA0011: Recovery disk inaccessible</td>
</tr>
<tr>
<td>VRA0012: Cannot write to recovery disk</td>
</tr>
<tr>
<td>VRA0013: I/O error to recovery disk</td>
</tr>
<tr>
<td>VRA0014: Cloned disks turned off</td>
</tr>
<tr>
<td>VRA0015: Cloned disk inaccessible</td>
</tr>
<tr>
<td>VRA0016: Datastore for clone disk is full</td>
</tr>
<tr>
<td>VRA0017: I/O error to clone</td>
</tr>
<tr>
<td>VRA0018: Protected disk and VM missing</td>
</tr>
<tr>
<td>VRA0019: Protected disk missing</td>
</tr>
<tr>
<td>VRA0020: VM powered off</td>
</tr>
<tr>
<td>VRA0021: VM disk inaccessible</td>
</tr>
<tr>
<td>VRA0022: VM disk incompatible</td>
</tr>
<tr>
<td>VRA0023: VRA cannot be registered.</td>
</tr>
<tr>
<td>VRA0024: VRA removed</td>
</tr>
<tr>
<td>VRA0025: I/O synchronization</td>
</tr>
<tr>
<td>VRA0026: Recovery disk removed</td>
</tr>
<tr>
<td>VRA0027: Journal disk removed</td>
</tr>
<tr>
<td>VRA0028: VRA powered off</td>
</tr>
<tr>
<td>VRA0029: VRA memory low</td>
</tr>
<tr>
<td>VRA0030: Journal size mismatch</td>
</tr>
<tr>
<td>VRA0032: VRA out-of-date</td>
</tr>
<tr>
<td>VRA0035: VRA reconciliation</td>
</tr>
<tr>
<td>VRA0036: For internal use only</td>
</tr>
<tr>
<td>VRA0037: Local MAC Address Conflict</td>
</tr>
<tr>
<td>VRA0038: MAC Address Conflict</td>
</tr>
<tr>
<td>VRA0039: Journal reached configured limit</td>
</tr>
<tr>
<td>VRA0040: Journal space low</td>
</tr>
<tr>
<td>VRA0049: Host rollback failed</td>
</tr>
<tr>
<td>VRA0050: Wrong host password</td>
</tr>
<tr>
<td>VRA0051: For internal use only</td>
</tr>
<tr>
<td>VRA0052: Disk visible but not recognized</td>
</tr>
<tr>
<td>VRA0053: System disk removed</td>
</tr>
</tbody>
</table>
### RESPONSE: DESCRIPTION

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ZCC0001</td>
<td>Zerto Cloud Connector removed</td>
</tr>
<tr>
<td>ZCC0002</td>
<td>Zerto Cloud Connector powered off</td>
</tr>
<tr>
<td>ZCC0003</td>
<td>Orphaned Zerto Cloud Connector</td>
</tr>
<tr>
<td>ZCM0001</td>
<td>No connection to Zerto Virtual Manager</td>
</tr>
<tr>
<td>ZCM0002</td>
<td>Zerto Cloud Manager not support</td>
</tr>
<tr>
<td>ZVM0001</td>
<td>No connection to hypervisor manager, such as VMware vCenter Server and Microsoft SCVMM, or to public cloud.</td>
</tr>
<tr>
<td>ZVM0002</td>
<td>No connection to VRA</td>
</tr>
<tr>
<td>ZVM0003</td>
<td>No connection to site</td>
</tr>
<tr>
<td>ZVM0004</td>
<td>Peer site out-of-date</td>
</tr>
<tr>
<td>ZVM0005</td>
<td>Zerto Virtual Manager space low</td>
</tr>
<tr>
<td>ZVM0006</td>
<td>Upgrade available</td>
</tr>
<tr>
<td>ZVM0007</td>
<td>Cannot upgrade</td>
</tr>
<tr>
<td>ZVM0008</td>
<td>Version mismatch</td>
</tr>
<tr>
<td>ZVM0009</td>
<td>Internal error</td>
</tr>
<tr>
<td>ZVM0010</td>
<td>Synchronization between Zerto Virtual Managers</td>
</tr>
<tr>
<td>ZVM0011</td>
<td>Metadata Collection</td>
</tr>
<tr>
<td>ZVM0012</td>
<td>Metadata Collection</td>
</tr>
<tr>
<td>ZVM0013</td>
<td>Metadata Collection</td>
</tr>
<tr>
<td>ZVM0014</td>
<td>VRA/Diskbox SCSI GUID mismatch</td>
</tr>
<tr>
<td>ZVM0015</td>
<td>Hyper-V host state</td>
</tr>
<tr>
<td>FLR0001</td>
<td>Files cannot be restored</td>
</tr>
<tr>
<td>Unknown</td>
<td></td>
</tr>
</tbody>
</table>

Response values for `https://zvm_ip/port/v1/alerts/levels`.

### RESPONSE: DESCRIPTION

<table>
<thead>
<tr>
<th>Level type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Warning</td>
<td>The alert is a warning.</td>
</tr>
<tr>
<td>Error</td>
<td>The indicates an error.</td>
</tr>
</tbody>
</table>

### Alerts: POST

Dismisses or undismisses a specified alert.

**URL**

- **Dismiss alert**  :  `https://zvm_ip/port/v1/alerts/{alertId}/dismiss`
- **Undismiss alert**:  `https://zvm_ip/port/v1/alerts/{alertId}/undismiss`

Where:

- **zvm_ip**  :  The IP address of the Zerto Virtual Manager where the API is run.
- **port**  :  The port to access the Zerto Virtual Manager. The default port is 9669.
- **alertId**  :  The identifier of the alert for which information is returned or which is dismissed or undismissed.

**Request Format**

The request body is empty.

**Response Format**

The response body is empty.
XML Response Format

For the XML response format, see “Alerts API XML Response Format”, on page 233.

Code Examples

For complete code examples, see “Code Samples”, on page 22.

/v1/alerts cURL Code Example

Retrieve the list of all alerts.

```bash
curl -D responseHeader -H "Content-Type: application/json" -H "x-zerto-session: 9UDQD6RG7YF3QJLWQXGJV8C453N277NA22P7FSNWVZCJTWCBRHQ" 
https://127.0.0.1:9669/v1/alerts
```

Dismiss a specific alert.

```bash
curl -D responseHeader -H "Content-Type: application/json" -H "x-zerto-session: 9UDQD6RG7YF3QJLWQXGJV8C453N277NA22P7FSNWVZCJTWCBRHQ" 
https://127.0.0.1:9669/v1/alerts/d00ea07a-9317-4ab6-a609-0a5fcb04680/dismiss -d ""
```

This cURL command does not return any response. The -d "" is required so that cURL uses the POST method with this API.

For more code examples, see “cURL Code”, on page 22.

Events API

/v1/events returns the last 1000 Zerto Virtual Replication events, with summary details about each event. You can use this API to see the results of operations such as cloning a VPG or testing a VPG.

For example, you can use this API to identify when a failover test was started and stopped, by looking for the FailOverTest and StopFailOverTest events and then using the timestamps for these events to see how long the VPG was tested.

URL

<table>
<thead>
<tr>
<th>All events</th>
<th>https://zvm_ip:port/v1/events</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filtered events</td>
<td>https://zvm_ip:port/v1/events?startDate={STARTDATE}&amp;endDate={ENDDATE}&amp;vpg={VPG}&amp;vpgIdentifier={VPGIDENTIFIER}&amp;eventType={EVENTTYPE}&amp;siteName={SITENAME}&amp;siteIdentifier={SITEIDENTIFIER}&amp;zorgIdentifier={ZORGIDENTIFIER}&amp;entityType={ENTITYTYPE}&amp;userName={USERNAME}&amp;category={CATEGORY}&amp;alertIdentifier={ALERTIDENTIFIER}</td>
</tr>
<tr>
<td>Single event</td>
<td>https://zvm_ip:port/v1/events/{eventId}</td>
</tr>
<tr>
<td>Valid event categories</td>
<td>https://zvm_ip:port/v1/events/categories</td>
</tr>
<tr>
<td>Valid event entities</td>
<td>https://zvm_ip:port/v1/events/entities</td>
</tr>
<tr>
<td>Valid event types</td>
<td>https://zvm_ip:port/v1/events/types</td>
</tr>
</tbody>
</table>

Where:

<table>
<thead>
<tr>
<th>zvm_ip</th>
<th>The IP address of the Zerto Virtual Manager where the API is run.</th>
</tr>
</thead>
<tbody>
<tr>
<td>port</td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
<tr>
<td>eventId</td>
<td>The identifier of the event for which information is returned.</td>
</tr>
<tr>
<td>Filters</td>
<td>Filters are optional and any combination of filters is valid. When no filter is specified, all events are returned. Filters are not case-sensitive.</td>
</tr>
<tr>
<td>---------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Filter</td>
<td>Description</td>
</tr>
<tr>
<td><code>startDate</code></td>
<td>The starting date for the list of events, supplied as a date with the format of the Zerto Virtual Manager where the API runs, for example, <code>yyyy-mm-dd</code>. You can also specify a local time with the following format: <code>yyyy-mm-ddThh:mm:ss</code>. Adding Z to the end of the time sets the time to UTC.</td>
</tr>
<tr>
<td><code>endDate</code></td>
<td>The end date for the list, supplied as a date with the format of the Zerto Virtual Manager where the API runs, for example, <code>yyyy-mm-dd</code>. You can also specify a local time with the following format: <code>yyyy-mm-ddThh:mm:ss</code>. Adding Z to the end of the time sets the time to UTC.</td>
</tr>
<tr>
<td><code>vpg, vpgIdentifier</code></td>
<td>The identifier of the VPG for which you want to return events.</td>
</tr>
<tr>
<td><code>eventType</code></td>
<td>The type of event. For the description of events, refer to the Zerto Virtual Replication documentation about alerts and events. Possible values are:</td>
</tr>
<tr>
<td>0</td>
<td>or Unknown</td>
</tr>
<tr>
<td>1</td>
<td>or CreateProtectionGroup</td>
</tr>
<tr>
<td>2</td>
<td>or RemoveProtectionGroup</td>
</tr>
<tr>
<td>3</td>
<td>or FailOver</td>
</tr>
<tr>
<td>4</td>
<td>or FailOverTest</td>
</tr>
<tr>
<td>5</td>
<td>or StopFailOverTest</td>
</tr>
<tr>
<td>6</td>
<td>or Move</td>
</tr>
<tr>
<td>7</td>
<td>or ProtectVM</td>
</tr>
<tr>
<td>8</td>
<td>or UnprotectVM</td>
</tr>
<tr>
<td>9</td>
<td>or InstallVra</td>
</tr>
<tr>
<td>10</td>
<td>or UninstallVra</td>
</tr>
<tr>
<td>11</td>
<td>or UpdateProtectionGroup</td>
</tr>
<tr>
<td>12</td>
<td>or InsertTaggedCP</td>
</tr>
<tr>
<td>13</td>
<td>or HandleMirrorPromotion</td>
</tr>
<tr>
<td>14</td>
<td>or ActivateAllMirrors</td>
</tr>
<tr>
<td>15</td>
<td>or LogCollection</td>
</tr>
<tr>
<td>16</td>
<td>or ForceReconfigurationOfNewVM</td>
</tr>
<tr>
<td>17</td>
<td>or ClearSite</td>
</tr>
<tr>
<td>18</td>
<td>or ForceRemoveProtectionGroup</td>
</tr>
<tr>
<td>19</td>
<td>or ForceUpdateProtectionGroup</td>
</tr>
<tr>
<td>20</td>
<td>or ForceKillProtectionGroup</td>
</tr>
<tr>
<td>21</td>
<td>or PrePostScript</td>
</tr>
<tr>
<td>22</td>
<td>or InitFullSync</td>
</tr>
<tr>
<td>23</td>
<td>or Pair</td>
</tr>
<tr>
<td>24</td>
<td>or Unpair</td>
</tr>
<tr>
<td>25</td>
<td>or InstallCloudConnector</td>
</tr>
<tr>
<td>26</td>
<td>or UninstallCloudConnector</td>
</tr>
<tr>
<td>27</td>
<td>or RedeployCloudConnector</td>
</tr>
<tr>
<td>28</td>
<td>or ScriptExecutionFailure</td>
</tr>
<tr>
<td>29</td>
<td>or SetAdvancedSiteSettings</td>
</tr>
<tr>
<td>30</td>
<td>or Clone</td>
</tr>
<tr>
<td>31</td>
<td>or KeepDisk</td>
</tr>
<tr>
<td>32</td>
<td>or FailoverBeforeCommit</td>
</tr>
<tr>
<td>33</td>
<td>or FailoverCommit</td>
</tr>
<tr>
<td>34</td>
<td>or FailoverRollback</td>
</tr>
<tr>
<td>35</td>
<td>or MoveBeforeCommit</td>
</tr>
<tr>
<td>36</td>
<td>or MoveRollback</td>
</tr>
<tr>
<td>37</td>
<td>or MoveCommit</td>
</tr>
<tr>
<td>38</td>
<td>or MaintainHost</td>
</tr>
<tr>
<td>39</td>
<td>or UpgradeVra</td>
</tr>
<tr>
<td>Event Code</td>
<td>Event Description</td>
</tr>
<tr>
<td>------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>40</td>
<td>MoveProtectionGroupToManualOperationNeeded</td>
</tr>
<tr>
<td>42</td>
<td>PauseProtectionGroup</td>
</tr>
<tr>
<td>43</td>
<td>ResumeProtectionGroup</td>
</tr>
<tr>
<td>44</td>
<td>UpgradeZVM</td>
</tr>
<tr>
<td>45</td>
<td>BulkUpgradeVras</td>
</tr>
<tr>
<td>46</td>
<td>BulkUninstallVras</td>
</tr>
<tr>
<td>47</td>
<td>AlertTurnedOn</td>
</tr>
<tr>
<td>48</td>
<td>AlertTurnedOff</td>
</tr>
<tr>
<td>49</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>ChangeRecoveryHost</td>
</tr>
<tr>
<td>51</td>
<td>BackupProtectionGroup</td>
</tr>
<tr>
<td>52</td>
<td>CleanupProtectionGroupVipDiskbox</td>
</tr>
<tr>
<td>53</td>
<td>RestoreProtectionGroup</td>
</tr>
<tr>
<td>54</td>
<td>PreScript</td>
</tr>
<tr>
<td>55</td>
<td>PostScript</td>
</tr>
<tr>
<td>56</td>
<td>RemoveVmFromVc</td>
</tr>
<tr>
<td>57</td>
<td>ChangeVraPasswordIpSettings</td>
</tr>
<tr>
<td>58</td>
<td>FlrJournalMount</td>
</tr>
<tr>
<td>59</td>
<td>FlrJournalUnmount</td>
</tr>
<tr>
<td>60</td>
<td>Login</td>
</tr>
<tr>
<td>61</td>
<td>StartVMsWithOrder</td>
</tr>
<tr>
<td>62</td>
<td>HostEnteringMaintenanceMode</td>
</tr>
<tr>
<td>63</td>
<td>HostExitingMaintenanceMode</td>
</tr>
<tr>
<td>64</td>
<td>VmRestoredToSnapshot</td>
</tr>
<tr>
<td>65</td>
<td>ProtectedVmRemovedFromHypervisor</td>
</tr>
<tr>
<td>66</td>
<td>ProtectedVmAddedToHypervisor</td>
</tr>
<tr>
<td>67</td>
<td>PauseProtectionGroupForMissingVm</td>
</tr>
<tr>
<td>68</td>
<td>ResumeProtectionGroupAfterUserRemovedMissingVm</td>
</tr>
<tr>
<td>69</td>
<td>ResumeProtectionGroupAfterVmReadded</td>
</tr>
</tbody>
</table>

- **siteName**: The name of the site.
- **siteIdentifier**: The internal site identifier.
- **zorgIdentifier**: The identifier of the ZORG, Zerto organization, defined in the Zerto Cloud Manager.
- **entityType**: The type of entity to return. Possible values are:
  - 0 or **VPG**: The entity is a VPG.
  - 1 or **VRA**: The entity is a VRA.
  - 2 or **Unknown**: The entity is unknown.
  - 3 or **Site**: The entity is the site.
- **userName**: The name of the user for which the event occurred. If the event occurred as a result of a task started by the Zerto Virtual Manager, for example, when moving a VPG before the commit stage, the user is *System*.
- **category**: The type of event to return. This filter behaves in the same way as the `eventCategory` filter. Possible values are:
  - 0 or **All**: All event types.
  - 1 or **Events**: The event is not an alert event (not EV0056 nor EV0057).
  - 2 or **Alerts**: The event is an alert event (EV0056 and EV0057).
Events API

HTTP Method

GET

Security

The API is exposed over HTTPS. Client code must use the x-zerto-session HTTP authorization header.

See Also

Starting a session: Session: POST

Format

Json, XML

Request Format

The request body is empty.

Json Response Format

The following is an example response Json body for https://zvm_ip:port/v1/events and for https://zvm_ip:port/v1/events/{eventId}.

```json
[
  {
    "Description": "String content",
    "EntityType": 0,
    "EventCategory": "String content",
    "EventCompletedSuccessfully": true,
    "EventIdentifier": "String content",
    "EventType": 0,
    "HelpLink": "String content",
    "Link": {
      "href": "String content",
      "identifier": "String content",
      "rel": "String content",
      "type": "String content"
    },
    "Link__x007B_0_x007D_": {
      "href": "String content",
      "rel": "String content",
      "type": "String content"
    },
    "OccurredOn": "\Date(928142400000+0300)\",
  }
]```

```
"RelatedEntities": {
  "Alerts": [{
    "href": "String content",
    "ma": "String content",
    "rel": "String content",
    "type": "String content"
  }],
  "FlrSessions": [{
    "href": "String content",
    "identifier": "String content",
    "rel": "String content",
    "type": "String content"
  }],
  "Hosts": [{
    "href": "String content",
    "identifier": "String content",
    "rel": "String content",
    "type": "String content"
  }],
  "Sites": [{
    "href": "String content",
    "identifier": "String content",
    "rel": "String content",
    "type": "String content"
  }],
  "Vpgs": [{
    "href": "String content",
    "identifier": "String content",
    "rel": "String content",
    "type": "String content"
  }],
  "Zorgs": [{
    "href": "String content",
    "identifier": "String content",
    "rel": "String content",
    "type": "String content"
  }]
},

"SiteIdentifier": "String content",
"SiteName": "String content",
"UserName": "String content",
"Vpgs": [{
  "Link": { "href": "String content",
    "rel": "String content",
    "type": "String content"
  },
  "VpgIdentifier": "String content",
  "VpgName": "String content"
}],
"ZorgIdentifier": "String content",
"ZorgName": "String content"
}
```


```
["String content"]
```
XML Response Format

For the XML response format, see “Events API XML Response Format”, on page 234.

Response Values

Response values for `https://zvm_ip:port/v1/events` and for `https://zvm_ip:port/v1/events/{eventId}`.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>A description of the event.</td>
</tr>
<tr>
<td><strong>EntityType</strong></td>
<td>The type of entity. Possible values are (Json/XML):</td>
</tr>
<tr>
<td>0/VPG: The entity is a VPG.</td>
<td></td>
</tr>
<tr>
<td>1/VRA: The entity is a VRA.</td>
<td></td>
</tr>
<tr>
<td>2/Unknown: The entity is unknown.</td>
<td></td>
</tr>
<tr>
<td>3/Site: The entity is the site.</td>
<td></td>
</tr>
<tr>
<td><strong>EventCategory</strong></td>
<td>The type of event. Possible values are:</td>
</tr>
<tr>
<td>0/All: All event types.</td>
<td></td>
</tr>
<tr>
<td>1/Events: The event is not an alert event (not EV0056 nor EV0057).</td>
<td></td>
</tr>
<tr>
<td>2/Alerts: The event is an alert event (EV0056 and EV0057).</td>
<td></td>
</tr>
<tr>
<td><strong>EventCompletedSuccessfully</strong></td>
<td>Whether the event completed successfully or not.</td>
</tr>
<tr>
<td><strong>EventIdentifier</strong></td>
<td>The unique internal identifier of the event.</td>
</tr>
<tr>
<td><strong>EventType</strong></td>
<td>The type of event. Possible values are (Json/XML):</td>
</tr>
<tr>
<td>1/Unknown</td>
<td></td>
</tr>
<tr>
<td>2/CreateProtectionGroup</td>
<td></td>
</tr>
<tr>
<td>3/RemoveProtectionGroup</td>
<td></td>
</tr>
<tr>
<td>4/FailOver</td>
<td></td>
</tr>
<tr>
<td>5/FailOverTest</td>
<td></td>
</tr>
<tr>
<td>6/StopFailOverTest</td>
<td></td>
</tr>
<tr>
<td>7/Move</td>
<td></td>
</tr>
<tr>
<td>8/ProtectVM</td>
<td></td>
</tr>
<tr>
<td>9/UnprotectVM</td>
<td></td>
</tr>
<tr>
<td>10/InstallVra</td>
<td></td>
</tr>
<tr>
<td>11/UninstallVra</td>
<td></td>
</tr>
<tr>
<td>12/UpdateProtectionGroup</td>
<td></td>
</tr>
<tr>
<td>13/InsertTaggedCP</td>
<td></td>
</tr>
<tr>
<td>14/HandleMirrorPromotion</td>
<td></td>
</tr>
<tr>
<td>15/ActivateAllMirrors</td>
<td></td>
</tr>
<tr>
<td>16/LogCollection</td>
<td></td>
</tr>
<tr>
<td>17/ForceReconfigurationOfNewVM</td>
<td></td>
</tr>
<tr>
<td>18/ClearSite</td>
<td></td>
</tr>
<tr>
<td>19/ForceRemoveProtectionGroup</td>
<td></td>
</tr>
<tr>
<td>20/ForceUpdateProtectionGroup</td>
<td></td>
</tr>
<tr>
<td>21/ForceKillProtectionGroup</td>
<td></td>
</tr>
<tr>
<td>22/PrePostScript</td>
<td></td>
</tr>
<tr>
<td>23/InitFullSync</td>
<td></td>
</tr>
<tr>
<td>24/Pair</td>
<td></td>
</tr>
<tr>
<td>25/Unpair</td>
<td></td>
</tr>
<tr>
<td>26/InstallCloudConnector</td>
<td></td>
</tr>
<tr>
<td>27/UninstallCloudConnector</td>
<td></td>
</tr>
<tr>
<td>28/RedeployCloudConnector</td>
<td></td>
</tr>
<tr>
<td>29/ScriptExecutionFailure</td>
<td></td>
</tr>
<tr>
<td>30/SetAdvancedSiteSettings</td>
<td></td>
</tr>
</tbody>
</table>

Events API
<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>31/Clone</td>
<td></td>
</tr>
<tr>
<td>32/KeepDisk</td>
<td></td>
</tr>
<tr>
<td>33/FailoverBeforeCommit</td>
<td></td>
</tr>
<tr>
<td>34/FailoverCommit</td>
<td></td>
</tr>
<tr>
<td>35/FailoverRollback</td>
<td></td>
</tr>
<tr>
<td>36/MoveBeforeCommit</td>
<td></td>
</tr>
<tr>
<td>37/MoveCommit</td>
<td></td>
</tr>
<tr>
<td>38/MoveRollback</td>
<td></td>
</tr>
<tr>
<td>39/MaintainHost</td>
<td></td>
</tr>
<tr>
<td>40/UpgradeVra</td>
<td></td>
</tr>
<tr>
<td>41/MoveProtectionGroupToManualOperationNeeded</td>
<td></td>
</tr>
<tr>
<td>42/ChangeVralpSettings</td>
<td></td>
</tr>
<tr>
<td>43/PauseProtectionGroup</td>
<td></td>
</tr>
<tr>
<td>44/ResumeProtectionGroup</td>
<td></td>
</tr>
<tr>
<td>45/UpgradeZVM</td>
<td></td>
</tr>
<tr>
<td>46/BulkUpgradeVras</td>
<td></td>
</tr>
<tr>
<td>47/BulkUninstallVras</td>
<td></td>
</tr>
<tr>
<td>48/AlertTurnedOn</td>
<td></td>
</tr>
<tr>
<td>49/AlertTurnedOff</td>
<td></td>
</tr>
<tr>
<td>50/ChangeVraPassword</td>
<td></td>
</tr>
<tr>
<td>51/ChangeRecoveryHost</td>
<td></td>
</tr>
<tr>
<td>52/BackupProtectionGroup</td>
<td></td>
</tr>
<tr>
<td>53/CleanupProtectionGroupVipDiskbox</td>
<td></td>
</tr>
<tr>
<td>54/RestoreProtectionGroup</td>
<td></td>
</tr>
<tr>
<td>55/PreScript</td>
<td></td>
</tr>
<tr>
<td>56/PostScript</td>
<td></td>
</tr>
<tr>
<td>57/RemoveVmFromVc</td>
<td></td>
</tr>
<tr>
<td>58/ChangeVraPasswordIpSettings</td>
<td></td>
</tr>
<tr>
<td>59/FlrJournalMount</td>
<td></td>
</tr>
<tr>
<td>60/FlrJournalUnmount</td>
<td></td>
</tr>
<tr>
<td>61/Login</td>
<td></td>
</tr>
<tr>
<td>62/HostEnteringMaintenanceMode</td>
<td></td>
</tr>
<tr>
<td>63/HostExitingMaintenanceMode</td>
<td></td>
</tr>
<tr>
<td>64/VmRestoredToSnapshot</td>
<td></td>
</tr>
<tr>
<td>65/ProtectedVmRemovedFromHypervisor</td>
<td></td>
</tr>
<tr>
<td>66/ProtectedVmAddedToHypervisor,</td>
<td></td>
</tr>
<tr>
<td>67/PauseProtectionGroupForMissingVm</td>
<td></td>
</tr>
<tr>
<td>68/ResumeProtectionGroupAfterUserRemovedMissingVm</td>
<td></td>
</tr>
<tr>
<td>69/ResumeProtectionGroupAfterVmReadded</td>
<td></td>
</tr>
</tbody>
</table>

**HelpLink**

The link to additional information about the event.

**Link**

The link details.

**href**

The URL used.

**identifier**

The unique internal identifier of the event.

**rel**

The next path level for the API relative to the current path.

**type**

The API interface service.

**OccurredOn**

The date the event occurred. The value can be converted to an understandable date using code similar to the following:

```javascript
var date = new Date(jsonDate);
```

or code similar to the Perl code example, `jsonDateToString($)` in “Perl Code Example: Retrieving the First 100 Records”, on page 193.
<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>RelatedEntities</td>
<td>Entities possibly affected by the event.</td>
</tr>
<tr>
<td>Alerts</td>
<td>The alerts affected by the event.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used to retrieve alert information.</td>
</tr>
<tr>
<td>identifier</td>
<td>The unique internal identifier of the alert.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level for the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>FlrSessions</td>
<td>The file level restore sessions affected by the event.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used to retrieve file level restore session information.</td>
</tr>
<tr>
<td>identifier</td>
<td>The unique internal identifier of the file level restore session.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level for the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>Hosts</td>
<td>The hosts affected by the event.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used to retrieve host information.</td>
</tr>
<tr>
<td>identifier</td>
<td>The unique internal identifier of the host.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level for the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>Sites</td>
<td>The sites affected by the event.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used to retrieve site information.</td>
</tr>
<tr>
<td>identifier</td>
<td>The unique internal identifier of the site where the API runs.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level for the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>Vpgs</td>
<td>The VPGs affected by the event.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used to retrieve VPG information for each VPG affected by the event.</td>
</tr>
<tr>
<td>identifier</td>
<td>The unique internal identifier of the VPG.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level for the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>Zorgs</td>
<td>The ZORGS affected by the event.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used to retrieve ZORG information.</td>
</tr>
<tr>
<td>identifier</td>
<td>The unique internal identifier of the ZORG.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level for the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>SiteIdentifier</td>
<td>The internal site identifier.</td>
</tr>
<tr>
<td>SiteName</td>
<td>The name of the site where the event occurred.</td>
</tr>
<tr>
<td>UserName</td>
<td>The username responsible for the event, system or an administrator user.</td>
</tr>
<tr>
<td>Vpgs</td>
<td>The VPGs affected by the event.</td>
</tr>
<tr>
<td>Link</td>
<td>The link to the specific VPG.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used to retrieve VPG information for each VPG affected by the event.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level for the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>VpgIdentifier</td>
<td>The internal VPG identifier.</td>
</tr>
<tr>
<td>VpgName</td>
<td>The name of the VPG.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ZorgIdentifier</td>
<td>The identifier of the ZORG, Zerto organization, defined in the Zerto Cloud Manager.</td>
</tr>
<tr>
<td>ZorgName</td>
<td>The name of the Zerto organization, defined by a Zerto Cloud Manager, where the event occurred.</td>
</tr>
</tbody>
</table>

Response values for https://zvm_ip:port/v1/events/categories.

**RESPONSE: DESCRIPTION**

Event categories:

- **All**: All event types.
- **Events**: The event is not an alert event (not EV0056 nor EV0057).
- **Alerts**: The event is an alert event (EV0056 and EV0057).

Response values for https://zvm_ip:port/v1/events/entities.

**RESPONSE: DESCRIPTION**

Entity types:

- **VPG**: The entity is a VPG.
- **VRA**: The entity is a VRA.
- **Unknown**: The entity is unknown.
- **Site**: The entity is the site.

Response values for https://zvm_ip:port/v1/events/types.

**RESPONSE: DESCRIPTION**

Event types:

- **Unknown**
- **CreateProtectionGroup**
- **RemoveProtectionGroup**
- **FailOver**
- **FailOverTest**
- **StopFailOverTest**
- **Move**
- **ProtectVM**
- **UnprotectVM**
- **InstallVra**
- **UninstallVra**
- **UpdateProtectionGroup**
- **InsertTaggedCP**
- **HandleMirrorPromotion**
- **ActivateAllMirrors**
- **LogCollection**
- **ForceReconfigurationOfNewVM**
- **ClearSite**
- **ForceRemoveProtectionGroup**
- **ForceUpdateProtectionGroup**
- **ForceKillProtectionGroup**
### Code Examples

For complete code examples, see "Code Samples", on page 22.
/v1/events cURL Code Example

Retrieve the list of all events.

```
curl -D responseHeader -H "Content-Type: application/json" -H "Accept: application/json" -H "x-zerto-session: 9UDQD6RG7YF33QJLwXGJv8C453N277NA22P7FSNWVZCJTWCBRHQ"
https://127.0.0.1:9669/v1/events
```

/v1/events with a filter cURL Code Example

Retrieve the list of events generated when VPGs are created.

```
curl -D responseHeader -H "Content-Type: application/json" -H "Accept: application/json" -H "x-zerto-session: 9UDQD6RG7YF33QJLwXGJv8C453N277NA22P7FSNWVZCJTWCBRHQ"
https://127.0.0.1:9669/v1/events?eventType=CreateProtectionGroup
```

For more code examples, see “cURL Code”, on page 22.

Local Site API

/v1/localsite returns information about the local site processing the API.

URL

- Local site information: https://zvm_ip:port/v1/localsite
- Valid pairing statuses: https://zvm_ip:port/v1/localsite/pairingstatuses

Where:

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>zvm_ip</td>
<td>The IP address of the Zerto Virtual Manager where the API is run.</td>
</tr>
<tr>
<td>port</td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
</tbody>
</table>

HTTP Method

GET

Security

The API is exposed over HTTPS. Client code must use the x-zerto-session HTTP authorization header.

See Also

- Starting a session: Session: POST
- Return the details of the peer sites: Peer Sites API
- Return the list of all hypervisor sites: Virtualization Sites API

Format

Json, XML
Local Site API

Request Format

The request body is empty.

Json Response Format

The following is an example response Json body for `https://zvm_ip:port/v1/localsite`.

```json
{
  "ContactEmail": "String content",
  "ContactName": "String content",
  "ContactPhone": "String content",
  "IsReplicationToSelfEnabled": Boolean,
  "Link": {
    "href": "String content",
    "identifier": "String content",
    "rel": "String content",
    "type": "String content"
  },
  "Location": "String content",
  "SiteIdentifier": "String content",
  "SiteName": "String content",
  "UtcOffsetInMinutes":2147483647,
  "Version": "String content"
}
```

The following is an example response Json body for `https://zvm_ip:port/v1/localsite/pairingstatuses`.

```json
["String content"]
```

XML Response Format

For the XML response format, see “Local Site API XML Response Format”, on page 237.

Response Values


<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ContactEmail</td>
<td>The email address defined in the Zerto User Interface Site Information dialog.</td>
</tr>
<tr>
<td>ContactName</td>
<td>The name of the contact person defined in the Zerto User Interface Site Information dialog.</td>
</tr>
<tr>
<td>ContactPhone</td>
<td>The phone number of the contact defined in the Zerto User Interface Site Information dialog.</td>
</tr>
<tr>
<td>IsReplicationToSelfEnabled</td>
<td>True: Protected virtual machines can be recovered to this protected site. False: Protected virtual machines can only be recovered to a different recovery site.</td>
</tr>
<tr>
<td>Link</td>
<td>The link details.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used.</td>
</tr>
<tr>
<td>identifier</td>
<td>The internal site identifier.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level for the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>Location</td>
<td>The location of the site defined during installation or in the Site Information dialog.</td>
</tr>
<tr>
<td>SiteIdentifier</td>
<td>The internal site identifier.</td>
</tr>
<tr>
<td>SiteName</td>
<td>The name of the site defined during installation or in the Site Information dialog.</td>
</tr>
<tr>
<td>UtcOffsetInMinutes</td>
<td>The offset of the site time from UTC in minutes.</td>
</tr>
<tr>
<td>Version</td>
<td>The Zerto Virtual Manager version.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Response: Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pairing statuses:</td>
</tr>
<tr>
<td>Paired: The site is paired.</td>
</tr>
<tr>
<td>Pairing: The site is in the process of being paired.</td>
</tr>
<tr>
<td>Unpaired: The site is not paired.</td>
</tr>
</tbody>
</table>

**Code Examples**

For complete code examples, see “Code Samples”, on page 22.

/v1/localsite cURL Code Example

Retrieve information about the local site where the API is run.

```
```

For more code examples, see “cURL Code”, on page 22.

**Peer Sites API**

/v1/peersites returns a list of all peer sites of the site processing the API and pairs the site processing the API with another site. The following API are available:

- “Peersites - GET”, below
- “Peersites - POST”, on page 65

<table>
<thead>
<tr>
<th>PURPOSE</th>
<th>METHOD</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>List of peer sites</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/peersites</td>
</tr>
<tr>
<td>A single peer site</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/peersites/{siteidentifier}</td>
</tr>
<tr>
<td>The status of peer sites</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/peersites/pairingstatuses</td>
</tr>
<tr>
<td>Pair to another sites</td>
<td>POST</td>
<td>https://zvm_ip:port/v1/peersites/pair</td>
</tr>
<tr>
<td>Unpair a site</td>
<td>DELETE</td>
<td>https://zvm_ip:port/v1/peersites/{SiteIdentifier}</td>
</tr>
</tbody>
</table>

**HTTP Method**

GET, POST, DELETE

**Security**

The API is exposed over HTTPS. Client code must use the x-zerto-session HTTP authorization header.

**See Also**

Starting a session: Session: POST
Return the details of the local site: Local Site API
Return the list of all hypervisor sites: Virtualization Sites API
Peersites API

**Peersites - GET**

Returns information about peer sites.

**URL**

<table>
<thead>
<tr>
<th>Type</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>All peer sites</td>
<td>https://zvm_ip:port/v1/peersites</td>
</tr>
<tr>
<td>Filtered peer sites</td>
<td>https://zvm_ip:port/v1/peersites?peerName={PEERNAME}&amp;pairingStatus={PAIRINGSTATUS}&amp;location={LOCATION}&amp;hostName={HOSTNAME}&amp;port={PORT}</td>
</tr>
<tr>
<td>Single peer site</td>
<td>https://zvm_ip:port/v1/peersites/{siteIdentifier}</td>
</tr>
<tr>
<td>Valid pairing statuses</td>
<td>https://zvm_ip:port/v1/peersites/pairingstatuses</td>
</tr>
</tbody>
</table>

Where:

- **zvm_ip**: The IP address of the Zerto Virtual Manager where the API is run.
- **port**: The port to access the Zerto Virtual Manager. The default port is 9669.
- **siteIdentifier**: The identifier of the peer site for which information is to be returned.
- **Filters**: Filters are optional and any combination of filters is valid. When no filter is specified, all peer sites are returned. Filters are not case-sensitive.

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>peerName</td>
<td>The name of a peer site for which information is to be returned. The name is case-sensitive.</td>
</tr>
<tr>
<td>pairingStatus</td>
<td>The pairing status for which information is to be returned.</td>
</tr>
<tr>
<td>location</td>
<td>The site location, as specified in the site information, for which information is to be returned.</td>
</tr>
<tr>
<td>hostName</td>
<td>The IP address of a Zerto Virtual Manager, paired with this site, for which information is to be returned.</td>
</tr>
<tr>
<td>port</td>
<td>The port used to access peer sites for which information is to be returned. The default port is 9081.</td>
</tr>
</tbody>
</table>
Json Response Format

The following is an example response Json body for https://zvm_ip:port/v1/peersites and for https://zvm_ip:port/v1/peersites/{siteIdentifier}.

```json
[{
    "HostName": "String content",
    "IncomingThroughputInMb":1.26743233E+15,
    "Link": {
        "href": "String content",
        "identifier": "String content",
        "rel": "String content",
        "type": "String content"
    },
    "Location": "String content",
    "OutgoingBandWidth":1.26743233E+15,
    "PairingStatus":0,
    "PeerSiteName": "String content",
    "Port":9081,
    "ProvisionedStorage":2147483647,
    "SiteIdentifier": "String content",
    "UsedStorage":2147483647,
    "Version": "String content"
}]
```

The following is an example response Json body for https://zvm_ip:port/v1/peersites/pairingstatuses.

```json
["String content"]
```

XML Response Format

For the XML response format, see “Peer Sites API GET Method Request and Response Formats”, on page 238.

Response Values


<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>HostName</td>
<td>The address of a machine where a peer site Zerto Virtual Manager runs.</td>
</tr>
<tr>
<td>IncomingThroughputInMb</td>
<td>The Mb/s for all the applications running on the virtual machines being recovered on the peer site.</td>
</tr>
<tr>
<td>Link</td>
<td>The link details.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used.</td>
</tr>
<tr>
<td>identifier</td>
<td>The internal site identifier.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level for the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>Link__x007B_0_x007D_</td>
<td>The link details.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level for the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>Location</td>
<td>The site location of the peer site defined during the installation or in the Site Information dialog.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>OutgoingBandWidth</td>
<td>The bandwidth throttling defined for the site.</td>
</tr>
<tr>
<td>PairingStatus</td>
<td>The connection status of the local site. Possible values are (Json/XML):</td>
</tr>
<tr>
<td></td>
<td><strong>Paired</strong>: The site is paired.</td>
</tr>
<tr>
<td></td>
<td><strong>Pairing</strong>: The site is in the process of being paired.</td>
</tr>
<tr>
<td></td>
<td><strong>Unpaired</strong>: The site is not paired.</td>
</tr>
<tr>
<td>PeerSiteName</td>
<td>The name of the peer site defined during installation or in the Site Information dialog.</td>
</tr>
<tr>
<td>Port</td>
<td>The port used for communication by the Zerto Virtual Managers. The default port is 9081.</td>
</tr>
<tr>
<td>ProvisionedStorage</td>
<td>The storage provisioned for all of the virtual machines in all the VPGs recovered to this site.</td>
</tr>
<tr>
<td>SiteIdentifier</td>
<td>The internal site identifier.</td>
</tr>
<tr>
<td>UsedStorage</td>
<td>The storage used by all of the virtual machines in all the VPGs recovered to this site.</td>
</tr>
<tr>
<td>Version</td>
<td>The Zerto Virtual Manager version.</td>
</tr>
</tbody>
</table>


**RESPONSE: DESCRIPTION**

Pairing statuses:

- **Paired**: The site is paired.
- **Pairing**: The site is in the process of being paired.
- **Unpaired**: The site is not paired.

**Code Examples**

For complete code examples, see “Code Samples”, on page 22.

/v1/peersites cURL Code Example

Retrieve the list of all peer sites.


For more code examples, see “cURL Code”, on page 22.

**Peersites - POST**

Pairs the current site with another, peesite.

**URL**

Pair with another site: https://zvm_ip:port/v1/peersites

Where:

- **zvm_ip**: The IP address of the Zerto Virtual Manager where the API is run.
- **port**: The port to access the Zerto Virtual Manager. The default port is 9669.
Json Request Body Format

The following is an example request Json body for `https://zvm_ip:port/v1/peersites` POST request.

```json
{
    "HostName": "String content",
    "Port": 9081
}
```

Request Values

Request values for `https://zvm_ip:port/v1/peersites` POST request.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>HostName</td>
<td>The address or DNS name of the Zerto Virtual Manager machine that will pair to the current site.</td>
</tr>
<tr>
<td>Port</td>
<td>The default port used for communication between paired Zerto Virtual Managers. The default port is 9081.</td>
</tr>
</tbody>
</table>

XML Request Format

For the XML request format, see “Peer Sites API POST Method Request and Response Formats”, on page 238.

Response Body Format

The following is an example response request Json body for `https://zvm_ip:port/v1/peersites` POST request.

```
["String content"]
```

XML Response Format

For the XML response format, see “Peer Sites API POST Method Request and Response Formats”, on page 238.

Response Values

Response values for `https://zvm_ip:port/v1/peersites` POST request.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>TaskIdentifier</td>
<td>The identifier of the pairing task.</td>
</tr>
</tbody>
</table>

Code Examples

For complete code examples, see “Code Samples”, on page 22.

For cURL code examples, see “cURL Code”, on page 22.

Peersites - DELETE

Unpair the current site from another site paired to it.
URL

Unpair with another site: \[https://zvm_ip:port/v1/peersites/{{SiteIdentifier}}\]

Where:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>zvm_ip</td>
<td>The IP address of the Zerto Virtual Manager machine where the API is run.</td>
</tr>
<tr>
<td>port</td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
<tr>
<td>SiteIdentifier</td>
<td>The identifier of the site to unpair.</td>
</tr>
</tbody>
</table>

Json Request Body Format

The following is an example request Json body for \[https://zvm_ip:port/v1/peersites/{{SiteIdentifier}}\] DELETE request.

```json
{
  "IsKeepTargetDisks": Boolean
}
```

Request Values

Request values for \[https://zvm_ip:port/v1/peersites/{{SiteIdentifier}}\] DELETE request.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>IsKeepTargetDisks</td>
<td>Optional. The default is false. \n\n\n<strong>True</strong>: Keep the target replica disks for any VPGs replicating between the sites as the VPGs will be deleted by unpairing the sites. This will enable faster synchronization in the event of recreating these VPGs at a later stage. \n\n<strong>False</strong>: The target replica disks for the virtual machines are deleted.</td>
</tr>
</tbody>
</table>

Response Body Format

The following is an example response request Json body for \[https://zvm_ip:port/v1/peersites/{{SiteIdentifier}}\] DELETE request.

```
["String content"]
```

Response Values

Response values for \[https://zvm_ip:port/v1/peersites/{{SiteIdentifier}}\] DELETE request.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>TaskIdentifier</td>
<td>The identifier of the pairing task.</td>
</tr>
</tbody>
</table>

XML Response Format

For the XML response format, see “Peer Sites API DELETE Method Request and Response Formats”, on page 239.

Code Examples

For complete code examples, see “Code Samples”, on page 22.

For cURL code examples, see “cURL Code”, on page 22.
Service Profiles API

/v1/serviceprofiles returns information about service profiles that are defined in the Zerto Cloud Manager that is connected to the site where the API runs.

**URL**

<table>
<thead>
<tr>
<th>Type</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>All service profiles</td>
<td>https://zvm_ip:port/v1/serviceprofiles</td>
</tr>
<tr>
<td>Filtered service profiles</td>
<td>https://zvm_ip:port/v1/serviceprofiles?site={SITEIDENTIFIER}</td>
</tr>
<tr>
<td>Single service profile</td>
<td>https://zvm_ip:port/v1/serviceprofiles/{serviceProfileId}</td>
</tr>
</tbody>
</table>

Where:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>zvm_ip</td>
<td>The IP address of the Zerto Virtual Manager where the API is run.</td>
</tr>
<tr>
<td>port</td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
<tr>
<td>serviceProfileId</td>
<td>The service profile ID for which information should be returned.</td>
</tr>
</tbody>
</table>

**Filters**

Filters are optional and any combination of filters is valid. When no filter is specified, all service profiles are returned. Filters are not case-sensitive.

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>siteIdentifier</td>
<td>The identifier of the site for which service profiles should be returned.</td>
</tr>
</tbody>
</table>

**HTTP Method**

GET

**Security**

The API is exposed over HTTPS. Client code must use the x-zerto-session HTTP authorization header.

**See Also**

Starting a session: [Session: POST](#)

**Format**

Json, XML

**Request Format**

The request body is empty.
Json Response Body

The following is an example response Json body for https://zvm_ip:port/v1/serviceprofiles and for https://zvm_ip:port/v1/serviceprofiles/{SiteIdentifier}.

```json
[
  {
    "Description": "String content",
    "History":"P428DT10H30M12.3S",
    "JournalWarningThresholdInPercent":4294967295,
    "Link": {
      "href": "String content",
      "identifier": "String content",
      "rel": "String content",
      "type": "String content"
    },
    "MaxJournalSizeInPercent":4294967295,
    "Rpo":"P428DT10H30M12.3S",
    "ServiceProfileIdentifier": "String content",
    "ServiceProfileName": "String content",
    "TestInterval":"P428DT10H30M12.3S"
  }
]
```

XML Response Format

For the XML response format, see “Service Profiles API XML Response Formats”, on page 239.

Response Values

Response values for https://zvm_ip:port/v1/serviceprofiles and for https://zvm_ip:port/v1/serviceprofiles/{SiteIdentifier}.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>A description of the service profile.</td>
</tr>
<tr>
<td>History</td>
<td>The length of time all write commands are saved in the journal.</td>
</tr>
<tr>
<td>JournalWarningThresholdInPercent</td>
<td>The journal size that triggers a warning that the journal has neared its hard limit, as a percentage of the virtual machine volume size.</td>
</tr>
<tr>
<td>Link</td>
<td>The link details.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used.</td>
</tr>
<tr>
<td>identifier</td>
<td>The unique identifier of the service profile.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level for the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>MaxJournalSizeInPercent</td>
<td>The maximum size that the journal can grow, as a percentage of the virtual machine volume size.</td>
</tr>
<tr>
<td>Rpo</td>
<td>The maximum desired time between each automatic checkpoint being written to the journal before an alert is issued.</td>
</tr>
<tr>
<td>ServiceProfileIdentifier</td>
<td>The unique identifier of the service profile.</td>
</tr>
<tr>
<td>ServiceProfileName</td>
<td>The name of the service profile.</td>
</tr>
<tr>
<td>TestInterval</td>
<td>The period of time specified in the VPG that should pass between testing the failover process.</td>
</tr>
</tbody>
</table>

Code Examples

For complete code examples, see “Code Samples”, on page 22.
Session API

/v1/session starts and ends a session. The following API are available:

- “Session: POST”, below
- “Session: DELETE”, on page 71

<table>
<thead>
<tr>
<th>PURPOSE</th>
<th>METHOD</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start a session</td>
<td>POST</td>
<td>https://zvm_ip:port/v1/session/Add</td>
</tr>
<tr>
<td>End a session</td>
<td>DELETE</td>
<td>https://zvm_ip:port/v1/session</td>
</tr>
</tbody>
</table>

HTTP Methods

POST, DELETE

Security

The API is exposed over HTTPS. Client code must use the x-zerto-session HTTP authorization header when ending a session.

See Also

Starting a session: Session: POST

Format

Json, XML

Session: POST

Sets up an authenticated session for the Zerto Virtual Replication RESTful API.

URL

Start session  https://zvm_ip:port/v1/session/Add

Where:

<table>
<thead>
<tr>
<th>zvm_ip</th>
<th>The IP address of the Zerto Virtual Manager where the API is run.</th>
</tr>
</thead>
<tbody>
<tr>
<td>port</td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
</tbody>
</table>
Zerto Virtual Replication RESTful API Reference Guide - Version 5.5
Zerto Virtual Replication APIs

Session API

Json Request Format

```json
{
  "AuthenticationMethod": "String content"
}
```

Request Values

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>AuthenticationMethod</td>
<td>The authentication to use for the APIs. Possible values are (JSON/XML):</td>
</tr>
<tr>
<td></td>
<td><strong>Windows</strong>: Authentication requires the username and password to access the</td>
</tr>
<tr>
<td></td>
<td>machine where the Zerto Virtual Manager is installed and where the APIs will</td>
</tr>
<tr>
<td></td>
<td>run. Windows authentication is the default if AuthenticationMethod is not set.</td>
</tr>
<tr>
<td></td>
<td><strong>VirtualizationManager</strong>: Authentication requires the username and password</td>
</tr>
<tr>
<td></td>
<td>to access the VMware vCenter Server or Microsoft SCVMM accessed by the Zerto</td>
</tr>
<tr>
<td></td>
<td>Virtual Manager where the APIs will run.</td>
</tr>
</tbody>
</table>

XML Request Format

For the XML response format, see “Session API XML Request Format”, on page 240.

Response Format

The response body is empty.

**Session: DELETE**

Ends a session for the Zerto Virtual Replication RESTful API.

**Note**: If a session is dormant for thirty minutes the session is automatically terminated.

**URL**

End session  https://zvm_ip:port/v1/session

Where:

<table>
<thead>
<tr>
<th>zvm_ip</th>
<th>The IP address of the Zerto Virtual Manager where the API is run.</th>
</tr>
</thead>
<tbody>
<tr>
<td>port</td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
</tbody>
</table>

Request Format

The request body is empty.

Response Format

The response body is empty.

**Code Examples**

For complete code examples, see “Code Samples”, on page 22.

**Python Code Examples**

The following code samples are extracts from the fuller code example, “Python Code”, on page 23.
Starting a Session with VMware vCenter Server or Microsoft SCVMM Credentials

The following code sample shows how to start a session with VMware vCenter Server or Microsoft SCVMM authentication, using the Requests library for making the REST call and the base64 library for password encryption.

```python
import requests
import base64

def GenerateZertoRestSession_VcenterAuthentication(zvmIp, user, password):
    credStr = user + "::" + password
    encodedCredStr = "Basic " + base64.b64encode(credStr)
    print encodedCredStr

    payload = {"AuthenticationMethod": 1}
    dataval = json.dumps(payload)
    print (dataval)

    headers = { 'Authorization': encodedCredStr, 'content-type': 'application/json' }
    url = "https://" + zvmIp + ":9669/v1/session/add"
    r = requests.post(url, data=dataval, headers=headers, verify=False)
    print r.status_code
    if(r.status_code != requests.codes.ok):
        raise Exception("Failed authenticating to ZVM")
    res = r.headers.get('x-zerto-session')
    print "session = " + res
    return res
```

Starting a Session with Windows Credentials

The following code sample shows how to start a session with Windows authentication, which is the default, using the Requests library for making the REST call and the base64 library for password encryption.

```python
import requests
import base64

def GenerateZertoRestSession_WindowsAuthentication(zvmIp, user, password):
    credStr = user + "::" + password
    encodedCredStr = "Basic " + base64.b64encode(credStr)
    print encodedCredStr

    headers = { 'Authorization': encodedCredStr }
    url = "https://" + zvmIp + ":9669/v1/session/add"
    r = requests.post(url, headers=headers, verify=False)
    print r.status_code
    if(r.status_code != requests.codes.ok):
        raise Exception("Failed authenticating to ZVM")
    res = r.headers.get('x-zerto-session')
    print "session = " + res
    return res
```

Ending a Session

The following code sample shows how to end a session.

```python
import requests

def EndZertoRestSession(zvmIp, sessionId):
    url = "https://" + zvmIp + ":9669/v1/session"
    headers = { 'x-zerto-session': sessionId, 'content-type': 'application/json' }
    r = requests.delete(url, headers=headers, verify=False)
    print r.status_code
```
PowerShell Samples

Starting a Session with VMware vCenter Server or Microsoft SCVMM Credentials

The following code sample shows how to start a session with VMware vCenter Server or Microsoft SCVMM authentication.

```powershell
$strZVMIP = "(ZVM IP)"
$strZVMPort = "(ZVM HTTPS port)"
$strZVMUser = "(ZVM user)"
$strZVMPwd = "(ZVM user password)"

## Perform authentication so that Zerto APIs can run. Return a session identifier that needs to be inserted in the header for subsequent requests.
function getxZertoSession ($userName, $password) {
    $baseURL = "https://" + $strZVMIP + ":" + $strZVMPort
    $xZertoSessionURL = $baseURL + "/v1/session/add"
    $authInfo = ("{0}: {1}" -f $userName, $password)
    $authInfo = [System.Convert]::ToBase64String($authInfo)
    $headers = @("Authorization="Basic {0}" -f $authInfo)
    $body = '{"AuthenticationMethod": "1"}'
    $contentType = "application/json"
    $xZertoSessionResponse = Invoke-WebRequest -Uri $xZertoSessionURL -Headers $headers -Method POST -Body $body -ContentType $contentType
    $xZertoSession = getxZertoSession $strZVMUser $strZVMPwd
    $zertoSessionHeader = @("x-zerto-session"=$xZertoSession)
}

Starting a Session with Windows Credentials
To start a session with Windows authentication, replace the line in the above code:

```powershell
$body = '{"AuthenticationMethod": "1"}'
```

with:

```powershell
$body = '{"AuthenticationMethod": "0"}'
```

Starting a Session with Windows Credentials
To start a session with Windows authentication, replace the line in the above code:

```powershell
```

with:

```powershell
String loginType = args.Length > 4 ? args[4] : "0";
```
Tasks API

/v1/tasks returns information about tasks run on the site.

URL

| All tasks | https://zvm_ip:port/v1/tasks |
| Filtered tasks | https://zvm_ip:port/v1/tasks?startedBeforeDate={STARTEDBEFOREDATE}&startedAfterDate={STARTEDAFTERDATE}&completedBeforeDate={COMPLETEDBEFOREDATE}&completedAfterDate={COMPLETEDAFTERDATE}&type={TYPE}&status={STATUS} |
| Single task | https://zvm_ip:port/v1/tasks/{taskIdentifier} |

Valid task types https://zvm_ip:port/v1/tasks/types

Where:

| zvm_ip | The IP address of the Zerto Virtual Manager where the API is run. |
| port | The port to access the Zerto Virtual Manager. The default port is 9669. |
| taskIdentifier | The identifier of the task for which information should be returned. The task identifier is the concatenation of the task and site identifiers, separated by a period. For example, 5e81d46e-c49e-4b2c-b65a-d742a4939192.d2da8a37-68f0-4464-a70c-abb19683fd01 |
| Filters | Filters are optional and any combination of filters is valid. When no filter is specified, all tasks are returned. Filters are not case-sensitive. |

| Filter | Description |
| startedBeforeDate | The date before which the tasks must have begun, supplied in the format yyyy-mm-dd. You can also specify a local time in the format: yyyy-mm-ddThh:mm:ss. Adding Z to the end of the time sets the time to UTC. |
| **startedAfterDate** | The date after which the tasks must have begun, supplied in the format `yyyy-mm-dd`. You can also specify a local time in the format: `yyyy-mm-ddThh:mm:ss`. Adding Z to the end of the time sets the time to UTC. |
| **completedBeforeDate** | The date before which the tasks must have ended, supplied in the format `yyyy-mm-dd`. You can also specify a local time in the format: `yyyy-mm-ddThh:mm:ss`. Adding Z to the end of the time sets the time to UTC. |
| **completedAfterDate** | The date after which the tasks must have ended, supplied in the format `yyyy-mm-dd`. You can also specify a local time in the format: `yyyy-mm-ddThh:mm:ss`. Adding Z to the end of the time sets the time to UTC. |
| **type** | The type of event. For the description of events, refer to the Zerto Virtual Replication documentation about alerts and events. Possible values are: |
| | 0 or `CreateProtectionGroup` |
| | 1 or `RemoveProtectionGroup` |
| | 2 or `FailOver` |
| | 3 or `FailOverTest` |
| | 4 or `StopFailOverTest` |
| | 5 or `Move` |
| | 6 or `GetCheckpointList` |
| | 7 or `ProtectVM` |
| | 8 or `UnprotectVM` |
| | 9 or `AddVMTToProtectionGroup` |
| | 10 or `RemoveVMTFromProtectionGroup` |
| | 11 or `InstallVra` |
| | 12 or `UninstallVra` |
| | 13 or `GetVMSettings` |
| | 14 or `UpdateProtectionGroup` |
| | 15 or `InsertTaggedCP` |
| | 16 or `WaitForCP` |
| | 17 or `HandleMirrorPromotion` |
| | 18 or `ActivateAllMirrors` |
| | 19 or `LogCollection` |
| | 20 or `ClearCheckpoints` |
| | 21 or `ForceReconfigurationOfNewVM` |
| | 22 or `ClearSite` |
| | 23 or `ForceRemoveProtectionGroup` |
| | 24 or `ForceUpdateProtectionGroup` |
| | 25 or `ForceKillProtectionGroup` |
| | 26 or `PrePostScript` |
| | 27 or `InitFullSync` |
| | 28 or `Pair` |
| | 29 or `Unpair` |
| | 30 or `AddPeerVraInfo` |
| | 31 or `RemovePeerVraInfo` |
| | 32 or `InstallCloudConnector` |
| | 33 or `UninstallCloudConnector` |
| | 34 or `HandleFirstSyncDone` |
### Tasks API

| 35 or Clone | 36 or MoveBeforeCommit | 37 or MoveRollback |
| 38 or MoveCommit | 39 or UpgradeVRA | 40 or MaintainHost |
| 41 or NotSupportedInThisVersion | 42 or MoveProtectionGroupToManualOperationNeeded |

| 43 or FailoverBeforeCommit | 44 or FailoverCommit | 45 or FailoverRollback |
| 46 or ChangeVralpSettings | 47 or PauseProtectionGroup | 48 or ResumeProtectionGroup |
| 49 or BulkUpgradeVras | 50 or BulkUninstallVras | 51 or ChangeVraPassword |
| 52 or ChangeRecoveryHost | 53 or ChangeRecoveryHostForProtectionGroup |

| 54 or VpgBackup | 55 or RedeployCloudConnector | 56 or RestoreVpg |
| 57 or VpgDeleteBackup | 58 or SubmitSupportTicket | 59 or PreScript |
| 60 or PostScript | 61 or ChangeVraPasswordIpSettings | 62 or FlrJournalMount |
| 63 or FlrJournalUnmount |

**status**
The status of the task. Possible values are:
1 or InProgress
3 or Paused
4 or Failed
6 or Completed
7 or Cancelling

### HTTP Method

GET

### Security

The API is exposed over HTTPS. Client code must use the `x-zerto-session` HTTP authorization header.

### See Also

Starting a session: [Session: POST](#)
Return the details of the peer sites: [Peer Sites API](#)
Return the details of the local site: [Local Site API](#)
Format

Json, XML

Request Format

The request body is empty.

Json Response Format

The following is an example response Json body for https://zvm_ip:port/v1/tasks and for https://zvm_ip:port/v1/tasks/{taskidentifier}

```json
[{
  "CompleteReason": "String content",
  "Completed": "\Date(928142400000+0300)\",
  "InitiatedBy": "String content",
  "IsCancellable": Boolean,
  "Link": {
    "href": "String content",
    "identifier": "String content",
    "rel": "String content",
    "type": "String content"
  },
  "RelatedEntities": {
    "Hosts": [{
      "href": "String content",
      "identifier": "String content",
      "rel": "String content",
      "type": "String content"
    }],
    "Sites": [{
      "href": "String content",
      "identifier": "String content",
      "rel": "String content",
      "type": "String content"
    }],
    "Vpgs": [{
      "href": "String content",
      "identifier": "String content",
      "rel": "String content",
      "type": "String content"
    }]
  }
},
  "Started": "\Date(928142400000+0300)\",
  "Status": {
    "Progress": 2147483647,
    "State": 0
  },
  "TaskIdentifier": "String content",
  "Type": "String content"
}]
```
XML Response Format

For the XML response format, see “Tasks API XML Response Format”, on page 240.

Response Values

Response values for https://zvm_ip:port/v1/tasks and for https://zvm_ip:port/v1/task/{taskidentifier}.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>CompleteReason</td>
<td>The reason the task completed.</td>
</tr>
<tr>
<td>Completed</td>
<td>The date the task completed. The value can be converted to an understandable date using code similar to the following: var date = new Date(jsonDate); or code similar to the Perl code example, <code>jsonDateToString($)</code> in “Perl Code Example: Retrieving the First 100 Records”, on page 193.</td>
</tr>
<tr>
<td>InitiatedBy</td>
<td>The name of the user who initiated the task.</td>
</tr>
<tr>
<td>IsCancellable</td>
<td><strong>True</strong>: The task can be canceled via user intervention. <strong>False</strong>: The task cannot be canceled via user intervention.</td>
</tr>
<tr>
<td>Link</td>
<td>The link details.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used.</td>
</tr>
<tr>
<td>identifier</td>
<td>The unique identifier of the task.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level for the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>RelatedEntities</td>
<td>Entities possibly affected by the event.</td>
</tr>
<tr>
<td>Hosts</td>
<td>The hosts affected by the task.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used to retrieve host information for each host affected by the event.</td>
</tr>
<tr>
<td>identifier</td>
<td>The unique internal identifier of the host where the API runs.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level for the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>Sites</td>
<td>The sites affected by the task.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used to retrieve site information: peersites and localsite APIs.</td>
</tr>
<tr>
<td>identifier</td>
<td>The unique internal identifier of the site where the API runs.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level for the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>Vpgs</td>
<td>The VPGs affected by the task.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used to retrieve VPG information for each VPG affected by the event.</td>
</tr>
<tr>
<td>identifier</td>
<td>The unique internal identifier of the VPG.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level for the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>FlrSessions</td>
<td>The file level restore sessions affected by the task.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used to retrieve file level restore session information for each session affected by the event.</td>
</tr>
<tr>
<td>identifier</td>
<td>The unique internal identifier of the file level restore session.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level for the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| Started   | The date the task started. The value can be converted to an understandable date using code similar to the following:  
```javascript
var date = new Date(jsonDate);
```

or code similar to the Perl code example, `jsonDateToString($)`, in “Perl Code Example: Retrieving the First 100 Records”, on page 193.

<table>
<thead>
<tr>
<th>Status</th>
<th>The status of the task.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Progress</td>
<td>The progress of the task.</td>
</tr>
</tbody>
</table>

| State | The state of the task. Possible values are (Json/XML):  
0/FirstUnusedValue  
1/InProgress  
2/WaitingForUserInput  
3/Paused  
4/Failed  
5/Stopped  
6/Completed  
7/Cancelling |

<table>
<thead>
<tr>
<th>TaskIdentifier</th>
<th>The unique identifier of the task.</th>
</tr>
</thead>
</table>
| Type           | The type of task:  
0/CreateProtectionGroup  
1/RemoveProtectionGroup  
2/FailOver  
3/FailOverTest  
4/StopFailOverTest  
5/Move  
6/GetCheckpointList  
7/ProtectVM  
8/UnprotectVM  
9/AddVMToProtectionGroup  
10/RemoveVMFromProtectionGroup  
11/InstallVra  
12/UninstallVra  
13/GetVMSettings  
14/UpdateProtectionGroup  
15/InsertTaggedCP  
16/WaitForCP  
17/HandleMirrorPromotion  
18/ActivateAllMirrors  
19/LogCollection  
20/ClearCheckpoints  
21/ForceReconfigurationOfNewVM  
22/ClearSite  
23/ForceRemoveProtectionGroup  
24/ForceUpdateProtectionGroup  
25/ForceKillProtectionGroup |
<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>26/PrePostScript</td>
<td></td>
</tr>
<tr>
<td>27/InitFullSync</td>
<td></td>
</tr>
<tr>
<td>28/Pair</td>
<td></td>
</tr>
<tr>
<td>29/Unpair</td>
<td></td>
</tr>
<tr>
<td>30/AddPeerVraInfo</td>
<td></td>
</tr>
<tr>
<td>31/RemovePeerVraInfo</td>
<td></td>
</tr>
<tr>
<td>32/InstallCloudConnector</td>
<td></td>
</tr>
<tr>
<td>33/UninstallCloudConnector</td>
<td></td>
</tr>
<tr>
<td>34/HandleFirstSyncDone</td>
<td></td>
</tr>
<tr>
<td>35/Clone</td>
<td></td>
</tr>
<tr>
<td>36/MoveBeforeCommit</td>
<td></td>
</tr>
<tr>
<td>37/MoveRollback</td>
<td></td>
</tr>
<tr>
<td>38/MoveCommit</td>
<td></td>
</tr>
<tr>
<td>39/UpgradeVRA</td>
<td></td>
</tr>
<tr>
<td>40/MaintainHost</td>
<td></td>
</tr>
<tr>
<td>41/NotSupportedInThisVersion</td>
<td></td>
</tr>
<tr>
<td>42/MoveProtectionGroupToManualOperationNeeded</td>
<td></td>
</tr>
<tr>
<td>43/FailoverBeforeCommit</td>
<td></td>
</tr>
<tr>
<td>44/FailoverCommit</td>
<td></td>
</tr>
<tr>
<td>45/FailoverRollback</td>
<td></td>
</tr>
<tr>
<td>46/ChangeVralpSettings</td>
<td></td>
</tr>
<tr>
<td>47/PauseProtectionGroup</td>
<td></td>
</tr>
<tr>
<td>48/ResumeProtectionGroup</td>
<td></td>
</tr>
<tr>
<td>49/BulkUpgradeVras</td>
<td></td>
</tr>
<tr>
<td>50/BulkUninstallVras</td>
<td></td>
</tr>
<tr>
<td>51/ChangeVraPassword</td>
<td></td>
</tr>
<tr>
<td>52/ChangeRecoveryHost</td>
<td></td>
</tr>
<tr>
<td>53/ChangeRecoveryHostForProtectionGroup</td>
<td></td>
</tr>
<tr>
<td>54/VpgBackup</td>
<td></td>
</tr>
<tr>
<td>55/RedeployCloudConnector</td>
<td></td>
</tr>
<tr>
<td>56/RestoreVpg</td>
<td></td>
</tr>
<tr>
<td>57/VpgDeleteBackup</td>
<td></td>
</tr>
<tr>
<td>58/SubmitSupportTicket</td>
<td></td>
</tr>
<tr>
<td>59/PreScript</td>
<td></td>
</tr>
<tr>
<td>60/PostScript</td>
<td></td>
</tr>
<tr>
<td>61/ChangeVraPasswordIpSettings</td>
<td></td>
</tr>
<tr>
<td>62/FlrJournalMount</td>
<td></td>
</tr>
<tr>
<td>63/FlrJournalUnmount</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RESPONSE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task types:</td>
<td></td>
</tr>
<tr>
<td>CreateProtectionGroup</td>
<td></td>
</tr>
<tr>
<td>RemoveProtectionGroup</td>
<td></td>
</tr>
<tr>
<td>FailOver</td>
<td></td>
</tr>
<tr>
<td>FailOverTest</td>
<td></td>
</tr>
<tr>
<td>StopFailOverTest</td>
<td></td>
</tr>
<tr>
<td>Move</td>
<td></td>
</tr>
<tr>
<td>GetCheckpointList</td>
<td></td>
</tr>
<tr>
<td>ProtectVM</td>
<td></td>
</tr>
<tr>
<td>UnprotectVM</td>
<td></td>
</tr>
<tr>
<td>AddVMToProtectionGroup</td>
<td></td>
</tr>
<tr>
<td>RemoveVMFromProtectionGroup</td>
<td></td>
</tr>
<tr>
<td>InstallVra</td>
<td></td>
</tr>
<tr>
<td>UninstallVra</td>
<td></td>
</tr>
<tr>
<td>GetVMSettings</td>
<td></td>
</tr>
<tr>
<td>UpdateProtectionGroup</td>
<td></td>
</tr>
<tr>
<td>InsertTaggedCP</td>
<td></td>
</tr>
<tr>
<td>WaitForCP</td>
<td></td>
</tr>
<tr>
<td>HandleMirrorPromotion</td>
<td></td>
</tr>
<tr>
<td>ActivateAllMirrors</td>
<td></td>
</tr>
<tr>
<td>LogCollection</td>
<td></td>
</tr>
<tr>
<td>ClearCheckpoints</td>
<td></td>
</tr>
<tr>
<td>ForceReconfigurationOfNewVM</td>
<td></td>
</tr>
<tr>
<td>ClearSite</td>
<td></td>
</tr>
<tr>
<td>ForceRemoveProtectionGroup</td>
<td></td>
</tr>
<tr>
<td>ForceUpdateProtectionGroup</td>
<td></td>
</tr>
<tr>
<td>ForceKillProtectionGroup</td>
<td></td>
</tr>
<tr>
<td>PrePostScript</td>
<td></td>
</tr>
<tr>
<td>InitFullSync</td>
<td></td>
</tr>
<tr>
<td>Pair</td>
<td></td>
</tr>
<tr>
<td>Unpair</td>
<td></td>
</tr>
<tr>
<td>AddPeerVraInfo</td>
<td></td>
</tr>
<tr>
<td>RemovePeerVraInfo</td>
<td></td>
</tr>
<tr>
<td>InstallCloudConnector</td>
<td></td>
</tr>
<tr>
<td>UninstallCloudConnector</td>
<td></td>
</tr>
<tr>
<td>HandleFirstSyncDone</td>
<td></td>
</tr>
<tr>
<td>Clone</td>
<td></td>
</tr>
<tr>
<td>MoveBeforeCommit</td>
<td></td>
</tr>
<tr>
<td>MoveRollback</td>
<td></td>
</tr>
<tr>
<td>MoveCommit</td>
<td></td>
</tr>
<tr>
<td>UpgradeVra</td>
<td></td>
</tr>
<tr>
<td>MaintainHost</td>
<td></td>
</tr>
<tr>
<td>NotSupportedInThisVersion</td>
<td></td>
</tr>
<tr>
<td>MoveProtectionGroupToManualOperationNeeded</td>
<td></td>
</tr>
</tbody>
</table>
Code Examples

For complete code examples, see "Code Samples", on page 22.

/v1/tasks cURL Code Example
Retrieve information about a specific task.


For more code examples, see “cURL Code”, on page 22.

/v1/tasks Python Code Example
The following code samples are extracts from the fuller code example, “Python Code”, on page 23.
The following code sample checks if a task has completed.

```python
import requests
import base64

def IsTaskComplete(zvmIp, sessionId, TaskId):
    url = "https://" + zvmIp + ":9669/v1/tasks/" + TaskId
    headers = {'x-zerto-session': sessionId, 'content-type': 'application/json'}
    r = requests.get(url, headers=headers, verify=False)
    print r.status_code
    if(r.status_code != requests.codes.ok):
        raise Exception("Failed getting tasks")
    tasksRes = json.loads(r.text)
    print tasksRes
    taskStatus = tasksRes["Status"]
    state = taskStatus["State"]
    print state
    if(state == 4 or state == 5):
        raise Exception("task failed")
    res = state == 6
    print res
    return res
```

**Virtualization Sites API**

/v1/virtualizationsites returns information about the hypervisor site where the API is run and all the sites paired with this site. The information returned can be tailored to specific information about the resources managed at a specified site.

For **VCD Virtualization Sites APIs** see “vCD Virtualization Sites APIs”, on page 196.

**URL**

- All sites: https://zvm_ip:port/v1/virtualizationsites
- Single site: https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}
- Storage clusters at site: https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/datastoreclusters
- Storage at site: https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/datastores
- Folders at site: https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/folders
- Host clusters at site: https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/hostclusters
- Hosts at site: https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/hosts
- Single host at site: https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/hosts/{hostIdentifier}
- Networks at site: https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/networks
- Resource pools at site: https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/resourcepools
- Unprotected VMs at site: https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/vms
Zerto Virtual Replication RESTful API Reference Guide - Version 5.5
Zerto Virtual Replication APIs

Where:

<table>
<thead>
<tr>
<th>zvm_ip</th>
<th>The IP address of the Zerto Virtual Manager where the API is run.</th>
</tr>
</thead>
<tbody>
<tr>
<td>port</td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
<tr>
<td>siteIdentifier</td>
<td>The identifier of the site for which information is returned.</td>
</tr>
<tr>
<td>hostIdentifier</td>
<td>The identifier of the host for which information is returned.</td>
</tr>
</tbody>
</table>

**HTTP Method**

GET

**Security**

The API is exposed over HTTPS. Client code must use the x-zerto-session HTTP authorization header.

**See Also**

Starting a session: Session: POST
vCD APIs: Managing vCD APIs
Return the details of the peer sites: Peer Sites API
Return a list of protected virtual machines: Protected VMs API

**Format**

Json, XML

**Request Format**

The request body is empty.

**Json Response Format**

**Virtualization sites** The following is an example response Json body for https://zvm_ip:port/v1/virtualizationsites and for https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}.

```json
[
  {
    "SiteIdentifier": "String content",
    "VirtualizationSiteName": "String content"
  }
]
```

**Datastore clusters** The following is an example response Json body for https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/datastoreclusters.

```json
[
  {
    "DatastoreClusterName": "String content"
  }
]
```

**Datastores** The following is an example response Json body for https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/datastores.
Folders

The following is an example response JSON body for
https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/folders:

```json
[
  {
    "FolderIdentifier": "String content",
    "FolderName": "String content"
  }
]
```

Host clusters

The following is an example response JSON body for
https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/hostclusters:

```json
[
  {
    "ClusterIdentifier": "String content",
    "VirtualizationClusterName": "String content"
  }
]
```

Hosts

The following is an example response JSON body for
https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/hosts and for
https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/hosts/{hostIdentifier}:

```json
[
  {
    "HostIdentifier": "String content",
    "VirtualizationHostName": "String content"
  }
]
```

Networks

The following is an example response JSON body for
https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/networks:

```json
[
  {
    "NetworkIdentifier": "String content",
    "VirtualizationNetworkName": "String content"
  }
]
```

VMware only: Resource Pools

The following is an example response JSON body for
https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/resourcepools:

```json
[
  {
    "ResourcePoolIdentifier": "String content",
    "ResourcepoolName": "String content"
  }
]
```

Unprotected VMs

The following is an example response JSON body for
https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/vms:

```json
[
  {
    "VmIdentifier": "String content",
    "VmName": "String content"
  }
]
```

XML Response Format

For the XML response format, see “Virtualization Sites API XML Response Format”, on page 242.
### Response Values


<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>SiteIdentifier</td>
<td>The identifier of the Zerto Virtual Manager site.</td>
</tr>
<tr>
<td>VirtualizationSiteName</td>
<td>The hypervisor manager name. For example the VMware vCenter Server name or Microsoft SCVMM name.</td>
</tr>
</tbody>
</table>

Response values for `https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/datastoreclusters`.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>DatastoreClusterName</td>
<td>The storage cluster name.</td>
</tr>
</tbody>
</table>

**Note:** Storage in Microsoft SCVMM is also returned using this API.

Response values for `https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/datastores`.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>DatastoreIdentifier</td>
<td>The internal identifier of the storage. The identifier comprises the server identifier and the storage moref, with the format, <code>serverid.moref</code>.</td>
</tr>
<tr>
<td>DatastoreName</td>
<td>The storage name.</td>
</tr>
</tbody>
</table>

**Note:** Storage in Microsoft SCVMM is also returned using this API and the response labels are `DatastoreIdentifier` and `DatastoreName`.

Response values for `https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/folders`.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>FolderIdentifier</td>
<td>The internal identifier of the folder. The identifier comprises the server identifier and the folder moref, with the format, <code>serverid.moref</code>.</td>
</tr>
<tr>
<td>FolderName</td>
<td>The folder name.</td>
</tr>
</tbody>
</table>

Response values for `https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/hostclusters`.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ClusterIdentifier</td>
<td>The unique identifier of the host cluster. The identifier comprises the server identifier and the host cluster moref, with the format, <code>serverid.moref</code>.</td>
</tr>
<tr>
<td>VirtualizationClusterName</td>
<td>The name of the host cluster.</td>
</tr>
</tbody>
</table>

Response values for `https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/hosts` and for `https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/hosts/{hostIdentifier}`.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>HostIdentifier</td>
<td>The unique identifier of the host.</td>
</tr>
<tr>
<td>VirtualizationHostName</td>
<td>The name of the host.</td>
</tr>
</tbody>
</table>


<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>NetworkIdentifier</td>
<td>The internal identifier of the network.</td>
</tr>
<tr>
<td>VirtualizationNetworkName</td>
<td>The network name.</td>
</tr>
</tbody>
</table>
VMware only: Response values for
https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/resourcepools.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResourcePoolId</td>
<td>The internal identifier of the resource pool.</td>
</tr>
<tr>
<td>ResourcePoolName</td>
<td>The resource pool name.</td>
</tr>
</tbody>
</table>

Response values for https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/vms.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>VmId</td>
<td>The internal identifier of the virtual machine that is not protected by Zerto Virtual Replication.</td>
</tr>
<tr>
<td>VmName</td>
<td>The virtual machine name that is not protected by Zerto Virtual Replication.</td>
</tr>
</tbody>
</table>

Note: The response includes:
- Virtual machines that are not protected in any VPG.
- Virtual machines that are protected in VPGs but can be protected in additional VPGs. A virtual machine can be protected in a maximum number of three VPGs as long as:
  - The VPGs do not recover to the same site.
  - The protected and recovery sites, as well as the VRAs on each site, are of version 5.0 and higher.

Code Examples
For complete code examples, see “Code Samples”, on page 22.

/v1/virtualizationsites cURL Code Example
Retrieve information about the VMware datastores/Microsoft storage available at a specific site.

```
```

For more code examples, see “cURL Code”, on page 22.
Protected VMs API

/v1/vms returns information about all virtual machines protected on the site processing the API.

**URL**

- **All VMs**  
  https://zvm_ip:port/v1/vms

- **Filtered VMs**  
  https://zvm_ip:port/v1/vms?vpgName={VPGNAME}&vmName={VMNAME}&status={STATUS}&substatus={SUBSTATUS}&organizationName={ORGNAME}&priority={PRIORITY}&protectedSiteType={PROTECTEDSITEIDENTIFIER}&recoverySiteType={RECOVERYSITEIDENTIFIER}&protectedSiteIdentifier={PROTECTEDSITEIDENTIFIER}&recoverySiteIdentifier={RECOVERYSITEIDENTIFIER}&vmIdentifier={vmIdentifier}

- **Single VM**  
  https://zvm_ip:port/v1/vms/{vmIdentifier}

**Note:** Running an API for a single virtual machine returns details of the single virtual machine only if it is protected in a single VPG. For information about virtual machines that are protected in several VPGs, use the filter `vmIdentifier`.

Where:

- **zvm_ip**  
  The IP address of the Zerto Virtual Manager where the API is run.
- **port**  
  The port to access the Zerto Virtual Manager. The default port is 9669.
- **vmIdentifier**  
  The identifier of the virtual machine for which information is returned. The identifier comprises the server identifier and the virtual machine moref, with the format, serverid.moref.
- **Filters**  
  Filters are optional and any combination of filters is valid. When no filter is specified, all virtual machines are returned. Filters are not case-sensitive.

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vpgName</td>
<td>The name of the VPG which protects the virtual machine.</td>
</tr>
<tr>
<td>vmName</td>
<td>The name of the virtual machine.</td>
</tr>
<tr>
<td>status</td>
<td>The status of the VPG. Possible values are:</td>
</tr>
<tr>
<td></td>
<td>0 or <strong>Initializing</strong>: The VPG is being initialized. This includes when a VPG is created, and during the initial sync between sites.</td>
</tr>
<tr>
<td></td>
<td>1 or <strong>MeetingSLA</strong>: The VPG is meeting the SLA specification.</td>
</tr>
<tr>
<td></td>
<td>2 or <strong>NotMeetingSLA</strong>: The VPG is not meeting the SLA specification for both the journal history and RPO SLA settings.</td>
</tr>
<tr>
<td></td>
<td>3 or <strong>RpoNotMeetingSLA</strong>: The VPG is not meeting the SLA specification for the RPO SLA setting.</td>
</tr>
<tr>
<td></td>
<td>4 or <strong>HistoryNotMeetingSLA</strong>: The VPG is not meeting the SLA specification for the journal history.</td>
</tr>
<tr>
<td></td>
<td>5 or <strong>FailingOver</strong>: The VPG is in a Failover operation.</td>
</tr>
<tr>
<td></td>
<td>6 or <strong>Moving</strong>: The VPG is in a Move operation.</td>
</tr>
<tr>
<td></td>
<td>7 or <strong>Deleting</strong>: The VPG is being deleted.</td>
</tr>
<tr>
<td></td>
<td>8 or <strong>Recovered</strong>: The VPG has been recovered.</td>
</tr>
<tr>
<td>Substatus</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>0 or None</td>
<td>The VPG is in a bitmap sync. For the description of substatuses, refer to the Zerto Virtual Manager Administration Guide.</td>
</tr>
<tr>
<td>1 or InitialSync</td>
<td>Possible values are:</td>
</tr>
<tr>
<td>2 or Creating</td>
<td>0 or None, 1 or InitialSync, 2 or Creating, 3 or VolumeInitialSync, 4 or Sync, 5 or RecoveryPossible, 6 or DeltaSync,</td>
</tr>
<tr>
<td></td>
<td>7 or NeedsConfiguration, 8 or Error, 9 or EmptyProtectionGroup, 10 or DisconnectedFromPeerNoRecoveryPoints, 11 or FullSync,</td>
</tr>
<tr>
<td></td>
<td>12 or VolumeDeltaSync, 13 or VolumeFullSync, 14 or FailingOverCommitting, 15 or FailingOverBeforeCommit, 16 or</td>
</tr>
<tr>
<td></td>
<td>FailingOverRollingBack, 17 or Promoting, 18 or MovingCommitting, 19 or MovingBeforeCommit, 20 or MovingRollingBack,</td>
</tr>
<tr>
<td></td>
<td>21 or Deleting, 22 or PendingRemove, 23 or BitmapSync, 24 or DisconnectedFromPeer, 25 or ReplicationPausedUserInitiated,</td>
</tr>
<tr>
<td></td>
<td>26 or ReplicationPausedSystemInitiated, 27 or RecoveryStorageProfileError, 29 or RollingBack, 30 or RecoveryStorageError,</td>
</tr>
<tr>
<td></td>
<td>31 or JournalStorageError, 32 or VmNotProtectedError, 33 or JournalOrRecoveryMissingError, 34 or AddedVmsInInitialSync,</td>
</tr>
<tr>
<td></td>
<td>35 or ReplicationPausedForMissingVolume</td>
</tr>
</tbody>
</table>

**sourceType**

<table>
<thead>
<tr>
<th>Description</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Deprecated. See protectedSiteType</td>
<td></td>
</tr>
</tbody>
</table>

**targetType**

<table>
<thead>
<tr>
<th>Description</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Deprecated. See recoverySiteType</td>
<td></td>
</tr>
</tbody>
</table>

**sourceSite**

<table>
<thead>
<tr>
<th>Description</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Deprecated. See protectedSiteIdentifier</td>
<td></td>
</tr>
</tbody>
</table>

**targetSite**

<table>
<thead>
<tr>
<th>Description</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Deprecated. See recoverySiteIdentifier</td>
<td></td>
</tr>
</tbody>
</table>

**protectedSiteType**

<table>
<thead>
<tr>
<th>Description</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>The protected site environment. Possible values are:</td>
<td></td>
</tr>
<tr>
<td>0 or VCVpg: The VPG is protecting virtual machines in a VMware vCenter Server.</td>
<td></td>
</tr>
<tr>
<td>1 or VCVApp: Deprecated. See VCDvApp.</td>
<td></td>
</tr>
<tr>
<td>2 or VCDvApp: The VPG is protecting a VMware vCloud Director vApp.</td>
<td></td>
</tr>
<tr>
<td>3 or PublicCloud: Not applicable.</td>
<td></td>
</tr>
<tr>
<td>4 or HyperV: The VPG is protecting virtual machines in Microsoft Hyper-V.</td>
<td></td>
</tr>
</tbody>
</table>
### HTTP Method

GET

### Security

The API is exposed over HTTPS. Client code must use the `x-zerto-session` HTTP authorization header.

### See Also

- Starting a session: Session: POST
- Return a list of unprotected virtual machines: Virtualization Sites API:  
  https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/vms
- Return a list of unprotected vCD vApps: List Unprotected vCD vApps in a Site:  
  https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/vcdvapps

### Format

Json, XML

### Request Format

The request body is empty.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>recoverySiteType</td>
<td>The recovery site environment. Possible values are:</td>
</tr>
<tr>
<td></td>
<td>0 or VCVpg: The VPG is recovering virtual machines in a VMware vCenter Server.</td>
</tr>
<tr>
<td></td>
<td>1 or VCDvApp: Deprecated. See VCDvApp.</td>
</tr>
<tr>
<td></td>
<td>2 or VCDvApp: The VPG is recovering a VMware vCloud Director vApp.</td>
</tr>
<tr>
<td></td>
<td>3 or PublicCloud: The VPG is recovering virtual machines to a public cloud.</td>
</tr>
<tr>
<td></td>
<td>4 or HyperV: The VPG is recovering virtual machines in Microsoft Hyper-V.</td>
</tr>
<tr>
<td>protectedSiteIdentifier</td>
<td>The identifier of the protected site.</td>
</tr>
<tr>
<td>recoverySiteIdentifier</td>
<td>The identifier of the recovery site.</td>
</tr>
<tr>
<td>organizationName</td>
<td>The ZORG for this VPG.</td>
</tr>
<tr>
<td>priority</td>
<td>The priority specified for the VPG. Possible values are:</td>
</tr>
<tr>
<td></td>
<td>0 or Low: The VPG has a low priority for transferring data.</td>
</tr>
<tr>
<td></td>
<td>1 or Medium: The VPG has a medium priority for transferring data.</td>
</tr>
<tr>
<td></td>
<td>2 or High: The VPG has a high priority for transferring data.</td>
</tr>
</tbody>
</table>
Json Response Format

The following is an example response Json body for https://zvm_ip:port/v1/vms and for https://zvm_ip:port/v1/vms/{vmIdentifier}

```json
[
    {
        "ActualRPO": 2147483647,
        "Entities": {
            "Protected": 0,
            "Recovery": 0,
            "Source": 0,
            "Target": 0
        },
        "IOPS": 2147483647,
        "LastTest": "/Date(928142400000+0300)/",
        "Link_{n}": {
            "href": "String content",
            "identifier": "String content",
            "rel": "String content",
            "type": "String content"
        },
        "OrganizationName": "String content",
        "OutgoingBandWidthInMbps": 1.26743233E+15
        "Priority": 0,
        "ProtectedSite_{n}": {
            "href": "String content",
            "identifier": "String content",
            "rel": "String content",
            "type": "String content"
        },
        "ProvisionedStorageInMB": 2147483647,
        "RecoverySite_{n}": {
            "href": "String content",
            "identifier": "String content",
            "rel": "String content",
            "type": "String content"
        },
        "SourceSite": "String content",
        "Status": 0,
        "SubStatus": 0,
        "TargetSite": "String content",
        "ThroughputInMB": 20,
        "UsedStorageInMB": 2147483647,
        "VmIdentifier": "String content"
        "VmName": "String content",
        "Volumes_{n}": [
            {"VmVolumeIdentifier": "String content",
        }
    },
    "VpgName": "String content",
    "EnabledActions": {
        "IsFlrEnabled": false
    },
    "VpgIdentifier": "String content"
]
```
XML Response Format

For the XML response format, see “VMs API XML Response Format”, on page 243.

### Response Values

Response values for https://zvm_ip:port/v1/vms and for https://zvm_ip:port/v1/vms/{vmIdentifier}

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ActualRPO</td>
<td>The time since the last checkpoint was written to the journal in seconds. This should be less than the Target RPO Alert value specified for the VPG. A value of -1 means that the RPO has not been calculated.</td>
</tr>
<tr>
<td>Entities</td>
<td>The source type and target types, vCenter Server of vCloud Director, where the virtual machine is protected and recovered.</td>
</tr>
</tbody>
</table>
| Protected     | The type of virtual machines being protected. Possible values are (Json/XML):
|               | 0/VCVpg: The VPG is protecting virtual machines in a VMware vCenter Server.
|               | 2/VCDvApp: The VPG is protecting a VMware vCloud Director vApp.                                   |
|               | 3/AWS: Not applicable.                                                                          |
|               | 4/HyperV: The VPG is protecting virtual machines in Microsoft Hyper-V.                            |
| Recovery      | The type of virtual machines being recovered. Possible values are (Json/XML):
|               | 0/VCVpg: The VPG is recovering virtual machines in a VMware vCenter Server.                      |
|               | 2/VCDvApp: The VPG is recovering a VMware vCloud Director vApp.                                   |
|               | 3/AWS: The VPG is recovering virtual machines in Amazon Web Services (AWS).                      |
|               | 4/HyperV: The VPG is recovering virtual machines in Microsoft Hyper-V.                           |
| Source        | Deprecated. See Protected.                                                                     |
| Target        | Deprecated. See Recovery.                                                                     |
| IOPS          | The IO per second between all the applications running on the virtual machine in the VPG and the VRA that sends a copy to the remote site for replication. |
| LastTest      | The date the last failover test occurred. The value can be converted to an understandable date using code similar to the following:
<p>|               | var date = new Date(jsonDate);                                                                   |
|               | or code similar to the Perl code example, jsonDateToString($), in “Perl Code Example: Retrieving the First 100 Records”, on page 193. |
| Link          | The link details.                                                                               |
| href          | The URL used.                                                                                   |
| identifier    | The unique identifier of the virtual machine.                                                   |
| rel           | The next path level for the API relative to the current path.                                   |
| type          | The API interface service.                                                                     |
| Link          | The link details.                                                                               |
| href          | The URL used.                                                                                   |
| rel           | The next path level for the API relative to the current path.                                   |
| type          | The API interface service.                                                                     |
| OrganizationName | The name of the organization set up in the Zerto Cloud Manager, the ZORG, that uses a cloud service provider for recovery. |
| OutgoingBandWidthInMbps | The bandwidth throttling defined for the virtual machines. |</p>
<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
</table>
| Priority               | The priority specified for the VPG. Possible values are (Json/XML):  
0/Low: The VPG has a low priority for transferring data.  
1/Medium: The VPG has a medium priority for transferring data.  
2/High: The VPG has a high priority for transferring data. |
| ProtectedSite          | The identifier of the protected site.                                                                                                                                                                     |
| href                   | The URL used.                                                                                                                                                                                           |
| identifier             | The unique identifier of the virtual machine.                                                                                                                                                             |
| rel                    | The next path level for the API relative to the current path.                                                                                                                                           |
| type                   | The API interface service.                                                                                                                                                                              |
| ProvisionedStorageInMB | The storage provisioned for the virtual machine in the recovery site.                                                                                                                                   |
| RecoverySite           | The identifier of the recovery site.                                                                                                                                                                     |
| href                   | The URL used.                                                                                                                                                                                           |
| identifier             | The unique identifier of the virtual machine.                                                                                                                                                             |
| rel                    | The next path level for the API relative to the current path.                                                                                                                                           |
| type                   | The API interface service.                                                                                                                                                                              |
| SourceSite             | Deprecated. See ProtectedSite > identifier.                                                                                                                                                             |
| Status                 | The status of the VPG that contains the virtual machine. Possible values are (Json/XML):  
0/Initializing: The VPG is being initialized. This includes when a VPG is created, and during the initial sync between sites.  
1/MeetingSLA: The VPG is meeting the SLA specification.  
2/NotMeetingSLA: The VPG is not meeting the SLA specification for both the journal history and RPO SLA settings, for example during a delta sync or when there is an error.  
3/RpoNotMeetingSLA: The VPG is not meeting the SLA specification for the RPO SLA setting.  
4/HistoryNotMeetingSLA: The VPG is not meeting the SLA specification for the journal history.  
5/FailingOver: The VPG is in a Failover operation.  
6/Moving: The VPG is in a Move operation.  
7/Deleting: The VPG is being deleted.  
8/Recovered: The VPG is recovered. |
| SubStatus              | The substatus of the VPG that contains the virtual machine, for example the VPG is in a bitmap sync. For the description of substatuses, refer to the Zerto Virtual Manager Administration Guide. Possible values are (Json/XML):  
0/None  
1/InitialSync  
2/Creating  
3/VolumeInitialSync  
4/Sync  
5/RecoveryPossible  
6/DeltaSync  
7/NeedsConfiguration  
8/Error  
9/EmptyProtectionGroup |
## Code Examples

For complete code examples, see “Code Samples”, on page 22.

### /v1/vms cURL Code Example

Retrieve the list of all protected virtual machines and information about them.

```bash
```
## VPGs API

/v1/vpgs returns information about VPGs or creates or performs actions on a specific VPG. The following API are available:

- **“VPGs: GET”, below**
- **“VPGs: POST”, on page 109**
- **“VPGs: DELETE”, on page 114**

<table>
<thead>
<tr>
<th>PURPOSE</th>
<th>METHOD</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Information for all VPGs</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/vpgs</td>
</tr>
<tr>
<td>Information for one VPG</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}</td>
</tr>
<tr>
<td>Checkpoints for a VPG</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/checkpoints</td>
</tr>
<tr>
<td>Checkpoints Summary for a VPG</td>
<td>GET</td>
<td>Deprecated</td>
</tr>
<tr>
<td>Checkpoints Statistics for a VPG</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/checkpoints/stats</td>
</tr>
<tr>
<td>Valid values for VPG entities</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/vpgs/entitytypes</td>
</tr>
<tr>
<td>Valid values for failover commit policies</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/vpgs/failovercommitpolicies</td>
</tr>
<tr>
<td>Valid values for failover shutdown policies</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/vpgs/failovershutdownpolicies</td>
</tr>
<tr>
<td>Valid values for VPG priorities</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/vpgs/priorities</td>
</tr>
<tr>
<td>Valid values for failover retention policies</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/vpgs/retentionpolicies</td>
</tr>
<tr>
<td>Valid values for VPG statuses</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/vpgs/statuses</td>
</tr>
<tr>
<td>Valid values for VPG substatuses</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/vpgs/substatuses</td>
</tr>
<tr>
<td>Create a VPG</td>
<td>POST</td>
<td>https://zvm_ip:port/v1/vpgs</td>
</tr>
<tr>
<td>Clone a VPG</td>
<td>POST</td>
<td>https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/CloneStart</td>
</tr>
<tr>
<td>Abort a VPG clone</td>
<td>POST</td>
<td>https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/CloneAbort</td>
</tr>
<tr>
<td>Failover a VPG</td>
<td>POST</td>
<td>https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/Failover</td>
</tr>
<tr>
<td>Commit a VPG failover</td>
<td>POST</td>
<td>https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/FailoverCommit</td>
</tr>
<tr>
<td>Rollback a VPG failover</td>
<td>POST</td>
<td>https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/FailoverRollback</td>
</tr>
<tr>
<td>Test a VPG</td>
<td>POST</td>
<td>https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/FailoverTest</td>
</tr>
<tr>
<td>Stop a VPG test</td>
<td>POST</td>
<td>https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/FailoverTestStop</td>
</tr>
<tr>
<td>Force synchronize a VPG</td>
<td>POST</td>
<td>https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/forcesync</td>
</tr>
<tr>
<td>Pause VPG protection</td>
<td>POST</td>
<td>https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/pause</td>
</tr>
<tr>
<td>Resume VPG protection</td>
<td>POST</td>
<td>https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/resume</td>
</tr>
<tr>
<td>Delete a VPG</td>
<td>DELETE</td>
<td>https://zvm_ip:port/v1/vpgs/{VpgIdentifier}</td>
</tr>
</tbody>
</table>
HTTP Methods
GET, POST, DELETE

Security
The API is exposed over HTTPS. Client code must use the x-zerto-session HTTP authorization header.

See Also
Starting a session: Session: POST
View the results of the create VPG operation: Events API
Return the datastore identifier: Datastores
Return the resource pool identifier: VMware only: Resource Pools
Return the source and target site identifiers: Virtualization sites
Return the vCenter vApp identifier to protect: Unprotected VMs
vCD APIs: Managing vCD APIs
Return the list of unprotected virtual machine identifiers: Unprotected VMs
Retrieve details of the task being performed: Tasks API
Manage a VPG: VPG Management API:

Format
Json, XML
# VPGs: GET

Returns information about VPGs.

## URL

- **All VPGs**
  - `https://zvm_ip:port/v1/vpgs`

- **Filtered VPGs**
  - `https://zvm_ip:port/v1/vpgs?name={NAME}&status={STATUS}&substatus={SUBSTATUS}&protectedSiteType={PROTECTEDSITETYPE}&recoverySiteType={RECOVERYSITETYPE}&protectedSiteIdentifier={PROTECTEDSITEIDENTIFIER}&recoverySiteIdentifier={RECOVERYSITEIDENTIFIER}&organizationName={ORGNAME}&zorgIdentifier={ZORGIDENTIFIER}&priority={PRIORITY}&serviceProfileIdentifier={SERVICEPROFILEIDENTIFIER}&backupEnabled={BACKUPENABLED}`

- **Single VPG**
  - `https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}`

- **VPG Checkpoints**
  - `https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/checkpoints?startDate={STARTDATE}&endDate={ENDDATE}`

- **VPG Checkpoints summary**
  - Deprecated

- **VPG Checkpoints stats**
  - `https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/checkpoints/stats`

- **Valid VPG entities**
  - `https://zvm_ip:port/v1/vpgs/entitytypes`

- **Valid failover commit policies**
  - `https://zvm_ip:port/v1/vpgs/failovercommitpolicies`

- **Valid failover shutdown policies**
  - `https://zvm_ip:port/v1/vpgs/failovershutdownpolicies`

- **Valid VPG priorities**
  - `https://zvm_ip:port/v1/vpgs/priorities`

- **Valid failover retention policies**
  - `https://zvm_ip:port/v1/vpgs/retentionpolicies`

- **Valid VPG statuses**
  - `https://zvm_ip:port/v1/vpgs/statuses`

- **Valid VPG substatuses**
  - `https://zvm_ip:port/v1/vpgs/substatuses`

Where:

**Filters for: All VPGs**

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>zvm_ip</td>
<td>The IP address of the Zerto Virtual Manager where the API is run.</td>
</tr>
<tr>
<td>port</td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
<tr>
<td>protectionGroupIdentifier</td>
<td>The identifier of the VPG for which information is retrieved.</td>
</tr>
</tbody>
</table>

**Filters**

Filters are optional and any combination of filters is valid. When no filter is specified, all VPGs are returned. Filters are not case-sensitive.

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>The name of the VPG.</td>
</tr>
</tbody>
</table>
### status

The status of the VPG. Possible values are:

- **0** or **Initializing**: The VPG is being initialized. This includes when a VPG is created, and during the initial sync between sites.
- **1** or **MeetingSLA**: The VPG is meeting the SLA specification.
- **2** or **NotMeetingSLA**: The VPG is not meeting the SLA specification for both the journal history and RPO SLA settings.
- **3** or **RpoNotMeetingSLA**: The VPG is not meeting the SLA specification for the RPO SLA setting.
- **4** or **HistoryNotMeetingSLA**: The VPG is not meeting the SLA specification for the journal history.
- **5** or **FailingOver**: The VPG is in a Failover operation.
- **6** or **Moving**: The VPG is in a Move operation.
- **7** or **Deleting**: The VPG is being deleted.
- **8** or **Recovered**: The VPG has been recovered.

### SubStatus

The substatus of the VPG, for example the VPG is in a bitmap sync. For the description of substatuses, refer to the Zerto Virtual Manager Administration Guide. Possible values are

- **0** or **None**
- **1** or **InitialSync**
- **2** or **Creating**
- **3** or **VolumeInitialSync**
- **4** or **Sync**
- **5** or **RecoveryPossible**
- **6** or **DeltaSync**
- **7** or **NeedsConfiguration**
- **8** or **Error**
- **9** or **EmptyProtectionGroup**

- **10** or **DisconnectedFromPeerNoRecoveryPoints**
- **11** or **FullSync**
- **12** or **VolumeDeltaSync**
- **13** or **VolumeFullSync**
- **14** or **FailingOverCommitting**
- **15** or **FailingOverBeforeCommit**
- **16** or **FailingOverRollingBack**
- **17** or **Promoting**
- **18** or **MovingCommitting**
- **19** or **MovingBeforeCommit**

- **20** or **MovingRollingBack**
- **21** or **Deleting**
- **22** or **PendingRemove**
- **23** or **BitmapSync**
- **24** or **DisconnectedFromPeer**
- **25** or **ReplicationPausedUserInitiated**
- **26** or **ReplicationPausedSystemInitiated**
- **27** or **RecoveryStorageProfileError**
- **29** or **RollingBack**

- **30** or **RecoveryStorageError**
- **31** or **JournalStorageError**
- **32** or **VmNotProtectedError**
- **33** or **JournalOrRecoveryMissingError**
- **34** or **AddedVmsInInitialSync**
- **35** or **ReplicationPausedForMissingVolume**
**protectedSiteType**
The protected site environment. This filter behaves in the same way as the `sourceType` filter. Possible values are:

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 or VCVpg</td>
<td>The VPG is protecting virtual machines in a VMware vCenter Server.</td>
</tr>
<tr>
<td>1 or VCVApp</td>
<td>Deprecated. See VCDvApp.</td>
</tr>
<tr>
<td>2 or VCDvApp</td>
<td>The VPG is protecting a VMware vCloud Director vApp.</td>
</tr>
<tr>
<td>3 or PublicCloud</td>
<td>Not applicable.</td>
</tr>
<tr>
<td>4 or HyperV</td>
<td>The VPG is protecting virtual machines in Microsoft Hyper-V.</td>
</tr>
</tbody>
</table>

**recoverySiteType**
The recovery site environment. This filter behaves in the same way as the `targetType` filter. Possible values are:

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 or VCVpg</td>
<td>The VPG is recovering virtual machines to a VMware vCenter Server.</td>
</tr>
<tr>
<td>1 or VCVApp</td>
<td>Deprecated. See VCDvApp.</td>
</tr>
<tr>
<td>2 or VCDvApp</td>
<td>The VPG is recovering virtual machines to a VMware vCloud Director vApp.</td>
</tr>
<tr>
<td>3 or PublicCloud</td>
<td>The VPG is recovering virtual machines to a Public Cloud.</td>
</tr>
<tr>
<td>4 or HyperV</td>
<td>The VPG is recovering virtual machines to Microsoft Hyper-V.</td>
</tr>
</tbody>
</table>

**protectedSiteIdentifier**
The identifier of the source site where the VPG virtual machines are protected.

**recoverySiteIdentifier**
The identifier of the target site where the VPG virtual machines are recovered.

**sourceSite**
Deprecated. See `protectedSiteIdentifier`.

**targetSite**
Deprecated. See `recoverySiteIdentifier`.

**sourceType**
Deprecated. See `protectedSiteType`.

**targetType**
Deprecated. See `recoverySiteType`.

**organizationName**
The ZORG for this VPG.

**zorgIdentifier**
The internal identifier for the ZORG.

**priority**
The VPG priority. Possible values are:

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 or Low</td>
<td>The VPG has a low priority for transferring data.</td>
</tr>
<tr>
<td>1 or Medium</td>
<td>The VPG has a medium priority for transferring data.</td>
</tr>
<tr>
<td>2 or High</td>
<td>The VPG has a high priority for transferring data.</td>
</tr>
</tbody>
</table>

**serviceProfileIdentifier**
The identifier of the service profile to use for the VPG when a Zerto Cloud Manager is used.

**backupEnabled**
True: The VPG is defined for both disaster recovery and for long term recovery via offsite backup.  
False: The VPG is defined for disaster recovery only.

---

**Filters for: Checkpoints**

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>zvm_ip</td>
<td>The IP address of the Zerto Virtual Manager where the API is run.</td>
</tr>
<tr>
<td>port</td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
<tr>
<td>protectionGroupId</td>
<td>The identifier of the VPG for which information is retrieved.</td>
</tr>
</tbody>
</table>

**Filters**
Filters are optional and any combination of filters is valid. When no filter is specified, all VPGs are returned. Filters are not case-sensitive.

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>startDate</td>
<td>The start date of the checkpoint.</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>endDate</td>
<td>The end date of the checkpoint.</td>
</tr>
</tbody>
</table>

**Request Format**

The request body is empty.

**Json Response Format**

The following is an example response Json body for https://zvm_ip:port/v1/vpgs and, without the array statement, for https://zvm_ip:port/v1/vpgs/(protectionGroupIdentifier).

```json
[
  {
    "ActiveProcessesApi": {
      "RunningFailOverTestApi": {
        "Stage": "String content"
      }
    },
    "ActualRPO": 2147483647,
    "BackupEnabled": Boolean,
    "Entities": {
      "Protected": 0,
      "Recovery": 0,
      "Source": 0,
      "Target": 0
    },
    "FailSafeHistory": {
      "ActualFailSafeHistory": 240,
      "ConfiguredFailSafeHistory": 240,
      "FailSafeDescription": "";
    },
    "HistoryStatusApi": {
      "ActualHistoryInMinutes": 1440,
      "ConfiguredHistoryInMinutes": 1440,
      "EarliestCheckpoint": {
        "CheckpointIdentifier": "203456",
        "Tag": null,
        "TimeStamp": "/Date(1474877983000)/",
        "Vss": Boolean
      },
      "IOPS": 2147483647,
      "LastTest": "/Date(9281424000+0300)/",
      "Link": {
        "href": "String content",
        "identifier": "String content",
        "rel": "String content",
        "type": "String content"
      },
      "Link__x007B_0_x007D_": {
        "href": "String content",
        "rel": "String content",
        "type": "String content"
      }
  }
]
```
The following is an example response JSON body for

https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/checkpoints?startDate={STARTDATE}&endDate={ENDDATE}.

```json
{
  "CheckpointIdentifier": "String content",
  "Tag": "String content",
  "TimeStamp": "/Date(928142400000+0300)/",
  "Vss": : Boolean
}
```
The following is an example response Json body for
https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/checkpoints/stats

```json
{
   "Earliest": {
      "CheckpointId": "String content",
      "Tag": "String content",
      "TimeStamp": "\Date(1479047837000)\",
      "Vss": false
   },
   "Latest": {
      "CheckpointId": "String content",
      "Tag": "String content",
      "TimeStamp": "\Date(1479047837000)\",
      "Vss": false
   }
}
```

The following is an example response Json body for
https://zvm_ip:port/v1/vpgs/entitytypes,

```json
["String content"]
```

XML Response Format

For the XML response format, see “VPGs API GET Method Response Formats”, on page 245.

Response Values

VPGs Response values for https://zvm_ip:port/v1/vpgs and for
https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ActiveProcessApi</td>
<td>The current stage of an operation being performed on the VPG.</td>
</tr>
<tr>
<td>RunningFailoverTestApi</td>
<td>The operation being performed.</td>
</tr>
<tr>
<td>Stage</td>
<td>The stage of the running operation:</td>
</tr>
<tr>
<td></td>
<td><strong>InTest</strong>: The VPG is in a failover test operation.</td>
</tr>
<tr>
<td></td>
<td><strong>Starting</strong>: The VPG virtual machines in the recovery site are being initialized for a failover test.</td>
</tr>
<tr>
<td></td>
<td><strong>Stopping</strong>: The recovery site is being cleaned up after a failover test of the VPG.</td>
</tr>
<tr>
<td>ActualRPO</td>
<td>The time since the last checkpoint was written to the journal in seconds. This should be less than the Target RPO Alert value specified for the VPG. A value of -1 means that the RPO has not been calculated.</td>
</tr>
<tr>
<td>BackupEnabled</td>
<td><strong>True</strong>: The VPG is defined for both disaster recovery and for long term recovery via offsite backup.</td>
</tr>
<tr>
<td></td>
<td><strong>False</strong>: The VPG is defined for disaster recovery only.</td>
</tr>
<tr>
<td>ConfiguredRpoSeconds</td>
<td>The source type and target types where the VPG is protected and recovered.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Protected</strong></td>
<td>The protected site environment. This parameter behaves in the same way as the Source parameter. Possible values are (Json/XML):</td>
</tr>
<tr>
<td>0/VCVpg</td>
<td>The VPG is protecting virtual machines in a VMware vCenter Server.</td>
</tr>
<tr>
<td>1/VCvApp</td>
<td>Deprecated. See VCDvApp.</td>
</tr>
<tr>
<td>2/VCDvApp</td>
<td>The VPG is protecting a VMware vCloud Director vApp.</td>
</tr>
<tr>
<td>3/PublicCloud</td>
<td>Not applicable.</td>
</tr>
<tr>
<td>4/HyperV</td>
<td>The VPG is protecting virtual machines in Microsoft Hyper-V.</td>
</tr>
<tr>
<td><strong>Recovery</strong></td>
<td>The recovery site environment. This parameter behaves in the same way as the Target parameter. Possible values are (Json/XML):</td>
</tr>
<tr>
<td>0/VCVpg</td>
<td>The VPG is recovering virtual machines in a VMware vCenter Server.</td>
</tr>
<tr>
<td>1/VCvApp</td>
<td>Deprecated. See VCDvApp.</td>
</tr>
<tr>
<td>2/VCDvApp</td>
<td>The VPG is recovering a VMware vCloud Director vApp.</td>
</tr>
<tr>
<td>3/PublicCloud</td>
<td>The VPG is recovering virtual machines to a Public Cloud.</td>
</tr>
<tr>
<td>4/HyperV</td>
<td>The VPG is recovering virtual machines in Microsoft Hyper-V.</td>
</tr>
<tr>
<td><strong>Source</strong></td>
<td>Deprecated. See Protected.</td>
</tr>
<tr>
<td><strong>Target</strong></td>
<td>Deprecated. See Recovery.</td>
</tr>
<tr>
<td><strong>FailSafeHistory</strong></td>
<td>Details of the journal’s healthy hours history.</td>
</tr>
<tr>
<td><strong>ActualFailSafeHistory</strong></td>
<td>The actual number of hours in a journal that were marked as healthy, in minutes.</td>
</tr>
<tr>
<td><strong>ConfiguredFailSafeHistory</strong></td>
<td>The configured number of hours that have been marked as healthy, in minutes. The default is 4 hours.</td>
</tr>
<tr>
<td><strong>FailSafeDescription</strong></td>
<td></td>
</tr>
<tr>
<td><strong>HistoryStatusApi</strong></td>
<td>Details of the journal’s history.</td>
</tr>
<tr>
<td><strong>ActualHistoryInMinutes</strong></td>
<td>The actual journal history, in minutes.</td>
</tr>
<tr>
<td><strong>ConfiguredHistoryInMinutes</strong></td>
<td>The configured journal history, in minutes. The default value is 4 hours.</td>
</tr>
<tr>
<td><strong>EarliestCheckpoint</strong></td>
<td>The earliest checkpoint in a journal.</td>
</tr>
<tr>
<td><strong>CheckpointIdentifier</strong></td>
<td>The identifier of the checkpoint of the VPG.</td>
</tr>
<tr>
<td><strong>Tag</strong></td>
<td>The tag when the checkpoint was added manually.</td>
</tr>
<tr>
<td><strong>TimeStamp</strong></td>
<td>The date and time the checkpoint was written to the journal. The value can be converted to an understandable date using code similar to the following:</td>
</tr>
<tr>
<td></td>
<td><code>var date = new Date(jsonDate);</code></td>
</tr>
<tr>
<td></td>
<td>or code similar to the Perl code example, <code>jsonDateToString($)</code> in “Perl Code Example: Retrieving the First 100 Records”, on page 193</td>
</tr>
<tr>
<td><strong>Vss</strong></td>
<td>Whether the checkpoint was added via the ZertoVssAgent or not.</td>
</tr>
<tr>
<td><strong>IOPS</strong></td>
<td>The IO per second between all the applications running on the virtual machines in the VPG and the VRAs that sends a copy to the remote site for replication.</td>
</tr>
<tr>
<td><strong>LastTest</strong></td>
<td>The date the last failover test occurred. The value can be converted to an understandable date using code similar to the following:</td>
</tr>
<tr>
<td></td>
<td><code>var date = new Date(jsonDate);</code></td>
</tr>
<tr>
<td></td>
<td>or code similar to the Perl code example, <code>jsonDateToString($)</code> in “Perl Code Example: Retrieving the First 100 Records”, on page 193</td>
</tr>
<tr>
<td><strong>Link</strong></td>
<td>The link details.</td>
</tr>
<tr>
<td><strong>href</strong></td>
<td>The URL used.</td>
</tr>
<tr>
<td><strong>identifier</strong></td>
<td>The VPG identifier.</td>
</tr>
<tr>
<td><strong>rel</strong></td>
<td>The next path level of the API relative to the current path.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>Link</td>
<td>The link details.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level of the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>OrganizationName</td>
<td>The name of the organization set up in the Zerto Cloud Manager that uses a cloud service provider for recovery.</td>
</tr>
<tr>
<td>Priority</td>
<td>The priority specified for the VPG. Possible values are (Json/XML):</td>
</tr>
<tr>
<td></td>
<td>0/Low: The VPG has a low priority for transferring data.</td>
</tr>
<tr>
<td></td>
<td>1/Medium: The VPG has a medium priority for transferring data.</td>
</tr>
<tr>
<td></td>
<td>2/High: The VPG has a high priority for transferring data.</td>
</tr>
<tr>
<td>ProgressPercentage</td>
<td>The percentage of an operation completed on the VPG, such as a bitmap sync or updating the VPG.</td>
</tr>
<tr>
<td>ProtectedSite</td>
<td>The protected site details.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used.</td>
</tr>
<tr>
<td>identifier</td>
<td>The site identifier.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level of the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>ProvisionedStorageInMB</td>
<td>The storage provisioned for the virtual machine in the recovery site.</td>
</tr>
<tr>
<td>RecoverySite</td>
<td>The recovery site details.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used.</td>
</tr>
<tr>
<td>identifier</td>
<td>The site identifier.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level of the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>ServiceProfile</td>
<td>The service profile details.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used.</td>
</tr>
<tr>
<td>identifier</td>
<td>The service profile identifier.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level of the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>ServiceProfileIdentifier</td>
<td>The identifier of the service profile to use for the VPG when a Zerto Cloud Manager is used.</td>
</tr>
<tr>
<td>ServiceProfileName</td>
<td>The service profile name.</td>
</tr>
<tr>
<td>SourceSite</td>
<td>Deprecated. See ProtectedSite &gt; identifier.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| Status    | The status of the VPG that contains the virtual machine. Possible values are (Json/XML):
|           | 0/Initializing: The VPG is being initialized. This includes when a VPG is created, and during the initial sync between sites. |
|           | 1/MeetingSLA: The VPG is meeting the SLA specification. |
|           | 2/NotMeetingSLA: The VPG is not meeting the SLA specification for both the journal history and RPO SLA settings, for example during a delta sync or when there is an error. |
|           | 3/RpoNotMeetingSLA: The VPG is not meeting the SLA specification for the RPO SLA setting. |
|           | 4/HistoryNotMeetingSLA: The VPG is not meeting the SLA specification for the journal history. |
|           | 5/FailingOver: The VPG is in a Failover operation. |
|           | 6/Moving: The VPG is in a Move operation. |
|           | 7/Deleting: The VPG is being deleted. |
|           | 8/Recovered: The VPG is recovered. |
| SubStatus | The substatus of the VPG that contains the virtual machine, for example the VPG is in a bitmap sync. For the description of substatuses, refer to the Zerto Virtual Manager Administration Guide. Possible values are (Json/XML):
<p>|           | 0/None |
|           | 1/InitialSync |
|           | 2/Creating |
|           | 3/VolumeInitialSync |
|           | 4/Sync |
|           | 5/RecoveryPossible |
|           | 6/DeltaSync |
|           | 7/NeedsConfiguration |
|           | 8/Error |
|           | 9/EmptyProtectionGroup |
|           | 10/DisconnectedFromPeerNoRecoveryPoints |
|           | 11/FullSync |
|           | 12/VolumeDeltaSync |
|           | 13/VolumeFullSync |
|           | 14/FailingOverCommitting |
|           | 15/FailingOverBeforeCommit |
|           | 16/FailingOverRollingBack |
|           | 17/Promoting |
|           | 18/MovingCommitting |
|           | 19/MovingBeforeCommit |
|           | 20/MovingRollingBack |
|           | 21/Deleting |
|           | 22/PendingRemove |
|           | 23/BitmapSync |
|           | 24/DisconnectedFromPeer |
|           | 25/ReplicationPausedUserInitiated |
|           | 26/ReplicationPausedSystemInitiated |
|           | 27/RecoveryStorageProfileError |
|           | 29/RollingBack |
|           | 30/RecoveryStorageError |
|           | 31/JournalStorageError |
|           | 32/VmNotProtectedError |
|           | 33/JournalOrRecoveryMissingError |
|           | 34/AddedVmsInInitialSync |
|           | 35/ReplicationPausedForMissingVolume |</p>
<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>TargetSite</td>
<td>Deprecated. See RecoverySite &gt; identifier.</td>
</tr>
<tr>
<td>ThroughputInMb</td>
<td>The MBs for all the applications running on the virtual machines being protected in the VPG.</td>
</tr>
<tr>
<td>UsedStoragelnMB</td>
<td>The storage used by the virtual machines in the VPG at the recovery site.</td>
</tr>
<tr>
<td>VmsCount</td>
<td>The number of virtual machines protected in the VPG.</td>
</tr>
<tr>
<td>VpgIdentifier</td>
<td>The internal VPG identifier.</td>
</tr>
<tr>
<td>VpgName</td>
<td>The VPG name.</td>
</tr>
<tr>
<td>Zorg</td>
<td>The ZORG details.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used.</td>
</tr>
<tr>
<td>identifier</td>
<td>The ZORG identifier.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level of the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
</tbody>
</table>

**Checkpoints** Response values for:
https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/checkpoints?startDate={STARTDATE}&endDate={ENDDATE}.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>checkpointIdentifier</td>
<td>The identifier of the checkpoint.</td>
</tr>
<tr>
<td>Tag</td>
<td>The tag when the checkpoint was added manually.</td>
</tr>
<tr>
<td>Timestamp</td>
<td>The date and time the checkpoint was written to the journal. The value can be converted to an understandable date using code similar to the following:</td>
</tr>
<tr>
<td></td>
<td>var date = new Date(jsonDate);</td>
</tr>
<tr>
<td></td>
<td>or code similar to the Perl code example, jsonDateToString($), in “Perl Code Example: Retrieving the First 100 Records”, on page 193.</td>
</tr>
<tr>
<td>Vss</td>
<td>Whether the checkpoint was added via the ZertoVssAgent or not.</td>
</tr>
</tbody>
</table>

**Checkpoints stats** Response values for:

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Earliest</td>
<td>Information related to the first checkpoint of a specific VPG in the recovery site.</td>
</tr>
<tr>
<td>CheckpointIden</td>
<td>The identifier of the first checkpoint.</td>
</tr>
<tr>
<td>Tag</td>
<td>The tag when the checkpoint was added.</td>
</tr>
<tr>
<td>TimeStamp</td>
<td>The date and time the checkpoint was written to the journal. The value can be converted to an understandable date using code similar to the following:</td>
</tr>
<tr>
<td></td>
<td>var date = new Date(jsonDate);</td>
</tr>
<tr>
<td></td>
<td>or code similar to the Perl code example, jsonDateToString($), in “Perl Code Example: Retrieving the First 100 Records”, on page 193.</td>
</tr>
<tr>
<td>Vss</td>
<td>Whether the checkpoint was added via the ZertoVssAgent or not.</td>
</tr>
<tr>
<td>LastCheckpoint</td>
<td>Information related to the last checkpoint of a specific VPG in the recovery site</td>
</tr>
<tr>
<td>CheckpointIden</td>
<td>The identifier of the last checkpoint.</td>
</tr>
<tr>
<td>Tag</td>
<td>The tag when the checkpoint was added.</td>
</tr>
</tbody>
</table>
PARAMETER | DESCRIPTION
---|---
TimeStamp | The date and time the checkpoint was written to the journal. The value can be converted to an understandable date using code similar to the following:
```
var date = new Date(jsonDate);
```
or code similar to the Perl code example, `jsonDateToString($)`, in “Perl Code Example: Retrieving the First 100 Records”, on page 193.

Vss | Whether the checkpoint was added via the ZertoVssAgent or not.


**RESPONSE: DESCRIPTION**

Possible entity types: the type of site for the VPG:

**VCVpg**: The VPG is recovering virtual machines in a VMware vCenter Server.

**VCvApp**: Deprecated. See **VCDvApp**

**VCDvApp**: The VPG is recovering a VMware vCloud Director vApp.

**PublicCloud**: The VPG is recovering virtual machines in Amazon Web Services (AWS).

**HyperV**: The VPG is recovering virtual machines in Microsoft Hyper-V.

**Failover commit policies** Response values for `https://zvm_ip:port/v1/vpgs/failovercommitpolicies`.

**RESPONSE: DESCRIPTION**

Possible policies used for a failover:

**Rollback**: After the seconds specified in the `commitValue` setting have elapsed, the failover is rolled back.

**Commit**: After the seconds specified in the `commitValue` setting have elapsed, the failover continues, committing the virtual machines in the recovery site.

**None**: The virtual machines in the VPG being failed over remain in the **Before Commit** state until either they are committed with **Commit a failover** or rolled back with **Roll back a failover**.

**Failover shutdown policies** Response values for `https://zvm_ip:port/v1/vpgs/failovershutdownpolicies`.

**RESPONSE: DESCRIPTION**

Possible shutdown polices when failing over a VPG:

**None**: The protected virtual machines are not shut down before the failover begins.

**Shutdown**: If VMware Tools or Microsoft Integration Services are available, the virtual machines are shut down gracefully. Otherwise, the failover operation fails.

**ForceShutdown**: The virtual machines are forcibly shut down even if they cannot be gracefully shut down.

**Priorities** Response values for `https://zvm_ip:port/v1/vpgs/priorities`.

**RESPONSE: DESCRIPTION**

Possible VPG priorities:

**Low**: The VPG has a low priority for transferring data.

**Medium**: The VPG has a medium priority for transferring data.

**High**: The VPG has a high priority for transferring data.


**RESPONSE: DESCRIPTION**

Possible VPG retention policies:

**StandardDR**: The VPG is defined for disaster recovery and not for additional long term recovery via offsite backup.

**ExtendedDR**: The VPG is defined for both disaster recovery and for additional long term recovery via offsite backup.
## Statuses

<table>
<thead>
<tr>
<th>RESPONSE: DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Possible statuses of a VPG:</strong></td>
</tr>
<tr>
<td><strong>Initializing</strong>: The VPG is being initialized. This includes when a VPG is created, and during the initial sync between sites.</td>
</tr>
<tr>
<td><strong>MeetingSLA</strong>: The VPG is meeting the SLA specification.</td>
</tr>
<tr>
<td><strong>NotMeetingSLA</strong>: The VPG is not meeting the SLA specification for both the journal history and RPO SLA settings, for example during a delta sync or when there is an error.</td>
</tr>
<tr>
<td><strong>RpoNotMeetingSLA</strong>: The VPG is not meeting the SLA specification for the RPO SLA setting.</td>
</tr>
<tr>
<td><strong>HistoryNotMeetingSLA</strong>: The VPG is not meeting the SLA specification for the journal history.</td>
</tr>
<tr>
<td><strong>FailingOver</strong>: The VPG is in a Failover operation.</td>
</tr>
<tr>
<td><strong>Moving</strong>: The VPG is in a Move operation.</td>
</tr>
<tr>
<td><strong>Deleting</strong>: The VPG is being deleted.</td>
</tr>
<tr>
<td><strong>Recovered</strong>: The VPG is recovered.</td>
</tr>
</tbody>
</table>

## Substatuses

<table>
<thead>
<tr>
<th>RESPONSE: DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Possible substatus of a VPG:</strong></td>
</tr>
<tr>
<td><strong>None</strong></td>
</tr>
<tr>
<td><strong>InitialSync</strong></td>
</tr>
<tr>
<td><strong>Creating</strong></td>
</tr>
<tr>
<td><strong>VolumenInitialSync</strong></td>
</tr>
<tr>
<td><strong>Sync</strong></td>
</tr>
<tr>
<td><strong>RecoveryPossible</strong></td>
</tr>
<tr>
<td><strong>DeltaSync</strong></td>
</tr>
<tr>
<td><strong>NeedsConfiguration</strong></td>
</tr>
<tr>
<td><strong>Error</strong></td>
</tr>
<tr>
<td><strong>EmptyProtectionGroup</strong></td>
</tr>
<tr>
<td><strong>DisconnectedFromPeerNoRecoveryPoints</strong></td>
</tr>
<tr>
<td><strong>FullSync</strong></td>
</tr>
<tr>
<td><strong>VolumeDeltaSync</strong></td>
</tr>
<tr>
<td><strong>VolumeFullSync</strong></td>
</tr>
<tr>
<td><strong>FailingOverCommitting</strong></td>
</tr>
<tr>
<td><strong>FailingOverBeforeCommit</strong></td>
</tr>
<tr>
<td><strong>FailingOverRollingBack</strong></td>
</tr>
<tr>
<td><strong>Promoting</strong></td>
</tr>
<tr>
<td><strong>MovingCommitting</strong></td>
</tr>
<tr>
<td><strong>MovingBeforeCommit</strong></td>
</tr>
<tr>
<td><strong>MovingRollingBack</strong></td>
</tr>
<tr>
<td><strong>Deleting</strong></td>
</tr>
<tr>
<td><strong>PendingRemove</strong></td>
</tr>
<tr>
<td><strong>BitmapSync</strong></td>
</tr>
<tr>
<td><strong>DisconnectedFromPeer</strong></td>
</tr>
<tr>
<td><strong>ReplicationPausedUserInitiated</strong></td>
</tr>
<tr>
<td><strong>ReplicationPausedSystemInitiated</strong></td>
</tr>
<tr>
<td><strong>RecoveryStorageProfileError</strong></td>
</tr>
<tr>
<td><strong>RollingBack</strong></td>
</tr>
<tr>
<td><strong>RecoveryStorageError</strong></td>
</tr>
<tr>
<td><strong>JournalStorageError</strong></td>
</tr>
<tr>
<td><strong>VmNotProtectedError</strong></td>
</tr>
<tr>
<td><strong>JournalOrRecoveryMissingError</strong></td>
</tr>
<tr>
<td><strong>AddedVmsInInitialSync</strong></td>
</tr>
<tr>
<td><strong>ReplicationPausedForMissingVolume</strong></td>
</tr>
</tbody>
</table>
VPGs: POST

Creates a VPG or performs actions on a specific VPG.

URL

Create VPG (VMware only) https://zvm_ip:port/v1/vpgs
Clone VPG https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/CloneStart
Abort VPG clone https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/CloneAbort
Failover VPG https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/Failover
Commit VPG failover https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/FailoverCommit
Rollback VPG failover https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/FailoverRollback
Test VPG https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/failoverTest
Stop VPG test https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/failoverTestStop
Force synchronize a VPG https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/forcesync
Pause VPG protection https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/pause
Resume VPG protection https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/resume

Where:

<table>
<thead>
<tr>
<th>zvm_ip</th>
<th>The IP address of the Zerto Virtual Manager where the API is run.</th>
</tr>
</thead>
<tbody>
<tr>
<td>port</td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
<tr>
<td>protectionGroupIdentifier</td>
<td>The identifier of the VPG for which an action is executed, such as a clone, failover or failover test.</td>
</tr>
</tbody>
</table>

Json Request Format

The following is an example request Json body for https://zvm_ip:port/v1/vpgs.

```json
{
    "DatastoreIdentifier": "String content",
    "OrgVdcIdentifier": "String content",
    "Priority":0,
    "ResourcePoolIdentifier": "String content",
    "ServiceProfileIdentifier": "String content",
    "SourceSiteIdentifier": "String content",
    "TargetSiteIdentifier": "String content",
    "VcdVappIdentifier": "String content",
    "VmsIdentifiers": [
        "String content",
        ...
    ],
    "VpgName": "String content",
    "ZorgIdentifier": "String content"
}
```

The following is an example request Json body for https://zvm_ip:port/v1/vpgs
The following is an example request Json body for https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/CloneStart.

```json
{
  "DatastoreIdentifier": "String content",
  "OrgVdcIdentifier": "String content",
  "Priority":0,
  "ResourcePoolIdentifier": "String content",
  "ServiceProfileIdentifier": "String content",
  "SourceSiteIdentifier": "String content",
  "TargetSiteIdentifier": "String content",
  ...
}
```

The following is an example request Json body for https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/Failover.

```json
{
  "CheckpointIdentifier": "String content",
  "CommitPolicy":1,
  "ShutdownPolicy":0,
  "TimeToWaitBeforeShutdownInSec":2147483647
  "IsReverseProtection": Boolean
}
```

The following is an example request Json body for https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/FailoverTest.

```json
{
  "CheckpointIdentifier": "String content"
}
```

The following is an example request Json body for https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/FailoverTestStop.

```json
{
  "FailoverTestSuccess": Boolean,
  "FailoverTestSummary": "String content"
}
```

The following is an example request Json body for https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/FailoverCommit

```json
{
  "IsReverseProtection": Boolean
}
```

The request bodies for the other VPGs POST APIs, such as https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/FailoverRollback, is empty.
**Request Values**

**VMware only: Create a VPG** Request values for [https://zvm_ip:port/v1/vpgs/{VpgIdentifier}](https://zvm_ip:port/v1/vpgs/{VpgIdentifier}).

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>datastoreIdentifier</td>
<td>Optional: The identifier of the datastore for the recovered virtual machine disks. Get the identifier using <a href="https://zerto.com">Datastores</a>. If a datastore is not specified, the datastore used is the datastore with the most free space, or the first datastore accessible by the ZORG, if specified. The identifier comprises the server identifier and the storage moref, with the format, serverid:moref.</td>
</tr>
<tr>
<td>OrgVdcIdentifier</td>
<td>Optional: The OrgvCD identifier when recovering to vCD. Get the identifier using <a href="https://zerto.com">VMware only: Resource Pools</a>.</td>
</tr>
<tr>
<td>Priority</td>
<td>Optional: The priority to assign to the VPG. Possible values are (Json/XML): 0/Low: The VPG has a low priority for transferring data. 1/Medium: The VPG has a medium priority for transferring data. 2/High: The VPG has a high priority for transferring data.</td>
</tr>
<tr>
<td>ServiceProfileIdentifier</td>
<td>Optional: The identifier of the service profile to use for the VPG when a Zerto Cloud Manager is used. Get the identifier using <a href="https://zerto.com">Service Profiles API</a>.</td>
</tr>
<tr>
<td>SourceSiteIdentifier</td>
<td>Optional: The identifier of the protected site. Get the identifier using <a href="https://zerto.com">Virtualization sites</a>. If a source site is not specified, the site where the API runs is used as the source site.</td>
</tr>
<tr>
<td>TargetSiteIdentifier</td>
<td>Optional: The identifier of the target site. Get the identifier using <a href="https://zerto.com">Virtualization sites</a>. If a target site is not specified, the target site is the same as the source site, and enabling replication to the same vCenter Server must be set in the Zerto user interface.</td>
</tr>
<tr>
<td>VcVappIdentifier</td>
<td>Deprecated. Deprecated. See VcdVappIdentifier.</td>
</tr>
<tr>
<td>VcdVappIdentifier</td>
<td>The identifier of the vCD vApp to protect. When protecting virtual machines in a vCenter Server, this value is null. Get the identifier using <a href="https://zerto.com">List Unprotected vCD vApps in a Site</a>.</td>
</tr>
<tr>
<td>VmsIdentifier</td>
<td>The identifiers of the virtual machines to protect. When protecting a vCenter Server vApp or a vCD vApp, this value is null. Get the identifiers using <a href="https://zerto.com">Unprotected VMs</a>.</td>
</tr>
<tr>
<td>VpgName</td>
<td>The name to identify the VPG. This name must be unique for a specific ZORG.</td>
</tr>
<tr>
<td>ZorgIdentifier</td>
<td>Optional: The identifier of the ZORG, Zerto organization, defined in the Zerto Cloud Manager. Get the identifier using <a href="https://zerto.com">ZOrgs API</a>.</td>
</tr>
</tbody>
</table>

**Note:** Only one of VcdVappIdentifier and VmsIdentifier is required. If a value is not specified in the request, a null value is used. Nulls are specified in Json as null, all lowercase. For example, the following Json object is to protect two virtual machines to a recovery site, with the site where the API is run being the protected site:

```
```

The above Json can be shortened to the following, as the values are defaulted to null when not specified:

```
{"TargetSite":"a9cc87d9-1a97-41c0-a91f-9ef306097d40","Vms":["598e5def-3500-4409-a691-d25b5cd10d22.vm-51","598e5def-3500-4409-a691-d25b5cd10d22.vm-50"],"VpgName":"Reports"}
```

**Clone a VPG** Request values for [https://zvm_ip:port/v1/vpgs/{VpgIdentifier}/CloneStart](https://zvm_ip:port/v1/vpgs/{VpgIdentifier}/CloneStart).

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>CheckpointId</td>
<td>Deprecated. See CheckpointIdentifier.</td>
</tr>
</tbody>
</table>
### Aborting a Clone

Request body for `https://zvm_ip:port/v1/vpgs/{VpgIdentifier}/CloneAbort` is empty.

### Failover a VPG

Request values for `https://zvm_ip:port/v1/vpgs/{VpgIdentifier}/failover`.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>CheckpointIdentifier</td>
<td>Optional: The identifier of the checkpoint to use for cloning. If a checkpoint is not specified, the latest checkpoint is used.</td>
</tr>
<tr>
<td>CommitPolicy</td>
<td>The policy to use after the failover enters a Before Commit state. Possible values are (Json/XML):</td>
</tr>
<tr>
<td></td>
<td>- Rollback: After the seconds specified in the commitValue setting have elapsed, the failover is rolled back.</td>
</tr>
<tr>
<td></td>
<td>- Commit: After the seconds specified in the commitValue setting have elapsed, the failover continue, committing the virtual machines in the recovery site.</td>
</tr>
<tr>
<td></td>
<td>- None: The virtual machines in the VPG being failed over remain in the Before Commit state until either they are committed with Commit a failover, or rolled back with Roll back a failover.</td>
</tr>
<tr>
<td>CommitValue</td>
<td>The amount of time in seconds the failover waits in a Before Commit state to enable checking that the failover is as required before performing the commit Policy setting.</td>
</tr>
<tr>
<td>ShutdownPolicy</td>
<td>The shutdown policy to apply to the virtual machines on the protected site that are being failed over Possible values are (Json/XML):</td>
</tr>
<tr>
<td></td>
<td>- 0 or None: The protected virtual machines are not touched before starting the failover. This assumes that you do not have access to the protected virtual machines.</td>
</tr>
<tr>
<td></td>
<td>- 1 or Shutdown: If the protected virtual machines have VMware Tools or Microsoft Integration Services available, the virtual machines are gracefully shut down, otherwise the failover operation fails. This is similar to performing a Move operation to a specified checkpoint.</td>
</tr>
<tr>
<td></td>
<td>- 2 or ForceShutdown: The protected virtual machines are forcibly shut down before starting the failover. If the protected virtual machines have VMware Tools or Microsoft Integration Services available, the procedure waits five minutes for the virtual machines to be gracefully shut down before forcibly powering them off. This is similar to performing a Move operation to a specified checkpoint.</td>
</tr>
<tr>
<td>TimeToWaitBeforeShutdownInSec</td>
<td>The amount of time in seconds the failover waits in a Before Commit state to enable checking that the failover is as required before performing the commit policy setting. The commit policy setting can either be commit or rollback.</td>
</tr>
<tr>
<td>IsReverseProtection</td>
<td>Optional. If IsReverseProtection is not set, the default value is False.</td>
</tr>
<tr>
<td></td>
<td>- 1/True - Enable reverse protection. The virtual machines are recovered on the recovery site and then protected using the default reverse protection settings.</td>
</tr>
<tr>
<td></td>
<td>- 0/False - Do not enable reverse protection. The VPG definition is kept with the status Needs Configuration and the reverse settings in the VPG definition are not set.</td>
</tr>
</tbody>
</table>

**Note:** Attempting to commit a failover during the initial stage, before it is ready to commit or roll back, throws an exception.
Roll back a failover Request body for `https://zvm_ip:port/v1/vpgs/{VpgIdentifier}/FailoverRollback` is empty.

**Note:** Attempting to rollback a failover during the initial stage, before it is ready to commit or roll back, throws an exception.

Test a failover Request values for `https://zvm_ip:port/v1/vpgs/{VpgIdentifier}/FailoverTest`.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>checkpointId</td>
<td>Deprecated. See CheckpointIdentifier.</td>
</tr>
<tr>
<td>CheckpointIdentifier</td>
<td>Optional: The identifier of the checkpoint to use for testing. If a checkpoint is not specified, the latest checkpoint is used.</td>
</tr>
</tbody>
</table>

Stopping a failover test Request values for `https://zvm_ip:port/v1/vpgs/{VpgIdentifier}/FailoverTestStop`.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>FailoverTestSuccess</td>
<td>Optional. The default is true. True: The test was successful. False: The test was not successful.</td>
</tr>
<tr>
<td>FailoverTestSummary</td>
<td>Optional: Free text describing the test.</td>
</tr>
</tbody>
</table>

Force synchronize a VPG Request body for `https://zvm_ip:port/v1/vpgs/{VpgIdentifier}/forcesync` is empty.

Pause VPG protection Request body for `https://zvm_ip:port/v1/vpgs/{VpgIdentifier}/pause` is empty.

Resume VPG protection Request body for `https://zvm_ip:port/v1/vpgs/{VpgIdentifier}/resume` is empty.

XML Request Format
For the XML request format, see “VPGs API POST Method Request and Response Formats”, on page 247.

Json Response Format

"String content"

XML Response Format
For the XML response format, see “VPGs API POST Method Request and Response Formats”, on page 247.

Response values
The response value is the task identifier that can be used with the Tasks API to monitor the delete action.
**VPGs: DELETE**

Delete a specified VPG, keeping the target disks to use for preseeding if the virtual machines are reprotected.

**URL**

Delete VPG  
https://zvm_ip:port/v1/vpgs/{VpgIdentifier}

Where:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>zvm_ip</td>
<td>The IP address of the Zerto Virtual Manager where the API is run.</td>
</tr>
<tr>
<td>port</td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
<tr>
<td>VpgIdentifier</td>
<td>The identifier of the VPG to be deleted.</td>
</tr>
</tbody>
</table>

**Json Request Format**

The following is an example request Json body for https://zvm_ip:port/v1/vpgs/{VpgIdentifier}.

```json
{
    "Force": Boolean,
    "KeepRecoveryVolumes": Boolean
}
```

**XML Request Format**

For the XML response format, see “VPGs API DELETE Method Request and Response Formats”, on page 249.

**Request Values**

Request values for https://zvm_ip:port/v1/vpgs/{VpgIdentifier}.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
</table>
| Force                   | True: Force deletion of the VPG.  
|                         | False: Do not force deletion of the VPG. This is the default if Force is not set. |
| KeepRecoveryVolumes     | True: Keep the recovery volumes. If the virtual machines in the deleted VPG are reprotected, these volumes can be used as preseeded volumes to speed up the initial synchronization of the new VPG.  
|                         | False: Do not keep the recovery volumes. |

**Json Response Format**

The following is an example response Json body for https://zvm_ip:port/v1/vpgs/{VpgIdentifier}.

"String content"

**Response Values**

The response value is the task identifier which can be used with the Tasks API to monitor the delete action.

**XML Response Format**

For the XML response format, see “VPGs API DELETE Method Request and Response Formats”, on page 249.

**Code Examples**

For complete code examples, see “Code Samples”, on page 22.
/v1/vpgs cURL Code Example

Retrieve the list of all VPGs.

```
```

Retrieve the list of all VPGs where the protected site is a vCenter Server.

```
```

Retrieve the list of checkpoints for a specified VPG with the identifier 8c3d56d-1289-48ba-a054-72447ee73821.

```
```

Perform a failover test on a specified VPG with the identifier a8c3d56d-1289-48ba-a054-72447ee73821, with a specified checkpoint as the request body.

```
```

Stop a failover test on a specified VPG with the identifier a8c3d56d-1289-48ba-a054-72447ee73821.

```
```

For more code examples, see “cURL Code”, on page 22.

/v1/vpgs Python Code Example

The following code samples are extracts from the fuller code example, “Python Code”, on page 23.

The following code sample retrieves a VPG identifier based on the VPG name.

```python
import requests
import base64

def GetVpgIdFromName(zvmIp, sessionId, vpgName):
    url = "https://" + zvmIp + ":9669/v1/vpgs?name=" + vpgName
    headers = {"x-zerto-session": sessionId, "content-type": "application/json"}
    r = requests.get(url, headers=headers, verify=False)
    if(r.status_code != requests.codes.ok):
        raise Exception("Call to ZVM failed")
    print r.status_code
    print r.text
    vpgData = json.loads(r.text)
    if(len(vpgData) == 0):
        raise Exception("VPG by name " + vpgName + " not found")
    if(len(vpgData) > 1):
        raise Exception("multiple vpgs named " + vpgName + " found")
    thisVpg = vpgData[0]
    res = thisVpg["VpgIdentifier"]
    print thisVpg
    print res
    return res
```
The following code sample retrieves the latest checkpoint for the VPG.

```python
import requests
import base64

def GetLatestCheckpointForVpg(zvmIp, sessionId, vpgId):
    url = "https://" + zvmIp + ":9669/v1/vpgs/" + vpgId + "/checkpoints"
    headers = {'x-zerto-session': sessionId, 'content-type': 'application/json'}
    r = requests.get(url, headers=headers, verify=False)
    print r.status_code
    if(r.status_code != requests.codes.ok):
        raise Exception("Failed GetLatestCheckpointForVpg")
    allcpData = json.loads(r.text)
    print str(len(allcpData))
    if(len(allcpData) == 0):
        raise Exception("no checkpoints found for vpg " + vpgId)
    sortedCps = sorted(allcpData, key=lambda x: x["TimeStamp"], reverse=True)
    latestCp = sortedCps[0]
    print latestCp
    return latestCp["CheckpointIdentifier"]
```

The following code starts a failover test for the VPG, using the VPG identifier and checkpoint retrieved by the previous code examples. The code returns the task identifier which can then be used to track the test.

```python
import requests
import base64

def StartFailoverTest(zvmIp, sessionId, vpgId, cpId):
    url = "https://" + zvmIp + ":9669/v1/vpgs/" + vpgId + "/failovertest"
    payload = {"CheckpointId": cpId}
    dataval = json.dumps(payload)
    headers = {'x-zerto-session': sessionId, 'content-type': 'application/json'}
    r = requests.post(url, data=dataval, headers=headers, verify=False)
    print r.status_code
    if(r.status_code != requests.codes.ok):
        raise Exception("Failed StartFailoverTest")
    taskId = json.loads(r.text)
    print "task id = " + taskId
    return taskId
```

The following code stops the failover test for the VPG.

```python
import requests
import base64

def StopFailoverTest(zvmIp, sessionId, vpgId, status):
    url = "https://" + zvmIp + ":9669/v1/vpgs/" + vpgId + "/failoverteststop"
    payload = {"FailoverTestSuccess": status, "FailoverTestSummary":"Automatically"}
    dataval = json.dumps(payload)
    headers = {'x-zerto-session': sessionId, 'content-type': 'application/json'}
    r = requests.post(url, data=dataval, headers=headers, verify=False)
    print r.status_code
    if(r.status_code != requests.codes.ok):
        raise Exception("Failed StopFailoverTest")
    taskId = json.loads(r.text)
    print "task id = " + taskId
    return taskId
```
# VPG Management API

**Note:** The `/v1/vpgSettings` API does not work in a vCloud Director environment.

 `/v1/vpgSettings` returns information about VPG settings, creates new VPGs, edits VPG settings, or deletes all or selected VPG settings for a VPG. The following API are available:

- “VPG Settings: GET”, on page 119
- “VPG Settings: POST”, on page 142
- “VPG Settings: PUT”, on page 155
- “VPG Settings: DELETE”, on page 175

For vCD VPG management APIs, go to “Managing vCD APIs”, on page 196.

<table>
<thead>
<tr>
<th>PURPOSE</th>
<th>METHOD</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Get settings for VPGs</td>
<td>GET</td>
<td><code>https://zvm_ip:port/v1/vpgSettings</code></td>
</tr>
<tr>
<td>Get settings for a VPG</td>
<td>GET</td>
<td><code>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}</code></td>
</tr>
<tr>
<td>Get VPG backup settings</td>
<td>GET</td>
<td><code>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/backup</code></td>
</tr>
<tr>
<td>Get the day a backup is scheduled</td>
<td>GET</td>
<td><code>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/backup/dayofweek</code></td>
</tr>
<tr>
<td>Get a VPG backup retention period</td>
<td>GET</td>
<td><code>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/backup/retentionperiod</code></td>
</tr>
<tr>
<td>Get a VPG backup schedule</td>
<td>GET</td>
<td><code>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/backup/schedule</code></td>
</tr>
<tr>
<td>Get VPG basic settings</td>
<td>GET</td>
<td><code>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/basic</code></td>
</tr>
<tr>
<td>Get VPG boot settings</td>
<td>GET</td>
<td><code>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/bootgroup</code></td>
</tr>
<tr>
<td>Get VPG journal settings</td>
<td>GET</td>
<td><code>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/journal</code></td>
</tr>
<tr>
<td>Get VPG network settings</td>
<td>GET</td>
<td><code>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/networks</code></td>
</tr>
<tr>
<td>Get the VPG priority</td>
<td>GET</td>
<td><code>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/priority</code></td>
</tr>
<tr>
<td>Get VPG recovery settings</td>
<td>GET</td>
<td><code>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/recovery</code></td>
</tr>
<tr>
<td>Get VPG script settings</td>
<td>GET</td>
<td><code>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/scripting</code></td>
</tr>
<tr>
<td>Get all VM settings for a VPG</td>
<td>GET</td>
<td><code>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms</code></td>
</tr>
<tr>
<td>Get VM settings for a VM in a VPG</td>
<td>GET</td>
<td><code>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}</code></td>
</tr>
<tr>
<td>Get all VM NIC settings</td>
<td>GET</td>
<td><code>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/nics</code></td>
</tr>
<tr>
<td>Get settings for a VM NIC</td>
<td>GET</td>
<td><code>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/nics/{nicIdentifier}</code></td>
</tr>
<tr>
<td>Get VM volume settings</td>
<td>GET</td>
<td><code>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/volumes</code></td>
</tr>
<tr>
<td>Get settings for a VM volume</td>
<td>GET</td>
<td><code>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/volumes/{volumeId}</code></td>
</tr>
<tr>
<td>Create new VPG settings object</td>
<td>POST</td>
<td><code>https://zvm_ip:port/v1/vpgSettings</code></td>
</tr>
<tr>
<td>Create VPG settings object for an existing VPG</td>
<td>POST</td>
<td><code>https://zvm_ip:port/v1/vpgSettings</code></td>
</tr>
<tr>
<td>PURPOSE</td>
<td>METHOD</td>
<td>URL</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>--------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>Commit a settings object</td>
<td>POST</td>
<td>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/commit</td>
</tr>
<tr>
<td>Add VMs to a settings object</td>
<td>POST</td>
<td>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms</td>
</tr>
<tr>
<td>Edit VPG settings object</td>
<td>PUT</td>
<td>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}</td>
</tr>
<tr>
<td>Edit backup settings object</td>
<td>PUT</td>
<td>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/backup</td>
</tr>
<tr>
<td>Edit basic settings object</td>
<td>PUT</td>
<td>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/basic</td>
</tr>
<tr>
<td>Edit boot settings object</td>
<td>PUT</td>
<td>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/bootgroup</td>
</tr>
<tr>
<td>Edit journal settings object</td>
<td>PUT</td>
<td>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/journal</td>
</tr>
<tr>
<td>Edit network settings object</td>
<td>PUT</td>
<td>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/networks</td>
</tr>
<tr>
<td>Edit recovery settings object</td>
<td>PUT</td>
<td>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/recovery</td>
</tr>
<tr>
<td>Edit script settings object</td>
<td>PUT</td>
<td>https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/scripting</td>
</tr>
</tbody>
</table>
| Edit VM settings object             | PUT    | https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmId}
| Edit NIC settings object            | PUT    | https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmId}
| Edit volume settings object         | PUT    | https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmId}
| Delete a settings object            | DELETE | https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}           |
| Delete backup settings object       | DELETE | https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/backup    |
| Delete basic settings object        | DELETE | https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/basic     |
| Delete boot settings object         | DELETE | https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/bootgroup |
| Delete journal settings object      | DELETE | https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/journal   |
| Delete network settings object      | DELETE | https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/networks  |
| Delete recovery settings object     | DELETE | https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/recovery  |
| Delete script settings object       | DELETE | https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/scripting |
| Delete VM settings from a VPG       | DELETE | https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmId}
| Delete NIC settings object          | DELETE | https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmId}

Zerto Virtual Replication RESTful API Reference Guide - Version 5.5
Zerto Virtual Replication APIs

VPG Management API
HTTP Methods
GET, POST, PUT, DELETE

Security
The API is exposed over HTTPS. Client code must use the x-zerto-session HTTP authorization header.

See Also
Starting a session: Session: POST
View the results of the create VPG operation: Events API
Return the datastore identifier: Datastores
Return the resource pool identifier: VMware only: Resource Pools
Return the source and target site identifiers: Virtualization sites
Return the vCenter vApp identifier to protect: Unprotected VMs
Return the list of unprotected virtual machine identifiers: Unprotected VMs
Retrieve details of the task being performed: Tasks API
Perform actions, such as testing, a VPG: VPGs API

Format
Json, XML

VPG Settings: GET
The GET methods are used to retrieve values in a VPG settings object.
See also:
- “VPG Settings: POST”, on page 142
- “VPG Settings: PUT”, on page 155
- “VPG Settings: DELETE”, on page 175

Get settings for VPGs
https://zvm_ip:port/v1/vpgSettings
Get settings for a VPG
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}
Get VPG backup settings
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/backup
Get the day a backup is scheduled
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/backup/dayofweek
Get a VPG backup retention period
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/backup/retentionperiod
Get a VPG backup schedule
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/backup/scheduler
Get VPG basic settings
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/basic
Get VPG boot settings
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/bootgroup
Get VPG journal settings
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/journal
Get VPG network settings
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/networks
Get the VPG priority
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/priority
Get VPG recovery settings
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/recovery
Get VPG script settings  https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/scripting

Get all VM settings for a VPG  https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms

Get VM settings for a VM in a VPG  https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}

Get all VM NIC settings  https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/nics

Get settings for a VM NIC  https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/nics/{nicIdentifier}

Get VM volume settings  https://zvm_ip:port/v1/vpgSettings/vpgSettingsIdentifier}/vms/{vmIdentifier}/volumes

Get settings for a VM volume  https://zvm_ip:port/v1/vpgSettings/vpgSettingsIdentifier}/vms/{vmIdentifier}/volumes/{volumeId}

Where:

<table>
<thead>
<tr>
<th>zvm_ip</th>
<th>The IP address of the Zerto Virtual Manager where the API is run.</th>
</tr>
</thead>
<tbody>
<tr>
<td>port</td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
<tr>
<td>vpgSettingsIdentifier</td>
<td>The identifier of the VPG settings object for which information is retrieved.</td>
</tr>
<tr>
<td>vmIdentifier</td>
<td>The identifier of the virtual machine for which settings are retrieved. The identifier comprises the server identifier and the virtual machine moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>nicIdentifier</td>
<td>The identifier of the NIC for which settings are retrieved.</td>
</tr>
<tr>
<td>volumeId</td>
<td>The identifier of the volume for which settings are retrieved.</td>
</tr>
</tbody>
</table>

**Request Format**

The request body is empty.

**Json Response Format**

The following is an example response Json body for https://zvm_ip:port/v1/vpgSettings and, without the array statement, for https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}.

```json
[
  {
    "Backup": {
      "RepositoryIdentifier": "String content",
      "RetentionPeriod": "String content",
      "Retry": {
        "IntervalInMinutes": 2147483647,
        "Number": 2147483647,
        "Retry": Boolean
      },
      "Scheduler": {
        "DayOfWeek": "String content",
        "SchedulerPeriod": "String content",
        "TimeOfDay": "String content"
      }
    },
    ...
  }
]```
"Basic": {
  "JournalHistoryInHours": 2147483647,
  "Name": "String content",
  "Priority": "String content",
  "ProtectedSiteIdentifier": "String content",
  "RecoverySiteIdentifier": "String content",
  "RpoInSeconds": 300,
  "ServiceProfileIdentifier": "String content",
  "TestIntervalInMinutes": 43200,
  "UseWanCompression": Boolean,
  "ZorgIdentifier": "String content"
},

"BootGroups": {
  "BootGroups": [{
    "BootDelayInSeconds": 0,
    "BootGroupIdentifier": "String content",
    "Name": "String content"
  }]
},

"Journal": {
  "DatastoreIdentifier": "String content",
  "Limitation": {
    "HardLimitInMB": 2147483647,
    "HardLimitInPercent": 2147483647,
    "WarningThresholdInMB": 2147483647,
    "WarningThresholdInPercent": 2147483647
  }
},

"Networks": {
  "Failover": {
    "Hypervisor": {
      "DefaultNetworkIdentifier": "String content"
    }
  },
  "VCD": null
  "FailoverTest": {
    "Hypervisor": {
      "DefaultNetworkIdentifier": "String content"
    }
  }
  "VCD": null
},

"Protected": {
  "VCD": null
},

"Recovery": {
  "DefaultDatastoreClusterIdentifier": "String content",
  "DefaultDatastoreIdentifier": "String content",
  "DefaultFolderIdentifier": "String content",
  "DefaultHostClusterIdentifier": "String content",
  "DefaultHostIdentifier": "String content",
  "ResourcePoolIdentifier": "String content"
},

"VCD": null
}
```json
"Scripting": {
    "PostBackup": {
        "Command": "String content",
        "Parameters": "String content",
        "TimeoutInSeconds": 2147483647
    },
    "PostRecovery": {
        "Command": "String content",
        "Parameters": "String content",
        "TimeoutInSeconds": 2147483647
    },
    "PreRecovery": {
        "Command": "String content",
        "Parameters": "String content",
        "TimeoutInSeconds": 2147483647
    }
},

"Vms": [{
    "BootGroupId": "String content",
    "Journal": {
        "DatastoreIdentifier": "String content",
        "Limitation": {
            "HardLimitInMB": 4294967295,
            "HardLimitInPercent": 4294967295,
            "WarningThresholdInMB": 4294967295,
            "WarningThresholdInPercent": 4294967295
        }
    }
}]
```
"Nics": [{
    "Failover": {
        "Hypervisor": {
            "DnsSuffix": "String content",
            "IpConfig": {
                "Gateway": "String content",
                "IsDhcp": Boolean,
                "PrimaryDns": "String content",
                "SecondaryDns": "String content",
                "StaticIp": "String content",
                "SubnetMask": "String content"
            },
            "NetworkIdentifier": "String content",
            "ShouldReplaceMacAddress": Boolean
        }
    },
    "FailoverTest": {
        "Hypervisor": {
            "DnsSuffix": "String content",
            "IpConfig": {
                "Gateway": "String content",
                "IsDhcp": Boolean,
                "PrimaryDns": "String content",
                "SecondaryDns": "String content",
                "StaticIp": "String content",
                "SubnetMask": "String content"
            },
            "NetworkIdentifier": "String content",
            "ShouldReplaceMacAddress": Boolean
        }
    }
},
"NicIdentifier": "String content"
}]

"Recovery": {
    "DatastoreClusterIdentifier": "String content",
    "DatastoreIdentifier": "String content",
    "FolderIdentifier": "String content",
    "HostClusterIdentifier": "String content",
    "HostIdentifier": "String content",
    "ResourcePoolIdentifier": "String content"
},

"VmIdentifier": "String content",
"Volumes": [{
    "Datastore": {
        "DatastoreClusterIdentifier": "String content",
        "DatastoreIdentifier": "String content",
        "IsThin": Boolean
    },
    "ExistingVolume": {
        "DatastoreIdentifier": "String content",
        "ExistedVmIdentifier": "String content",
        "Mode": "String content",
        "Path": "String content"
    },
    "IsSwap": Boolean,
    "VolumeIdentifier": "String content"
}]
}]}
The following is an example response JSON body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/backup.

```
{
  "RepositoryIdentifier": "String content",
  "RetentionPeriod": "String content",
  "Retry": {
    "IntervalInMinutes": 2147483647,
    "Number": 2147483647,
    "Retry": Boolean
  },
  "Scheduler": {
    "DayOfWeek": "String content",
    "SchedulerPeriod": "String content",
    "TimeOfDay": "String content"
  }
}
```

The following is an example response JSON body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/backup/dayofweek, and for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/backup/retentionperiod, and for

```
"String content"
```

The following is an example response JSON body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/basic.

```
{
  "JournalHistoryInHours": 2147483647,
  "Name": "String content",
  "Priority": "String content",
  "ProtectedSiteIdentifier": "String content",
  "RecoverySiteIdentifier": "String content",
  "RpoInSeconds": 4294967295,
  "ServiceProfileIdentifier": "String content",
  "TestIntervalInMinutes": 2147483647,
  "UseWanCompression": Boolean,
  "ZorgIdentifier": "String content"
}
```

The following is an example response JSON body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/bootgroup.

```
{
  "BootGroups": [{
    "BootDelayInSeconds": 4294967295,
    "BootGroupIdentifier": "String content",
    "Name": "String content"
  }]
}
```

The following is an example response JSON body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/journal.
The following is an example response Json body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/networks

```json
{
   "DatastoreIdentifier": "String content",
   "Limitation": {
      "HardLimitInMB": 2147483647,
      "HardLimitInPercent": 2147483647,
      "WarningThresholdInMB": 2147483647,
      "WarningThresholdInPercent": 2147483647
   }
}
```

The following is an example response Json body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/priority.

"String content"

The following is an example response Json body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/recovery.

```json
{
   "DefaultDatastoreClusterIdentifier": "String content",
   "DefaultDatastoreIdentifier": "String content",
   "DefaultFolderIdentifier": "String content",
   "DefaultHostClusterIdentifier": "String content",
   "DefaultHostIdentifier": "String content",
   "ResourcePoolIdentifier": "String content"
}
```

The following is an example response Json body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/scripting.
The following is an example response Json body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms.

```json
{
    "BootGroupIdentifier": "String content",
    "Journal": {
        "DatastoreIdentifier": "String content",
        "Limitation": {
            "HardLimitInMB":4294967295,
            "HardLimitInPercent":4294967295,
            "WarningThresholdInMB":4294967295,
            "WarningThresholdInPercent":4294967295
        }
    },
    "PostBackup": {
        "Command": "String content",
        "Parameters": "String content",
        "TimeoutInSeconds":2147483647
    },
    "PostRecovery": {
        "Command": "String content",
        "Parameters": "String content",
        "TimeoutInSeconds":2147483647
    },
    "PreRecovery": {
        "Command": "String content",
        "Parameters": "String content",
        "TimeoutInSeconds":2147483647
    }
}
```
"Nics": [{
  "Failover": {
    "Hypervisor": {
      "DnsSuffix": "String content",
      "IpConfig": {
        "Gateway": "String content",
        "IsDhcp": Boolean,
        "PrimaryDns": "String content",
        "SecondaryDns": "String content",
        "StaticIp": "String content",
        "SubnetMask": "String content"
      },
      "NetworkIdentifier": "String content",
      "ShouldReplaceMacAddress": Boolean
    }
  },
  "FailoverTest": {
    "Hypervisor": {
      "DnsSuffix": "String content",
      "IpConfig": {
        "Gateway": "String content",
        "IsDhcp": Boolean,
        "PrimaryDns": "String content",
        "SecondaryDns": "String content",
        "StaticIp": "String content",
        "SubnetMask": "String content"
      },
      "NetworkIdentifier": "String content",
      "ShouldReplaceMacAddress": Boolean
    }
  },
  "NicIdentifier": "String content"
}],
"Recovery": {
  "DatastoreClusterIdentifier": "String content",
  "DatastoreIdentifier": "String content",
  "FolderIdentifier": "String content",
  "HostClusterIdentifier": "String content",
  "HostIdentifier": "String content",
  "ResourcePoolIdentifier": "String content"
},
"VmIdentifier": "String content",
"Volumes": [{
  "Datastore": {
    "DatastoreClusterIdentifier": "String content",
    "DatastoreIdentifier": "String content",
    "IsThin": Boolean
  },
  "ExistingVolume": {
    "DatastoreIdentifier": "String content",
    "ExistedVmIdentifier": "String content",
    "Mode": "String content",
    "Path": "String content"
  },
  "IsSwap": Boolean,
  "VolumeIdentifier": "String content"
}]}
The following is an example response JSON body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}.

```json
{
    "BootGroupIdentifier": "String content",
    "Journal": {
        "DatastoreIdentifier": "String content",
        "Limitation": {
            "HardLimitInMB": 4294967295,
            "HardLimitInPercent": 4294967295,
            "WarningThresholdInMB": 4294967295,
            "WarningThresholdInPercent": 4294967295
        }
    },
    "NicIdentifier": "String content",
    "Nics": [{
        "Failover": {
            "Hypervisor": {
                "DnsSuffix": "String content",
                "IpConfig": {
                    "Gateway": "String content",
                    "IsDhcp": Boolean,
                    "PrimaryDns": "String content",
                    "SecondaryDns": "String content",
                    "StaticIp": "String content",
                    "SubnetMask": "String content"
                },
                "NetworkIdentifier": "String content",
                "ShouldReplaceMacAddress": Boolean
            }
        },
        "FailoverTest": {
            "Hypervisor": {
                "DnsSuffix": "String content",
                "IpConfig": {
                    "Gateway": "String content",
                    "IsDhcp": Boolean,
                    "PrimaryDns": "String content",
                    "SecondaryDns": "String content",
                    "StaticIp": "String content",
                    "SubnetMask": "String content"
                },
                "NetworkIdentifier": "String content",
                "ShouldReplaceMacAddress": Boolean
            }
        }
    }],
    "Recovery": {
        "DatastoreClusterIdentifier": "String content",
        "DatastoreIdentifier": "String content",
        "FolderIdentifier": "String content",
        "HostClusterIdentifier": "String content",
        "HostIdentifier": "String content",
        "ResourcePoolIdentifier": "String content"
    }
}
```
The following is an example response JSON body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/nics.

```json

{"VmIdentifier": "String content",
 "Volumes": [{
   "Datastore": {
      "DatastoreClusterIdentifier": "String content",
      "DatastoreIdentifier": "String content",
      "IsThin": Boolean
   },
   "ExistingVolume": {
      "DatastoreIdentifier": "String content",
      "ExistingVmIdentifier": "String content",
      "Mode": "String content",
      "Path": "String content"
   },
   "IsSwap": Boolean,
   "VolumeIdentifier": "String content"
 },
 ]
 [ {
   "Failover": {
      "Hypervisor": {
         "DnsSuffix": "String content",
         "IpConfig": {
            "Gateway": "String content",
            "IsDhcp": Boolean,
            "PrimaryDns": "String content",
            "SecondaryDns": "String content",
            "StaticIp": "String content",
            "SubnetMask": "String content"
         },
         "NetworkIdentifier": "String content",
         "ShouldReplaceMacAddress": Boolean
      }
   },
   "FailoverTest": {
      "Hypervisor": {
         "DnsSuffix": "String content",
         "IpConfig": {
            "Gateway": "String content",
            "IsDhcp": Boolean,
            "PrimaryDns": "String content",
            "SecondaryDns": "String content",
            "StaticIp": "String content",
            "SubnetMask": "String content"
         },
         "NetworkIdentifier": "String content",
         "ShouldReplaceMacAddress": Boolean
      }
   },
   "NicIdentifier": "String content"
 }
]
```

The following is an example response JSON body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/nics/{nicIdentifier}.

```json

```
The following is an example response Json body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/volumes.

```
{
  "Failover": {
    "Hypervisor": {
      "DnsSuffix": "String content",
      "IpConfig": {
        "Gateway": "String content",
        "IsDhcp": Boolean,
        "PrimaryDns": "String content",
        "SecondaryDns": "String content",
        "StaticIp": "String content",
        "SubnetMask": "String content"
      },
      "NetworkIdentifier": "String content",
      "ShouldReplaceMacAddress": Boolean
    }
  },
  "FailoverTest": {
    "Hypervisor": {
      "DnsSuffix": "String content",
      "IpConfig": {
        "Gateway": "String content",
        "IsDhcp": Boolean,
        "PrimaryDns": "String content",
        "SecondaryDns": "String content",
        "StaticIp": "String content",
        "SubnetMask": "String content"
      },
      "NetworkIdentifier": "String content",
      "ShouldReplaceMacAddress": Boolean
    }
  },
  "NicIdentifier": "String content"
}
```

The following is an example response Json body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/volumes/{volumeId}.

```
[
  {
    "Datastore": {
      "DatastoreClusterIdentifier": "String content",
      "DatastoreIdentifier": "String content",
      "IsThin": Boolean
    },
    "ExistingVolume": {
      "DatastoreIdentifier": "String content",
      "ExistedVmIdentifier": "String content",
      "Mode": "String content",
      "Path": "String content"
    },
    "IsSwap": Boolean,
    "VolumeIdentifier": "String content"
  }
]
```
Response Values


<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Backup</td>
<td>Information related to offsite backup.</td>
</tr>
<tr>
<td>RepositoryIdentifier</td>
<td>The identifier of the repository where offsite backups are written.</td>
</tr>
<tr>
<td>RetentionPeriod</td>
<td>The length of time to keep offsite backups, up to a maximum of 12 months. Over time, Zerto reduces the number of stored offsite backups to save space.</td>
</tr>
</tbody>
</table>

**VALID VALUES FOR RetentionPeriod**

<table>
<thead>
<tr>
<th>RetentionPeriod</th>
<th># OF BACKUPS SAVED WHEN RUN DAILY</th>
<th># OF BACKUPS SAVED WHEN RUN WEEKLY</th>
</tr>
</thead>
<tbody>
<tr>
<td>OneWeek</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>OneMonth</td>
<td>11</td>
<td>5</td>
</tr>
<tr>
<td>ThreeMonths</td>
<td>13</td>
<td>7</td>
</tr>
<tr>
<td>SixMonths</td>
<td>16</td>
<td>10</td>
</tr>
<tr>
<td>NineMonths</td>
<td>19</td>
<td>13</td>
</tr>
<tr>
<td>OneYear</td>
<td>22</td>
<td>16</td>
</tr>
</tbody>
</table>

The **SchedulerPeriod** parameter defines whether backups are created daily or weekly.

For an explanation of how Zerto reduces the number of offsite backups, see the **Zerto Virtual Replication Administration Guide**.

<table>
<thead>
<tr>
<th>Retry</th>
<th>Information about backup retries.</th>
</tr>
</thead>
<tbody>
<tr>
<td>IntervalInMinutes</td>
<td>How much time to wait, in minutes, after a backup job fails before running the backup job again.</td>
</tr>
<tr>
<td>Number</td>
<td>The number of retries that will be attempted.</td>
</tr>
<tr>
<td>Retry</td>
<td><strong>True</strong>: The offsite backup job is rerun automatically if it fails. <strong>False</strong>: The offsite backup job is not rerun automatically if it fails.</td>
</tr>
<tr>
<td>Scheduler</td>
<td>Offsite backup schedule settings.</td>
</tr>
<tr>
<td>DayOfWeek</td>
<td>The day of the week that the offsite backups are run. If the value of <strong>SchedulerPeriod</strong> is daily, the value in this parameter can be ignored.</td>
</tr>
<tr>
<td>SchedulerPeriod</td>
<td><strong>Daily</strong>: The offsite backups are run every day. <strong>Weekly</strong>: The offsite backups are run once a week.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>TimeOfDay</td>
<td>The time of day when offsite backup jobs are run. The time is based on a 24-hour clock.</td>
</tr>
<tr>
<td>Basic</td>
<td>Basic VPG settings.</td>
</tr>
<tr>
<td>JournalHistoryInHours</td>
<td>The time that all write commands are saved in the journal. The value is between 1 and 336 (14 days).</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the VPG.</td>
</tr>
<tr>
<td>Priority</td>
<td>The priority specified for the VPG. Possible values are:</td>
</tr>
<tr>
<td></td>
<td><strong>Low</strong>: The VPG has a low priority for transferring data.</td>
</tr>
<tr>
<td></td>
<td><strong>Medium</strong>: The VPG has a medium priority for transferring data.</td>
</tr>
<tr>
<td></td>
<td><strong>High</strong>: The VPG has a high priority for transferring data.</td>
</tr>
<tr>
<td>ProtectedSiteIdentifier</td>
<td>The identifier of the source site where the VPG virtual machines are protected.</td>
</tr>
<tr>
<td>RecoverySiteIdentifier</td>
<td>The identifier of the target site where the VPG virtual machines are recovered.</td>
</tr>
<tr>
<td>RpoInSeconds</td>
<td>The maximum desired time between each automatic checkpoint being written to the journal before an alert is issued.</td>
</tr>
<tr>
<td>ServiceProfileIdentifier</td>
<td>The identifier of the service profile to use for the VPG when a Zerto Cloud Manager is used.</td>
</tr>
<tr>
<td>TestIntervalInMinutes</td>
<td>The time, in minutes, recommended between testing the integrity of the VPG. A warning is issued if a test is not done within this time frame. Possible values are:</td>
</tr>
<tr>
<td></td>
<td><strong>0</strong>: No testing is expected.</td>
</tr>
<tr>
<td></td>
<td><strong>43200</strong> or <strong>null</strong>: Testing is expected monthly.</td>
</tr>
<tr>
<td></td>
<td><strong>131040</strong>: Testing is expected every three months.</td>
</tr>
<tr>
<td></td>
<td><strong>262080</strong>: Testing is expected every six months.</td>
</tr>
<tr>
<td></td>
<td><strong>394560</strong>: Testing is expected every nine months.</td>
</tr>
<tr>
<td></td>
<td><strong>525600</strong>: Testing is expected every twelve months.</td>
</tr>
<tr>
<td>UseWanCompression</td>
<td><strong>True</strong>: Data is compressed before sending it to the recovery site. <strong>False</strong>: Data is not compressed before sending it to the recovery site.</td>
</tr>
<tr>
<td>ZorgIdentifier</td>
<td>The identifier of the ZORG, Zerto organization, defined in the Zerto Cloud Manager.</td>
</tr>
<tr>
<td>BootGroups</td>
<td>Information about boot groups.</td>
</tr>
<tr>
<td>BootGroups</td>
<td>Information about boot groups.</td>
</tr>
<tr>
<td>BootDelayInSeconds</td>
<td>Specifies the delay, in seconds, between starting up the virtual machines in this group and starting up the virtual machines in the next group.</td>
</tr>
<tr>
<td>BootGroupIdentifier</td>
<td>The identifier of a boot group.</td>
</tr>
<tr>
<td>Name</td>
<td>The name of a boot group.</td>
</tr>
<tr>
<td>Journal</td>
<td>Information about the journal.</td>
</tr>
<tr>
<td>DatastoreIdentifier</td>
<td>The identifier of the storage used by the journal. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>Limitation</td>
<td>Information about the journal limitations.</td>
</tr>
<tr>
<td>HardLimitInMB</td>
<td>The maximum journal size in MBs. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>HardLimitInPercent</td>
<td>The percentage of the virtual machine volume size the journal can grow to. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>WarningThresholdInMB</td>
<td>The journal size, in MBs, that generates a warning that the journal is nearing its hard limit. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>WarningThresholdInPercent</td>
<td>The percentage of the virtual machine volume size that generates a warning. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>Networks</td>
<td>Information about the networks that connect the protected and recovery sites.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------------------------</td>
<td>--------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Failover</td>
<td>Information about the networks used for failover.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td></td>
</tr>
<tr>
<td>DefaultNetworkIdentifier</td>
<td>The network identifier of the network to use during a failover or move operation in which the recovered virtual machines will run.</td>
</tr>
<tr>
<td>FailoverTest</td>
<td>Information about the networks used for testing failover.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td></td>
</tr>
<tr>
<td>DefaultNetworkIdentifier</td>
<td>The network identifier of the network to use when testing the failover of virtual machines in the recovery site.</td>
</tr>
<tr>
<td>Recovery</td>
<td>Information about the recovery site.</td>
</tr>
<tr>
<td>DefaultDatastoreClusterIdentifier</td>
<td>The identifier of the default datastore cluster used in the recovery site.</td>
</tr>
<tr>
<td></td>
<td>Note: Only when the recovery site is a vSphere site.</td>
</tr>
<tr>
<td>DefaultDatastoreIdentifier</td>
<td>The identifier of the default storage where the metadata files for the virtual machines are stored, such as the vmx or vhdx files. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultFolderIdentifier</td>
<td>The identifier of the default folder used for recovery. The identifier comprises the server identifier and the folder moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultHostClusterIdentifier</td>
<td>The identifier of the default host cluster that handles the replicated data. The identifier comprises the server identifier and the host cluster moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultHostIdentifier</td>
<td>The identifier of the default host that handles the replicated data. The identifier comprises the server identifier and the host moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>ResourcePoolIdentifier</td>
<td>The identifier of the resource pool for the recovered virtual machines.</td>
</tr>
<tr>
<td>Scripting</td>
<td>Information about the scripts to run, either before or after recovery operation, or after an offsite backup is run.</td>
</tr>
<tr>
<td>PostBackup</td>
<td>Information about scripts that are run after an offsite backup is performed.</td>
</tr>
<tr>
<td>Command</td>
<td>The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site.</td>
</tr>
<tr>
<td>Parameters</td>
<td>Parameters to pass to the script.</td>
</tr>
<tr>
<td>TimeoutInSeconds</td>
<td>The time-out, in seconds, for the script to run.</td>
</tr>
<tr>
<td>PostRecovery</td>
<td>Information about scripts that are run after a recovery operation is performed.</td>
</tr>
<tr>
<td>Command</td>
<td>The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site.</td>
</tr>
<tr>
<td>Parameters</td>
<td>Parameters to pass to the script.</td>
</tr>
<tr>
<td>TimeoutInSeconds</td>
<td>The time-out, in seconds, for the script to run.</td>
</tr>
<tr>
<td>PreRecovery</td>
<td>Information about scripts that are run before a recovery operation is performed.</td>
</tr>
<tr>
<td>Command</td>
<td>The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site.</td>
</tr>
<tr>
<td>Parameters</td>
<td>Parameters to pass to the script.</td>
</tr>
<tr>
<td>TimeoutInSeconds</td>
<td>The time-out, in seconds, for the script to run.</td>
</tr>
<tr>
<td>Vms</td>
<td>Information about the virtual machines in a VPG.</td>
</tr>
<tr>
<td>BootGroupIdentifier</td>
<td>The boot group identifier of a virtual machine.</td>
</tr>
<tr>
<td>Journal</td>
<td>Journal information of the virtual machine.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>DatastoreIdentifier</td>
<td>The identifier of the storage used by the journal for the virtual machine. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>Limitation</td>
<td>Information about the journal limitations.</td>
</tr>
<tr>
<td>HardLimitInMB</td>
<td>The maximum journal size in MBs. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>HardLimitInPercent</td>
<td>The percentage of the virtual machine volume size the journal can grow to. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>WarningThresholdInMB</td>
<td>The journal size, in MBs, that generates a warning that the journal is nearing its hard limit. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>WarningThresholdInPercent</td>
<td>The percentage of the virtual machine volume size that generates a warning. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>Nics</td>
<td>Information about NICs used by the virtual machine in the VPG.</td>
</tr>
<tr>
<td>Failover</td>
<td>Information about the networks used for failover by this virtual machine.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td></td>
</tr>
<tr>
<td>DnsSuffix</td>
<td>The DNS name excluding the host.</td>
</tr>
<tr>
<td>IpConfig</td>
<td>Information about the IP configuration of the recovery site used for failovers.</td>
</tr>
<tr>
<td>Gateway</td>
<td>The mask for the network.</td>
</tr>
<tr>
<td>IsDhcp</td>
<td>True: DHCP is used for the virtual machine on the recovery site. False: A static IP address is used for the virtual machine on the recovery site.</td>
</tr>
<tr>
<td>PrimaryDns</td>
<td>The IP address of the primary DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>SecondaryDns</td>
<td>The IP address of the alternate, secondary, DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>StaticIp</td>
<td>The IP of the restored virtual machine.</td>
</tr>
<tr>
<td>SubnetMask</td>
<td>The subnet mask for the network.</td>
</tr>
<tr>
<td>NetworkIdentifier</td>
<td>The network identifier of the network used for failover or move by this virtual machine.</td>
</tr>
<tr>
<td>ShouldReplaceMacAddress</td>
<td>True: The Media Access Control (MAC) address used on the protected site is copied to the recovery site. False: The Media Access Control (MAC) address used on the protected site is not copied to the recovery site.</td>
</tr>
<tr>
<td>FailoverTest</td>
<td>Information about the networks used for testing failover by this virtual machine.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td></td>
</tr>
<tr>
<td>DnsSuffix</td>
<td>The DNS name excluding the host.</td>
</tr>
<tr>
<td>IpConfig</td>
<td>Information about the IP configuration of the recovery site used for testing failovers.</td>
</tr>
<tr>
<td>Gateway</td>
<td>The mask for the network.</td>
</tr>
<tr>
<td>IsDhcp</td>
<td>True: DHCP is used for the virtual machine on the recovery site. False: A static IP address is used for the virtual machine on the recovery site.</td>
</tr>
<tr>
<td>PrimaryDns</td>
<td>The IP address of the primary DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>SecondaryDns</td>
<td>The IP address of the alternate, secondary, DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>StaticIp</td>
<td>The IP of the restored virtual machine.</td>
</tr>
<tr>
<td>SubnetMask</td>
<td>The subnet mask for the network.</td>
</tr>
<tr>
<td>NetworkIdentifier</td>
<td>The network identifier of the network used for testing failover by this virtual machine.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| ShouldReplaceMacAddress          | **True**: The Media Access Control (MAC) address used on the protected site is copied to the recovery site.  
                              | **False**: The Media Access Control (MAC) address used on the protected site is not copied to the recovery site.                          |
| NicIdentifier                    | The identifier of the NIC for which settings are returned.                                                                                   |
| Recovery                          | Information about the recovery site.                                                                                                          |
| DatastoreClusterIdentifier       | The identifier of the datastore cluster used in the recovery site for the VM.  
                              | Note: Only when the recovery site is a vSphere site.                                                                                         |
| DatastoreIdentifier              | The identifier of the storage. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.         |
| FolderIdentifier                 | The identifier of the folder used for recovery by the virtual machine. The identifier comprises the server identifier and the folder moref, with the format, serverid.moref. |
| HostClusterIdentifier            | The identifier of the host cluster that handles the replicated data. The identifier comprises the server identifier and the host cluster moref, with the format, serverid.moref. |
| HostIdentifier                   | The identifier of the host that handles the replicated data. The identifier comprises the server identifier and the host moref, with the format, serverid.moref. |
| ResourcePoolIdentifier           | The identifier of the resource pool for the recovered virtual machines. The identifier comprises the server identifier and the resource pool moref, with the format, serverid.moref. |
| VmIdentifier                     | The identifier of the virtual machine. The identifier comprises the server identifier and the virtual machine moref, with the format, serverid.moref. |
| Volumes                          | Information about the volumes used by the virtual machine.                                                                                   |
| Datastore                        | Information about the datastore used by the virtual machine.                                                                                   |
| DatastoreClusterIdentifier       | The identifier of the datastore cluster used in the recovery site for the volume.  
                              | Note: Only when the recovery site is a vSphere site.                                                                                         |
| DatastoreIdentifier              | The identifier of the storage. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.         |
| IsThin                           | **True**: The recovery volumes are thin-provisioned.  
                              | **False**: The recovery volumes are not thin-provisioned.                                                                                   |
| ExistingVolume                   | For future use.                                                                                                                             |
| DatstoreIdentifier               | For future use.                                                                                                                             |
| ExistedVmIdentifier              | For future use.                                                                                                                             |
| Mode                             | For future use.                                                                                                                             |
| Path                             | For future use.                                                                                                                             |
| IsSwap                           | **True**: The recovery disk is marked as a temp data disk.  
                              | **False**: The recovery disk is not marked as a temp data disk.                                                                             |
| VolumeIdentifier                 | The identifier of the volume.                                                                                                               |
| VpgIdentifier                    | The identifier of the VPG.                                                                                                                  |
| VpgSettingsIdentifier            | The identifier of the VPG settings object.                                                                                                    |

Backup day of week Response values for

RESPONSE: DESCRIPTION
The day of the week that the offsite backups are run.

Backup retention period Response values for

RESPONSE: DESCRIPTION
The amount of time the offsite backups are kept.

Backup scheduler period Response values for

RESPONSE: DESCRIPTION
The schedule for running offsite backups:

Daily: The offsite backups are run every day.
Weekly: The offsite backups are run once a week.
Basic Response values for `https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/basic`.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>JournalHistoryInHours</td>
<td>The time that all write commands are saved in the journal. The value is between 1 and 336 (14 days).</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the VPG.</td>
</tr>
</tbody>
</table>
| Priority          | The priority specified for the VPG. Possible values are:  
|                   | Low: The VPG has a low priority for transferring data.  
|                   | Medium: The VPG has a medium priority for transferring data.  
|                   | High: The VPG has a high priority for transferring data. |
| ProtectedSiteIdentifier | The identifier of the source site where the VPG virtual machines are protected. |
| RecoverySiteIdentifier | The identifier of the target site where the VPG virtual machines are recovered. |
| RpoInSeconds      | The maximum desired time between each automatic checkpoint being written to the journal before an alert is issued. |
| ServiceProfileIdentifier | The identifier of the service profile to use for the VPG when a Zerto Cloud Manager is used. |
| TestIntervalInMinutes | The time, in minutes, recommended between testing the integrity of the VPG. A warning is issued if a test is not done within this time frame. Possible values are:  
|                   | 0: No testing is expected.  
|                   | 43200 or null: Testing is expected monthly.  
|                   | 131040: Testing is expected every three months.  
|                   | 262080: Testing is expected every six months.  
|                   | 394560: Testing is expected every nine months.  
|                   | 525600: Testing is expected every twelve months. |
| UseWanCompression | True: Data is compressed before sending it to the recovery site.  
|                   | False: Data is not compressed before sending it to the recovery site. |
| ZorgIdentifier    | The identifier of the ZORG, Zerto organization, defined in the Zerto Cloud Manager. |

Boot group Response values for `https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/bootgroup`.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>BootGroups</td>
<td>Information about boot groups.</td>
</tr>
<tr>
<td>BootDelayInSeconds</td>
<td>Specifies the delay, in seconds, between starting up the virtual machines in this group and starting up the virtual machines in the next group.</td>
</tr>
<tr>
<td>BootGroupIdentifier</td>
<td>The identifier of a boot group.</td>
</tr>
<tr>
<td>Name</td>
<td>The name of a boot group.</td>
</tr>
</tbody>
</table>


<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>DatastoreIdentifier</td>
<td>The identifier of the storage used by the journal. The identifier comprises the server identifier and the storage moref, with the format, <code>serverid.moref</code>.</td>
</tr>
<tr>
<td>Limitation</td>
<td>Information about the journal limitations.</td>
</tr>
<tr>
<td>HardLimitInMB</td>
<td>The maximum journal size in MBs. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>HardLimitInPercent</td>
<td>The percentage of the virtual machine volume size the journal can grow to. 0 means unlimited. Integer values.</td>
</tr>
</tbody>
</table>
### VPG Management API

#### Zerto Virtual Replication RESTful API Reference Guide - Version 5.5

**Zerto Virtual Replication APIs**

**Network Response values for** `https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/networks`

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>WarningThresholdInMB</td>
<td>The journal size, in MBs, that generates a warning that the journal is nearing its hard limit. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>WarningThresholdInPercent</td>
<td>The percentage of the virtual machine volume size that generates a warning. 0 means unlimited.</td>
</tr>
</tbody>
</table>

**Priority Response values for** `https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/priority`

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>DefaultNetworkIdentifier</td>
<td>The network identifier of the network to use during a failover or move operation in which the recovered virtual machines will run.</td>
</tr>
<tr>
<td>DefaultNetworkIdentifier</td>
<td>The network identifier of the network to use when testing the failover of virtual machines in the recovery site.</td>
</tr>
</tbody>
</table>

**Recovery Response values for** `https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/recovery`

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>DefaultDatastoreClusterIdentifier</td>
<td>The identifier of the default datastore cluster used in the recovery site. Note: Only when the recovery site is a vSphere site.</td>
</tr>
<tr>
<td>DefaultDatastoreIdentifier</td>
<td>The identifier of the default storage where the metadata files for the virtual machines are stored, such as the vmx or vhdx files. The identifier comprises the server identifier and the storage moref, with the format, <code>serverid.moref</code>.</td>
</tr>
<tr>
<td>DefaultFolderIdentifier</td>
<td>The identifier of the default folder used for recovery. The identifier comprises the server identifier and the folder moref, with the format, <code>serverid.moref</code>.</td>
</tr>
<tr>
<td>DefaultHostClusterIdentifier</td>
<td>The identifier of the default host cluster that handles the replicated data. The identifier comprises the server identifier and the host cluster moref, with the format, <code>serverid.moref</code>.</td>
</tr>
<tr>
<td>DefaultHostIdentifier</td>
<td>The identifier of the default host that handles the replicated data. The identifier comprises the server identifier and the host moref, with the format, <code>serverid.moref</code>.</td>
</tr>
<tr>
<td>ResourcePoolIdentifier</td>
<td>The identifier of the resource pool for the recovered virtual machines. The identifier comprises the server identifier and the resource pool moref, with the format, <code>serverid.moref</code>.</td>
</tr>
</tbody>
</table>

**Scripting Response values for** `https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/scripting`
<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>INFORMATION ABOUT THE SCRIPTS TO RUN, EITHER BEFORE OR AFTER RECOVERY OPERATION, OR AFTER AN OFFSITE BACKUP IS RUN.</th>
</tr>
</thead>
<tbody>
<tr>
<td>PostBackup</td>
<td>Information about scripts that are run after an offsite backup is performed.</td>
</tr>
<tr>
<td>Command</td>
<td>The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site.</td>
</tr>
<tr>
<td>Parameters</td>
<td>Parameters to pass to the script.</td>
</tr>
<tr>
<td>TimeoutInSeconds</td>
<td>The time-out, in seconds, for the script to run.</td>
</tr>
<tr>
<td>PostRecovery</td>
<td>Information about scripts that are run after a recovery operation is performed.</td>
</tr>
<tr>
<td>Command</td>
<td>The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site.</td>
</tr>
<tr>
<td>Parameters</td>
<td>Parameters to pass to the script.</td>
</tr>
<tr>
<td>TimeoutInSeconds</td>
<td>The time-out, in seconds, for the script to run.</td>
</tr>
<tr>
<td>PreRecovery</td>
<td>Information about scripts that are run before a recovery operation is performed.</td>
</tr>
<tr>
<td>Command</td>
<td>The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site.</td>
</tr>
<tr>
<td>Parameters</td>
<td>Parameters to pass to the script.</td>
</tr>
<tr>
<td>TimeoutInSeconds</td>
<td>The time-out, in seconds, for the script to run.</td>
</tr>
</tbody>
</table>


<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>BootGroupIdentifier</td>
<td>The boot group identifier of a virtual machine.</td>
</tr>
<tr>
<td>Journal</td>
<td>Journal information of the virtual machine.</td>
</tr>
<tr>
<td>DatastoreIdentifier</td>
<td>The identifier of the storage used by the journal for the virtual machine. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>Limitation</td>
<td>Information about the journal limitations.</td>
</tr>
<tr>
<td>HardLimitInMB</td>
<td>The maximum journal size in MBs. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>HardLimitInPercent</td>
<td>The percentage of the virtual machine volume size the journal can grow to. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>WarningThresholdInMB</td>
<td>The journal size, in MBs, that generates a warning that the journal is nearing its hard limit. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>WarningThresholdInPercent</td>
<td>The percentage of the virtual machine volume size that generates a warning. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>Nics</td>
<td>Information about NICs used by the virtual machine in the VPG.</td>
</tr>
<tr>
<td>Failover</td>
<td>Information about the networks used for failover by this virtual machine.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td></td>
</tr>
<tr>
<td>DnsSuffix</td>
<td>The DNS name excluding the host.</td>
</tr>
<tr>
<td>IpConfig</td>
<td>Information about the IP configuration of the recovery site used for failovers.</td>
</tr>
<tr>
<td>Gateway</td>
<td>The mask for the network.</td>
</tr>
<tr>
<td>IsDhcp</td>
<td>True: DHCP is used for the virtual machine on the recovery site.</td>
</tr>
<tr>
<td></td>
<td>False: A static IP address is used for the virtual machine on the recovery site.</td>
</tr>
<tr>
<td>PrimaryDns</td>
<td>The IP address of the primary DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>SecondaryDns</td>
<td>The IP address of the alternate, secondary, DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>StaticIp</strong></td>
<td>The IP of the restored virtual machine.</td>
</tr>
<tr>
<td><strong>SubnetMask</strong></td>
<td>The subnet mask for the network.</td>
</tr>
<tr>
<td><strong>NetworkIdentifier</strong></td>
<td>The network identifier of the network used for failover or move by this virtual machine.</td>
</tr>
<tr>
<td><strong>ShouldReplaceMacAddress</strong></td>
<td><strong>True</strong>: The Media Access Control (MAC) address used on the protected site is copied to the recovery site.</td>
</tr>
<tr>
<td></td>
<td><strong>False</strong>: The Media Access Control (MAC) address used on the protected site is not copied to the recovery site.</td>
</tr>
<tr>
<td><strong>FailoverTest</strong></td>
<td><strong>True</strong>: The Media Access Control (MAC) address used on the protected site is copied to the recovery site.</td>
</tr>
<tr>
<td></td>
<td><strong>False</strong>: The Media Access Control (MAC) address used on the protected site is not copied to the recovery site.</td>
</tr>
<tr>
<td><strong>Hypervisor</strong></td>
<td></td>
</tr>
<tr>
<td><strong>DnsSuffix</strong></td>
<td>The DNS name excluding the host.</td>
</tr>
<tr>
<td><strong>IpConfig</strong></td>
<td>Information about the IP configuration of the recovery site used for testing failovers.</td>
</tr>
<tr>
<td><strong>Gateway</strong></td>
<td>The mask for the network.</td>
</tr>
<tr>
<td><strong>IsDhcp</strong></td>
<td><strong>True</strong>: DHCP is used for the virtual machine on the recovery site.</td>
</tr>
<tr>
<td></td>
<td><strong>False</strong>: A static IP address is used for the virtual machine on the recovery site.</td>
</tr>
<tr>
<td><strong>PrimaryDns</strong></td>
<td>The IP address of the primary DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td><strong>SecondaryDns</strong></td>
<td>The IP address of the alternate, secondary, DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td><strong>StaticIp</strong></td>
<td>The IP of the restored virtual machine.</td>
</tr>
<tr>
<td><strong>SubnetMask</strong></td>
<td>The subnet mask for the network.</td>
</tr>
<tr>
<td><strong>NetworkIdentifier</strong></td>
<td>The network identifier of the network used for testing failover by this virtual machine.</td>
</tr>
<tr>
<td><strong>ShouldReplaceMacAddress</strong></td>
<td><strong>True</strong>: The Media Access Control (MAC) address used on the protected site is copied to the recovery site.</td>
</tr>
<tr>
<td></td>
<td><strong>False</strong>: The Media Access Control (MAC) address used on the protected site is not copied to the recovery site.</td>
</tr>
<tr>
<td><strong>NicIdentifier</strong></td>
<td>The identifier of the NIC for which settings are returned.</td>
</tr>
<tr>
<td><strong>Recovery</strong></td>
<td>Information about the recovery site.</td>
</tr>
<tr>
<td><strong>DatastoreClusterIdentifier</strong></td>
<td>The identifier of the datastore cluster used in the recovery site for the VM.</td>
</tr>
<tr>
<td></td>
<td>Note: Only when the recovery site is a vSphere site.</td>
</tr>
<tr>
<td><strong>DatastoreIdentifier</strong></td>
<td>The identifier of the storage. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td><strong>FolderIdentifier</strong></td>
<td>The identifier of the folder used for recovery by the virtual machine. The identifier comprises the server identifier and the folder moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td><strong>HostClusterIdentifier</strong></td>
<td>The identifier of the host cluster that handles the replicated data. The identifier comprises the server identifier and the host cluster moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td><strong>HostIdentifier</strong></td>
<td>The identifier of the host that handles the replicated data. The identifier comprises the server identifier and the host moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td><strong>ResourcePoolIdentifier</strong></td>
<td>The identifier of the resource pool for the recovered virtual machines. The identifier comprises the server identifier and the resource pool moref, with the format, serverid.moref.</td>
</tr>
</tbody>
</table>
### Virtual machine NICs Response values for

https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/nic

and

https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/nic/{{nicIdentifier}}.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>VmIdentifier</td>
<td>The identifier of the virtual machine. The identifier comprises the server identifier and the virtual machine moref, with the format, serverid:moref.</td>
</tr>
<tr>
<td>Volumes</td>
<td>Information about the volumes used by the virtual machine.</td>
</tr>
<tr>
<td>Datastore</td>
<td>Information about the datastore used by the virtual machine.</td>
</tr>
<tr>
<td>DatastoreClusterIdentifier</td>
<td>The identifier of the datastore cluster used in the recovery site for the volume. Note: Only when the recovery site is a vSphere site.</td>
</tr>
<tr>
<td>IsThin</td>
<td><strong>True</strong>: The recovery volumes are thin-provisioned. <strong>False</strong>: The recovery volumes are not thin-provisioned.</td>
</tr>
<tr>
<td>ExistingVolume</td>
<td>For future use.</td>
</tr>
<tr>
<td>DatastoreIdentifier</td>
<td>For future use.</td>
</tr>
<tr>
<td>ExistedVmIdentifier</td>
<td>For future use.</td>
</tr>
<tr>
<td>Mode</td>
<td>For future use.</td>
</tr>
<tr>
<td>Path</td>
<td>For future use.</td>
</tr>
<tr>
<td>IsSwap</td>
<td><strong>True</strong>: The recovery disk is marked as a temp data disk. <strong>False</strong>: The recovery disk is not marked as a temp data disk.</td>
</tr>
<tr>
<td>VolumeIdentifier</td>
<td>The identifier of the volume.</td>
</tr>
<tr>
<td>Failover</td>
<td>Information about the networks used for failover by this virtual machine.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td></td>
</tr>
<tr>
<td>DnsSuffix</td>
<td>The DNS name excluding the host.</td>
</tr>
<tr>
<td>IpConfig</td>
<td>Information about the IP configuration of the recovery site used for failovers.</td>
</tr>
<tr>
<td>Gateway</td>
<td>The mask for the network.</td>
</tr>
<tr>
<td>IsDhcp</td>
<td><strong>True</strong>: DHCP is used for the virtual machine on the recovery site. <strong>False</strong>: A static IP address is used for the virtual machine on the recovery site.</td>
</tr>
<tr>
<td>PrimaryDns</td>
<td>The IP address of the primary DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>SecondaryDns</td>
<td>The IP address of the alternate, secondary, DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>StaticIp</td>
<td>The IP of the restored virtual machine.</td>
</tr>
<tr>
<td>SubnetMask</td>
<td>The subnet mask for the network.</td>
</tr>
<tr>
<td>NetworkIdentifier</td>
<td>The network identifier of the network used for failover or move by this virtual machine.</td>
</tr>
<tr>
<td>ShouldReplaceMacAddress</td>
<td><strong>True</strong>: The Media Access Control (MAC) address used on the protected site is copied to the recovery site. <strong>False</strong>: The Media Access Control (MAC) address used on the protected site is not copied to the recovery site.</td>
</tr>
<tr>
<td>FailoverTest</td>
<td>Information about the networks used for testing failover by this virtual machine.</td>
</tr>
</tbody>
</table>
VPG Settings: POST

The POST method is used for the following purposes:

**PARAMETER** | **DESCRIPTION**
--- | ---
Hypervisor | 
DnsSuffix | The DNS name excluding the host. 
IpConfig | Information about the IP configuration of the recovery site used for testing failovers. 
Gateway | The mask for the network. 
IsDhcp | **True**: DHCP is used for the virtual machine on the recovery site. 
**False**: A static IP address is used for the virtual machine on the recovery site. 
PrimaryDns | The IP address of the primary DNS server that handles Internet protocol mapping. 
SecondaryDns | The IP address of the alternate, secondary, DNS server that handles Internet protocol mapping. 
StaticIp | The IP of the restored virtual machine. 
SubnetMask | The subnet mask for the network. 
NetworkIdentifier | The network identifier of the network used for testing failover by this virtual machine. 
ShouldReplaceMacAddress | **True**: Copy the Media Access Control (MAC) address used on the protected site to the recovery site. 
**False**: Do not copy the MAC address from the protected site to the recovery site. 
NicIdentifier | The identifier of the NIC for which settings are returned. 

**Virtual machine volumes** Response values for

https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/volumes and

https://zvm_ip:port/v1/vpgSettings/vpgSettingsIdentifier}/vms/{vmIdentifier}/volumes/{volumeId}.

**PARAMETER** | **DESCRIPTION**
--- | ---
Datastore | Information about the datastore used by the virtual machine. 
DatastoreClusterIdentifier | The identifier of the datastore cluster used in the recovery site for the volume. 
**Note**: Only when the recovery site is a vSphere site. 
DatastoreIdentifier | The identifier of the storage. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref. 
IsThin | **True**: The recovery volumes are thin-provisioned. 
**False**: The recovery volumes are not thin-provisioned. 
ExistingVolume | For future use. 
DatastoreIdentifier | For future use. 
ExistedVmlIdentifier | For future use. 
Mode | For future use. 
Path | For future use. 
IsSwap | **True**: The recovery disk is marked as a temp data disk. 
**False**: The recovery disk is not marked as a temp data disk. 
VolumeIdentifier | The identifier of the volume.
To create a VPG settings identifier, or a session identifier.
To commits the settings object to deploy the VPG in the Zerto Virtual Manager
To add virtual machines to a VPG
To add NIC settings to a virtual machine.

You can use the same URL, https://zvm_ip:port/v1/vpgSettings, to either create a new VPG or edit an existing one.
To create a new VPG, put the parameters you wish to include in the VPG - in the request body.
To update an existing VPG, put the VPG Identifier in the request body.

Note: A maximum of 100 VPGSettings objects can coexist in temporary state. Make sure you commit or delete VPG settings objects that are no longer in use.

See also:
- “VPG Settings: GET”, on page 119
- “VPG Settings: PUT”, on page 155
- “VPG Settings: DELETE”, on page 175

**URL**

**Create a new VPG settings Identifier**  https://zvm_ip:port/v1/vpgSettings

**Create VPG settings Identifier for an existing VPG**  
https://zvm_ip:port/v1/vpgSettings

**Commit a settings object**  
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/commit

**Add VMs to a settings object**  
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms

Where:

<table>
<thead>
<tr>
<th>zvm_ip</th>
<th>The IP address of the Zerto Virtual Manager where the API is run.</th>
</tr>
</thead>
<tbody>
<tr>
<td>port</td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
<tr>
<td>vpgSettingsIdentifier</td>
<td>The identifier of the VPG settings object.</td>
</tr>
<tr>
<td>vmIdentifier</td>
<td>The identifier of the virtual machine that is to be added to a VPG or for a NIC that is added to a virtual machine in the VPG. The identifier comprises the server identifier and the virtual machine moref, with the format, serverid.moref.</td>
</tr>
</tbody>
</table>

When creating a VPG settings object, the following is an example request Json body for https://zvm_ip:port/v1/vpgSettings.

```json
"Backup": {
  "RepositoryIdentifier": "String content",
  "RetentionPeriod": "String content",
  "Retry": {
    "IntervalInMinutes": 2147483647,
    "Number": 2147483647,
    "Retry": Boolean
  },
  "Scheduler": {
    "DayOfWeek": "String content",
    "SchedulerPeriod": "String content",
    "TimeOfDay": "String content"
  }
},
```

VPG Management API
"Basic": {
  "JournalHistoryInHours": 2147483647,
  "Name": "String content",
  "Priority": "String content",
  "ProtectedSiteIdentifier": "String content",
  "RecoverySiteIdentifier": "String content",
  "RpoInSeconds": 4294967295,
  "ServiceProfileIdentifier": "String content",
  "TestIntervalInMinutes": 2147483647,
  "UseWanCompression": Boolean,
  "ZorgIdentifier": "String content"
},

"BootGroups": {
  "BootGroups": [{
    "BootDelayInSeconds": 4294967295,
    "BootGroupIdentifier": "String content",
    "Name": "String content"
  }]
},

"Journal": {
  "DatastoreIdentifier": "String content",
  "Limitation": {
    "HardLimitInMB": 2147483647,
    "HardLimitInPercent": 2147483647,
    "WarningThresholdInMB": 2147483647,
    "WarningThresholdInPercent": 2147483647
  }
},

"Networks": {
  "Failover": {
    "Hypervisor": {
      "DefaultNetworkIdentifier": "String content"
    }
  },
  "FailoverTest": {
    "Hypervisor": {
      "DefaultNetworkIdentifier": "String content"
    }
  }
},

"Recovery": {
  "DefaultDatastoreClusterIdentifier": "String content",
  "DefaultDatastoreIdentifier": "String content",
  "DefaultFolderIdentifier": "String content",
  "DefaultHostClusterIdentifier": "String content",
  "DefaultHostIdentifier": "String content",
  "ResourcePoolIdentifier": "String content"
}
"Scripting": {
    "PostBackup": {
        "Command": "String content",
        "Parameters": "String content",
        "TimeoutInSeconds": 2147483647
    },
    "PostRecovery": {
        "Command": "String content",
        "Parameters": "String content",
        "TimeoutInSeconds": 2147483647
    },
    "PreRecovery": {
        "Command": "String content",
        "Parameters": "String content",
        "TimeoutInSeconds": 2147483647
    }
},

"Vms": [{
    "BootGroupIdentifier": "String content",
    "Journal": {
        "DatastoreIdentifier": "String content",
        "Limitation": {
            "HardLimitInMB": 4294967295,
            "HardLimitInPercent": 4294967295,
            "WarningThresholdInMB": 4294967295,
            "WarningThresholdInPercent": 4294967295
        }
    }
}]}
"Nics": [{
  "Failover": {
    "Hypervisor": {
      "DnsSuffix": "String content",
      "IpConfig": {
        "Gateway": "String content",
        "IsDhcp": Boolean,
        "PrimaryDns": "String content",
        "SecondaryDns": "String content",
        "StaticIp": "String content",
        "SubnetMask": "String content"
      },
      "NetworkIdentifier": "String content",
      "ShouldReplaceMacAddress": Boolean
    }
  },
  "FailoverTest": {
    "Hypervisor": {
      "DnsSuffix": "String content",
      "IpConfig": {
        "Gateway": "String content",
        "IsDhcp": Boolean,
        "PrimaryDns": "String content",
        "SecondaryDns": "String content",
        "StaticIp": "String content",
        "SubnetMask": "String content"
      },
      "NetworkIdentifier": "String content",
      "ShouldReplaceMacAddress": Boolean
    }
  },
  "NicIdentifier": "String content"
},
"Recovery": {
  "DatastoreClusterIdentifier": "String content",
  "DatastoreIdentifier": "String content",
  "FolderIdentifier": "String content",
  "HostClusterIdentifier": "String content",
  "HostIdentifier": "String content",
  "ResourcePoolIdentifier": "String content"
},
"VmIdentifier": "String content",
"Volumes": [{
  "Datastore": {
    "DatastoreClusterIdentifier": "String content",
    "DatastoreIdentifier": "String content",
    "IsThin": Boolean
  },
  "ExistingVolume": {
    "DatastoreIdentifier": "String content",
    "ExistedVmIdentifier": "String content",
    "Mode": "String content",
    "Path": "String content"
  },
  "IsSwap": Boolean,
  "VolumeIdentifier": "String content"
}]}},
When creating a VPG settings object for an existing VPG, the following is an example request JSON body for

```json
{
  "VpgIdentifier":"String content"
}
```

The request body for https://zvm_ip:port/v1/vpgSettings/commit is empty.

The following is an example request JSON body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms.

```json
{
  "BootGroupIdentifier":"String content",
  "Journal":{
    "DatastoreIdentifier":"String content",
    "Limitation":{
      "HardLimitInMB":4294967295,
      "HardLimitInPercent":4294967295,
      "WarningThresholdInMB":4294967295,
      "WarningThresholdInPercent":4294967295
    }
  },
  "Nics":[
    "Failover":{
      "Hypervisor":{
        "DnsSuffix":"String content",
        "IpConfig":{
          "Gateway":"String content",
          "IsDhcp":true,
          "PrimaryDns":"String content",
          "SecondaryDns":"String content",
          "StaticIp":"String content",
          "SubnetMask":"String content"
        },
        "NetworkIdentifier":"String content",
        "ShouldReplaceMacAddress":true
      }
    },
    "FailoverTest":{
      "Hypervisor":{
        "DnsSuffix":"String content",
        "IpConfig":{
          "Gateway":"String content",
          "IsDhcp":true,
          "PrimaryDns":"String content",
          "SecondaryDns":"String content",
          "StaticIp":"String content",
          "SubnetMask":"String content"
        },
        "NetworkIdentifier":"String content",
        "ShouldReplaceMacAddress":true
      }
    }
  ],
  "NicIdentifier":"String content"
}
```
Request Values

Create a VPG settings object for a new VPG Request values for https://zvm_ip:port/v1/vpgSettings.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Backup</td>
<td>Information related to offsite backup.</td>
</tr>
<tr>
<td>RepositoryIdentifier</td>
<td>The identifier of the repository where offsite backups will be written.</td>
</tr>
<tr>
<td>RetentionPeriod</td>
<td>The length of time to keep offsite backups, up to a maximum of 12 months. Over time, Zerto reduces the number of stored offsite backups to save space.</td>
</tr>
</tbody>
</table>

**VALID VALUES FOR RetentionPeriod**

<table>
<thead>
<tr>
<th>RetentionPeriod</th>
<th># OF BACKUPS SAVED WHEN RUN DAILY</th>
<th># OF BACKUPS SAVED WHEN RUN WEEKLY</th>
</tr>
</thead>
<tbody>
<tr>
<td>OneWeek</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>OneMonth</td>
<td>11</td>
<td>5</td>
</tr>
<tr>
<td>ThreeMonths</td>
<td>13</td>
<td>7</td>
</tr>
<tr>
<td>SixMonths</td>
<td>16</td>
<td>10</td>
</tr>
<tr>
<td>NineMonths</td>
<td>19</td>
<td>13</td>
</tr>
<tr>
<td>OneYear</td>
<td>22</td>
<td>16</td>
</tr>
</tbody>
</table>

The **SchedulerPeriod** parameter defines whether backups are created daily or weekly.

For an explanation of how Zerto reduces the number of offsite backups, see the Zerto Virtual Replication Administration Guide.

<table>
<thead>
<tr>
<th>Retry</th>
<th>Information about backup retries.</th>
</tr>
</thead>
<tbody>
<tr>
<td>IntervallnMinutes</td>
<td>How much time to wait, in minutes, after a backup job fails before running the backup job again.</td>
</tr>
<tr>
<td>Number</td>
<td>The number of retries that will be attempted.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Retry</td>
<td><strong>True:</strong> The offsite backup job will rerun automatically if it fails. <strong>False:</strong> The offsite backup job will not rerun automatically if it fails.</td>
</tr>
<tr>
<td>Scheduler</td>
<td>Offsite backup schedule settings.</td>
</tr>
<tr>
<td>DayOfWeek</td>
<td>The day of the week that the offsite backups will run.</td>
</tr>
<tr>
<td>SchedulerPeriod</td>
<td><strong>Daily:</strong> The offsite backups will run every day. <strong>Weekly:</strong> The offsite backups will run once a week.</td>
</tr>
<tr>
<td>TimeOfDay</td>
<td>The time of day when offsite backup jobs will run. The time is based on a 24-hour clock.</td>
</tr>
<tr>
<td>Basic</td>
<td>Basic VPG settings.</td>
</tr>
<tr>
<td>JournalHistoryInHours</td>
<td>The time that all write commands are saved in the journal. The value is between 1 and 336 (14 days).</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the VPG.</td>
</tr>
<tr>
<td>Priority</td>
<td>The priority specified for the VPG. Possible values are:  <strong>Low:</strong> The VPG has a low priority for transferring data. <strong>Medium:</strong> The VPG has a medium priority for transferring data. <strong>High:</strong> The VPG has a high priority for transferring data.</td>
</tr>
<tr>
<td>ProtectedSiteIdentifier</td>
<td>The identifier of the source site where the VPG virtual machines will be protected. This is the site where the API runs.</td>
</tr>
<tr>
<td>RecoverySiteIdentifier</td>
<td>The identifier of the target site where the VPG virtual machines will be recovered.</td>
</tr>
<tr>
<td>RpoInSeconds</td>
<td>The maximum desired time between each automatic checkpoint being written to the journal before an alert is issued.</td>
</tr>
<tr>
<td>ServiceProfileIdentifier</td>
<td>The identifier of the service profile to use for the VPG when a Zerto Cloud Manager is used.</td>
</tr>
<tr>
<td>TestIntervalInMinutes</td>
<td>The time, in minutes, recommended between testing the integrity of the VPG. A warning is issued if a test is not done within this time frame. Possible values are:  <strong>0:</strong> No testing is expected. <strong>43200</strong> or <strong>null:</strong> Testing is expected monthly. <strong>131040:</strong> Testing is expected every three months. <strong>262080:</strong> Testing is expected every six months. <strong>394560:</strong> Testing is expected every nine months. <strong>525600:</strong> Testing is expected every twelve months.</td>
</tr>
<tr>
<td>UseWanCompression</td>
<td><strong>True:</strong> Data will be compressed before sending it to the recovery site. <strong>False:</strong> Data will not be compressed before sending it to the recovery site.</td>
</tr>
<tr>
<td>ZorgIdentifier</td>
<td>The identifier of the ZORG, Zerto organization, defined in the Zerto Cloud Manager.</td>
</tr>
<tr>
<td>BootGroups</td>
<td>Information about boot groups.</td>
</tr>
<tr>
<td>BootDelayInSeconds</td>
<td>Specifies the delay, in seconds, between starting up the virtual machines in this group and starting up the virtual machines in the next group.</td>
</tr>
<tr>
<td>BootGroupIdentifier</td>
<td>The identifier of a boot group.</td>
</tr>
<tr>
<td>Name</td>
<td>The name of a boot group.</td>
</tr>
<tr>
<td>Journal</td>
<td>Information about the journal.</td>
</tr>
<tr>
<td>DatastoreIdentifier</td>
<td>The identifier of the storage used by the journal. The identifier comprises the server identifier and the storage moref, with the format, serverid:moref.</td>
</tr>
<tr>
<td>Limitation</td>
<td>Information about the journal limitations.</td>
</tr>
<tr>
<td>HardLimitInMB</td>
<td>The maximum journal size in MBs. 0 means unlimited. Integer values.</td>
</tr>
</tbody>
</table>
### HardLimitInPercent

The percentage of the virtual machine volume size the journal can grow to. 0 means unlimited. Integer values.

### WarningThresholdInMB

The journal size, in MBs, that generates a warning that the journal is nearing its hard limit. 0 means unlimited. Integer values.

### WarningThresholdInPercent

The percentage of the virtual machine volume size that generates a warning. 0 means unlimited. Integer values.

### Networks

Information about the networks that connect the protected and recovery sites.

### Failover

Information about the networks used for failover.

#### Hypervisor

- 

#### DefaultNetworkIdentifier

The network identifier of the network to use during a failover or move operation in which the recovered virtual machines will run.

### FailoverTest

Information about the networks used for testing failover.

#### Hypervisor

- 

#### DefaultNetworkIdentifier

The network identifier of the network to use when testing the failover of virtual machines in the recovery site.

### Recovery

The identifier of the storage cluster of the VRA virtual machine.

#### DefaultDatastoreClusterIdentifier

The identifier of the default datastore cluster used in the recovery site. Note: Only when the recovery site is a vSphere site.

#### DefaultDatastoreIdentifier

The identifier of the default storage where the metadata files for the virtual machines are stored, such as the vmx or vhdx files. The identifier comprises the server identifier and the storage moref, with the format, `serverid.moref`.

#### DefaultFolderIdentifier

The identifier of the default host cluster that handles the replicated data. The identifier comprises the server identifier and the folder moref, with the format, `serverid.moref`.

#### DefaultHostClusterIdentifier

The identifier of the default host that handles the replicated data. The identifier comprises the server identifier and the host cluster moref, with the format, `serverid.moref`.

#### DefaultHostIdentifier

The identifier of the resource pool for the recovered virtual machines. The identifier comprises the server identifier and the host moref, with the format, `serverid.moref`.

#### ResourcePoolIdentifier

The identifier of the storage cluster of the VRA virtual machine. The identifier comprises the server identifier and the resource pool moref, with the format, `serverid.moref`.

### Scripting

Information about the scripts to run, either before or after recovery operation, or after an offsite backup is run.

#### PostBackup

Information about scripts that are run after an offsite backup is performed.

##### Command

The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site.

##### Parameters

Parameters to pass to the script.

##### TimeoutInSeconds

The time-out, in seconds, for the script to run.

#### PostRecovery

Information about scripts that are run after a recovery operation is performed.

##### Command

The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site.

##### Parameters

Parameters to pass to the script.

##### TimeoutInSeconds

The time-out, in seconds, for the script to run.
<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>PreRecovery</td>
<td>Information about scripts that are run before a recovery operation is performed.</td>
</tr>
<tr>
<td>Command</td>
<td>The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site.</td>
</tr>
<tr>
<td>Parameters</td>
<td>Parameters to pass to the script.</td>
</tr>
<tr>
<td>TimeoutInSeconds</td>
<td>The time-out, in seconds, for the script to run.</td>
</tr>
<tr>
<td>Vms</td>
<td>Information about the virtual machines in a VPG.</td>
</tr>
<tr>
<td>BootGroupId</td>
<td>The boot group identifier of a virtual machine.</td>
</tr>
<tr>
<td>Journal</td>
<td>Journal information of the virtual machine</td>
</tr>
<tr>
<td>DatastoreIdentifier</td>
<td>The identifier of the storage used by the journal for the virtual machine. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>Limitation</td>
<td>Information about the journal limitations.</td>
</tr>
<tr>
<td>HardLimitInMB</td>
<td>The maximum journal size in MBs. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>HardLimitInPercent</td>
<td>The percentage of the virtual machine volume size the journal can grow to. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>WarningThresholdInMB</td>
<td>The journal size, in MBs, that generates a warning that the journal is nearing its hard limit. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>WarningThresholdInPercent</td>
<td>The percentage of the virtual machine volume size that generates a warning. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>Nics</td>
<td>Information about NICs used by the virtual machine in the VPG.</td>
</tr>
<tr>
<td>Failover</td>
<td>Information about the networks used for failover by this virtual machine.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td>-</td>
</tr>
<tr>
<td>DnsSuffix</td>
<td>The DNS name excluding the host.</td>
</tr>
<tr>
<td>IpConfig</td>
<td>Information about the IP configuration of the recovery site used for failovers.</td>
</tr>
<tr>
<td>Gateway</td>
<td>The mask for the network.</td>
</tr>
<tr>
<td>IsDhcp</td>
<td><strong>True</strong>: DHCP will be used for the virtual machine on the recovery site. <strong>False</strong>: A static IP address will be used for the virtual machine on the recovery site.</td>
</tr>
<tr>
<td>PrimaryDns</td>
<td>The IP address of the primary DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>SecondaryDns</td>
<td>The IP address of the alternate, secondary, DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>StaticDns</td>
<td>The IP of the restored virtual machine.</td>
</tr>
<tr>
<td>SubnetMask</td>
<td>The subnet mask for the network.</td>
</tr>
<tr>
<td>NetworkIdentifier</td>
<td>The network identifier of the network used for failover or move by this virtual machine.</td>
</tr>
<tr>
<td>ShouldReplaceMacAddress</td>
<td><strong>True</strong>: The Media Access Control (MAC) address used on the protected site will be copied to the recovery site. <strong>False</strong>: The Media Access Control (MAC) address used on the protected site will not be copied to the recovery site.</td>
</tr>
<tr>
<td>FailoverTest</td>
<td>Information about the networks used for testing failover by this virtual machine.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td>-</td>
</tr>
<tr>
<td>DnsSuffix</td>
<td>The DNS name excluding the host.</td>
</tr>
<tr>
<td>IpConfig</td>
<td>Information about the IP configuration of the recovery site used for testing failovers.</td>
</tr>
<tr>
<td>Gateway</td>
<td>The mask for the network.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>IsDhcp</td>
<td><strong>True</strong>: DHCP will be used for the virtual machine on the recovery site.</td>
</tr>
<tr>
<td></td>
<td><strong>False</strong>: A static IP address will be used for the virtual machine on the recovery site.</td>
</tr>
<tr>
<td>PrimaryDns</td>
<td>The IP address of the primary DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>SecondaryDns</td>
<td>The IP address of the alternate, secondary, DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>StaticIp</td>
<td>The IP of the restored virtual machine.</td>
</tr>
<tr>
<td>SubnetMask</td>
<td>The subnet mask for the network.</td>
</tr>
<tr>
<td>NetworkIdentifier</td>
<td>The network identifier of the network used for testing failover by this virtual machine.</td>
</tr>
<tr>
<td>ShouldReplaceMacAddress</td>
<td><strong>True</strong>: The Media Access Control (MAC) address used on the protected site will be copied to the recovery site.</td>
</tr>
<tr>
<td></td>
<td><strong>False</strong>: The Media Access Control (MAC) address used on the protected site will not be copied to the recovery site.</td>
</tr>
<tr>
<td>NicIdentifier</td>
<td>The identifier of the NIC for which settings are returned.</td>
</tr>
<tr>
<td>Recovery</td>
<td>Information about the recovery site.</td>
</tr>
<tr>
<td>DatastoreClusterIdentifier</td>
<td>The identifier of the datastore cluster used in the recovery site for the VM.</td>
</tr>
<tr>
<td></td>
<td>Note: Only when the recovery site is a vSphere site.</td>
</tr>
<tr>
<td>DatastoreIdentifier</td>
<td>The identifier of the storage. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>FolderIdentifier</td>
<td>The identifier of the folder used for recovery by the virtual machine. The identifier</td>
</tr>
<tr>
<td></td>
<td>comprises the server identifier and the folder moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>HostClusterIdentifier</td>
<td>The identifier of the host cluster that handles the replicated data. The identifier</td>
</tr>
<tr>
<td></td>
<td>comprises the server identifier and the host cluster moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>HostIdentifier</td>
<td>The identifier of the host that handles the replicated data. The identifier</td>
</tr>
<tr>
<td></td>
<td>comprises the server identifier and the host moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>ResourcePoolIdentifier</td>
<td>The identifier of the resource pool for the recovered virtual machines. The identifier</td>
</tr>
<tr>
<td></td>
<td>comprises the server identifier and the resource pool moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>VmIdentifier</td>
<td>The identifier of the virtual machine. The identifier comprises the server identifier and the</td>
</tr>
<tr>
<td></td>
<td>virtual machine moref, with the format, serverid.more.</td>
</tr>
<tr>
<td>Volumes</td>
<td>Information about the volumes used by the virtual machine.</td>
</tr>
<tr>
<td>Datastore</td>
<td>Information about the datastore used by the virtual machine.</td>
</tr>
<tr>
<td>DatastoreClusterIdentifier</td>
<td>The identifier of the datastore cluster used in the recovery site for the volume.</td>
</tr>
<tr>
<td></td>
<td>Note: Only when the recovery site is a vSphere site.</td>
</tr>
<tr>
<td>DatastoreIdentifier</td>
<td>The identifier of the storage. The identifier comprises the server identifier and the storage</td>
</tr>
<tr>
<td></td>
<td>moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>IsThin</td>
<td><strong>True</strong>: The recovery volumes are thin-provisioned.</td>
</tr>
<tr>
<td></td>
<td><strong>False</strong>: The recovery volumes are not thin-provisioned.</td>
</tr>
<tr>
<td>ExistingVolume</td>
<td>For future use.</td>
</tr>
<tr>
<td>DatastoreIdentifier</td>
<td>For future use.</td>
</tr>
<tr>
<td>ExistedVmIdentifier</td>
<td>For future use.</td>
</tr>
</tbody>
</table>
**Create a VPG settings object for an existing VPG** Request values for `https://zvm_ip:port/v1/vpgSettings`.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mode</td>
<td>For future use.</td>
</tr>
<tr>
<td>Path</td>
<td>For future use.</td>
</tr>
</tbody>
</table>
| **IsSwap**     | **True**: The recovery disk is marked as a temp data disk.  
                | **False**: The recovery disk is not marked as a temp data disk. |
| VolumemIdentifier | The identifier of the volume.                    |

**Commit a VPG settings object** Request values for `https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/commit` is empty.

**Add a virtual machine to a VPG settings object** Request values for `https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms`.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BootGroupIdentifier</strong></td>
<td>The boot group identifier of a virtual machine.</td>
</tr>
<tr>
<td>Journal</td>
<td>Journal information of the virtual machine</td>
</tr>
<tr>
<td><strong>DatastoreIdentifier</strong></td>
<td>The identifier of the storage used by the journal for the virtual machine. The identifier comprises the server identifier and the storage moref, with the format, <code>serverid.moref</code>.</td>
</tr>
<tr>
<td>Limitation</td>
<td>Information about the journal limitations.</td>
</tr>
<tr>
<td><strong>HardLimitInMB</strong></td>
<td>The maximum journal size in MBs. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td><strong>HardLimitInPercent</strong></td>
<td>The percentage of the virtual machine volume size the journal can grow to. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td><strong>WarningThresholdInMB</strong></td>
<td>The journal size, in MBs, that generates a warning that the journal is nearing its hard limit. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td><strong>WarningThresholdInPercent</strong></td>
<td>The percentage of the virtual machine volume size that generates a warning. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td><strong>Nics</strong></td>
<td>Information about NICs used by the virtual machine in the VPG.</td>
</tr>
<tr>
<td>Failover</td>
<td>Information about the networks used for failover by this virtual machine.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td>-</td>
</tr>
<tr>
<td><strong>DnsSuffix</strong></td>
<td>The DNS name excluding the host.</td>
</tr>
<tr>
<td><strong>IpConfig</strong></td>
<td>Information about the IP configuration of the recovery site used for failovers.</td>
</tr>
<tr>
<td>Gateway</td>
<td>The mask for the network.</td>
</tr>
</tbody>
</table>
| **IsDhcp**     | **True**: DHCP will be used for the virtual machine on the recovery site.  
<pre><code>            | **False**: A static IP address will be used for the virtual machine on the recovery site. |
</code></pre>
<p>| <strong>PrimaryDns</strong> | The IP address of the primary DNS server that handles Internet protocol mapping. |
| <strong>SecondaryDns</strong> | The IP address of the alternate, secondary, DNS server that handles Internet protocol mapping. |
| <strong>StaticIp</strong>   | The IP of the restored virtual machine.          |
| <strong>SubnetMask</strong> | The subnet mask for the network.                 |
| <strong>NetworkIdentifier</strong> | The network identifier of the network used for failover or move by this virtual machine. |</p>
<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ShouldReplaceMacAddress</td>
<td><strong>True:</strong> The Media Access Control (MAC) address used on the protected site will be copied to the recovery site. <strong>False:</strong> The Media Access Control (MAC) address used on the protected site will not be copied to the recovery site.</td>
</tr>
<tr>
<td>FailoverTest</td>
<td>Information about the networks used for testing failover by this virtual machine.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td>-</td>
</tr>
<tr>
<td>DnsSuffix</td>
<td>The DNS name excluding the host.</td>
</tr>
<tr>
<td>IpConfig</td>
<td>Information about the IP configuration of the recovery site used for testing failovers.</td>
</tr>
<tr>
<td>Gateway</td>
<td>The mask for the network.</td>
</tr>
<tr>
<td>IsDhcp</td>
<td><strong>True:</strong> DHCP will be used for the virtual machine on the recovery site. <strong>False:</strong> A static IP address will be used for the virtual machine on the recovery site.</td>
</tr>
<tr>
<td>PrimaryDns</td>
<td>The IP address of the primary DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>SecondaryDns</td>
<td>The IP address of the alternate, secondary, DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>StaticIp</td>
<td>The IP of the restored virtual machine.</td>
</tr>
<tr>
<td>SubnetMask</td>
<td>The subnet mask for the network.</td>
</tr>
<tr>
<td>NetworkIdentifier</td>
<td>The network identifier of the network used for testing failover by this virtual machine.</td>
</tr>
<tr>
<td>ShouldReplaceMacAddress</td>
<td><strong>True:</strong> The Media Access Control (MAC) address used on the protected site will be copied to the recovery site. <strong>False:</strong> The Media Access Control (MAC) address used on the protected site will not be copied to the recovery site.</td>
</tr>
<tr>
<td>NicIdentifier</td>
<td>The identifier of the NIC for which settings are returned.</td>
</tr>
<tr>
<td>Recovery</td>
<td>Information about the recovery site.</td>
</tr>
<tr>
<td>DatastoreClusterIdentifier</td>
<td>The identifier of the datastore cluster used in the recovery site for the VM. Note: Only when the recovery site is a vSphere site.</td>
</tr>
<tr>
<td>DatastoreIdentifier</td>
<td>The identifier of the storage. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>FolderIdentifier</td>
<td>The identifier of the folder used for recovery by the virtual machine. The identifier comprises the server identifier and the folder moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>HostClusterIdentifier</td>
<td>The identifier of the host cluster that handles the replicated data. The identifier comprises the server identifier and the host cluster moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>HostIdentifier</td>
<td>The identifier of the host that handles the replicated data. The identifier comprises the server identifier and the host moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>ResourcePoolIdentifier</td>
<td>The identifier of the resource pool for the recovered virtual machines. The identifier comprises the server identifier and the resource pool moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>VmIdentifier</td>
<td>The identifier of the virtual machine. The identifier comprises the server identifier and the virtual machine moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>Volumes</td>
<td>Information about the volumes used by the virtual machine.</td>
</tr>
<tr>
<td>Datastore</td>
<td>Information about the datastore used by the virtual machine.</td>
</tr>
</tbody>
</table>
**PARAMETER** | **DESCRIPTION**
--- | ---
DatastoreClusterIdentifier | The identifier of the datastore cluster used in the recovery site for the volume. Note: Only when the recovery site is a vSphere site.
DatastoreIdentifier | The identifier of the storage. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.
IsThin | True: The recovery volumes are thin-provisioned. False: The recovery volumes are not thin-provisioned.
ExistingVolume | For future use.
DatastoreIdentifier | For future use.
ExistedVmlIdentifier | For future use.
Mode | For future use.
Path | For future use.
IsSwap | True: The recovery disk is marked as a temp data disk. False: The recovery disk is not marked as a temp data disk.
VolumemIdentifier | The identifier of the volume.

**Json Response Format**

The following is an example response Json body for `https://zvm_ip:port/v1/vpgSettings` and `https://zvm_ip:port/v1/vpgSettings/commit`.

```
"String content"
```


**Response values**

The response value for `https://zvm_ip:port/v1/vpgSettings` is the `vpgSettingsIdentifier`.

The response value for `https://zvm_ip:port/v1/vpgSettings/commit` is the `task identifier`.

**VPG Settings: PUT**

The PUT method is used to update a VPG settings object.

- “VPG Settings: GET”, on page 119
- “VPG Settings: POST”, on page 142
- “VPG Settings: DELETE”, on page 175

**Edit VPG settings object**  
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}  
**Edit backup settings object**  
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/backup  
**Edit basic settings object**  
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/basic  
**Edit boot settings object**  
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/bootgroup  
**Edit journal settings object**  
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/journal  
**Edit network settings object**  
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/networks
Edit network mapping settings object: for future use

https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/networksmapping

Edit recovery settings object

https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/recovery

Edit script settings object

https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/scripting

Edit VM settings object

https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}

Edit NIC settings object

https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/nics/{nicIdentifier}

Edit volume settings object

https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/volumes/{volumeId}
Where:

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>zvm_ip</strong></td>
<td>The IP address of the Zerto Virtual Manager where the API is run.</td>
</tr>
<tr>
<td><strong>port</strong></td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
<tr>
<td><strong>vpgSettingsIdentifier</strong></td>
<td>The identifier of the VPG settings object.</td>
</tr>
<tr>
<td><strong>vmIdentifier</strong></td>
<td>The identifier of the virtual machine. The identifier comprises the server identifier and the virtual machine moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td><strong>nicIdentifier</strong></td>
<td>The identifier of a NIC that is to be updated.</td>
</tr>
<tr>
<td><strong>volumeld</strong></td>
<td>The identifier of the volume that is to be updated.</td>
</tr>
</tbody>
</table>

**Json Request Format**

The following is an example request Json body for `https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}`.

```json
{
  "Backup":{
    "RepositoryIdentifier":"String content",
    "RetentionPeriod":"String content",
    "Retry":{
      "IntervalInMinutes":2147483647,
      "Number":2147483647,
      "Retry":true
    },
    "Scheduler":{
      "DayOfWeek":"String content",
      "SchedulerPeriod":"String content",
      "TimeOfDay":"String content"
    }
  },
  "Basic":{
    "JournalHistoryInHours":2147483647,
    "Name":"String content",
    "Priority":"String content",
    "ProtectedSiteIdentifier":"String content",
    "RecoverySiteIdentifier":"String content",
    "RpoInSeconds":4294967295,
    "ServiceProfileIdentifier":"String content",
    "TestIntervalInMinutes":2147483647,
    "UseWanCompression":true,
    "ZorgIdentifier":"String content"
  },
  "BootGroups":{
    "BootGroups":[{
      "BootDelayInSeconds":4294967295,
      "BootGroupIdentifier":"String content",
      "Name":"String content"
    }]
  },
  "Journal":{
    "DatastoreIdentifier":"String content",
    "Limitation":{
      "HardLimitInMB":2147483647,
      "HardLimitInPercent":2147483647,
      "WarningThresholdInMB":2147483647,
      "WarningThresholdInPercent":2147483647
    }
  }
}
```
"Networks":{
  "Failover":{
    "Hypervisor":{
      "DefaultNetworkIdentifier":"String content"
    }
  },
  "FailoverTest":{
    "Hypervisor":{
      "DefaultNetworkIdentifier":"String content"
    }
  }
},

"Recovery":{
  "DefaultDatastoreClusterIdentifier":"String content",
  "DefaultDatastoreIdentifier":"String content",
  "DefaultFolderIdentifier":"String content",
  "DefaultHostClusterIdentifier":"String content",
  "DefaultHostIdentifier":"String content",
  "ResourcePoolIdentifier":"String content"
},

"Scripting":{
  "PostBackup":{
    "Command":"String content",
    "Parameters":"String content",
    "TimeoutInSeconds":2147483647
  },
  "PostRecovery":{
    "Command":"String content",
    "Parameters":"String content",
    "TimeoutInSeconds":2147483647
  },
  "PreRecovery":{
    "Command":"String content",
    "Parameters":"String content",
    "TimeoutInSeconds":2147483647
  }
},

"Vms":{
  "BootGroupIdentifier":"String content",
  "Journal":{
    "DatastoreIdentifier":"String content",
    "Limitation":{
      "HardLimitInMB":4294967295,
      "HardLimitInPercent":4294967295,
      "WarningThresholdInMB":4294967295,
      "WarningThresholdInPercent":4294967295
    }
  }
}
"Nics": [{
    "Failover": {
        "Hypervisor": {
            "DnsSuffix": "String content",
            "IpConfig": {
                "Gateway": "String content",
                "IsDhcp": true,
                "PrimaryDns": "String content",
                "SecondaryDns": "String content",
                "StaticIp": "String content",
                "SubnetMask": "String content"
            },
            "NetworkIdentifier": "String content",
            "ShouldReplaceMacAddress": true
        }
    },
    "FailoverTest": {
        "Hypervisor": {
            "DnsSuffix": "String content",
            "IpConfig": {
                "Gateway": "String content",
                "IsDhcp": true,
                "PrimaryDns": "String content",
                "SecondaryDns": "String content",
                "StaticIp": "String content",
                "SubnetMask": "String content"
            },
            "NetworkIdentifier": "String content",
            "ShouldReplaceMacAddress": true
        }
    },
    "NicIdentifier": "String content"
},
"Recovery": {
    "DatastoreClusterIdentifier": "String content",
    "DatastoreIdentifier": "String content",
    "FolderIdentifier": "String content",
    "HostClusterIdentifier": "String content",
    "HostIdentifier": "String content",
    "ResourcePoolIdentifier": "String content"
},
"VmIdentifier": "String content",
"Volumes": [{
    "Datastore": {
        "DatastoreClusterIdentifier": "String content",
        "DatastoreIdentifier": "String content",
        "IsThin": true
    },
    "ExistingVolume": {
        "DatastoreIdentifier": "String content",
        "ExistedVmIdentifier": "String content",
        "Mode": "String content",
        "Path": "String content"
    },
    "IsSwap": true,
    "VolumeIdentifier": "String content"
}]}
The following is an example request Json body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/backup.

```json
{
    "RepositoryIdentifier":"String content",
    "RetentionPeriod":"String content",
    "Retry":{
        "IntervalInMinutes":2147483647,
        "Number":2147483647,
        "Retry":true
    },
    "Scheduler":{
        "DayOfWeek":"String content",
        "SchedulerPeriod":"String content",
        "TimeOfDay":"String content"
    }
}
```

The following is an example request Json body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/basic.

```json
{
    "JournalHistoryInHours":2147483647,
    "Name":"String content",
    "Priority":"String content",
    "ProtectedSiteIdentifier":"String content",
    "RecoverySiteIdentifier":"String content",
    "RpoInSeconds":4294967295,
    "ServiceProfileIdentifier":"String content",
    "TestIntervalInMinutes":2147483647,
    "UseWanCompression":true,
    "ZorgIdentifier":"String content"
}
```

The following is an example request Json body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/bootgroup.

```json
{
    "BootGroups":[
        {
            "BootDelayInSeconds":4294967295,
            "BootGroupIdentifier":"String content",
            "Name":"String content"
        }
    ]
}
```

The following is an example request Json body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/journal.
The following is an example request Json body for

```json
{
  "Failover":{
    "Hypervisor":{
      "DefaultNetworkIdentifier":"String content"
    }
  },
  "FailoverTest":{
    "Hypervisor":{
      "DefaultNetworkIdentifier":"String content"
    }
  }
}
```

The following is an example request Json body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/recovery.

```json
{
  "DefaultDatastoreClusterIdentifier":"String content",
  "DefaultDatastoreIdentifier":"String content",
  "DefaultFolderIdentifier":"String content",
  "DefaultHostClusterIdentifier":"String content",
  "DefaultHostIdentifier":"String content",
  "ResourcePoolIdentifier":"String content"
}
```

The following is an example request Json body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/scripting.

```json
{
  "PostBackup":{
    "Command":"String content",
    "Parameters":"String content",
    "TimeoutInSeconds":2147483647
  },
  "PostRecovery":{
    "Command":"String content",
    "Parameters":"String content",
    "TimeoutInSeconds":2147483647
  },
  "PreRecovery":{
    "Command":"String content",
    "Parameters":"String content",
    "TimeoutInSeconds":2147483647
  }
}
```
The following is an example request Json body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms.

```
{
   "BootGroupIdentifier":"String content",
   "Journal":{
      "DatastoreIdentifier":"String content",
      "Limitation":{
         "HardLimitInMB":4294967295,
         "HardLimitInPercent":4294967295,
         "WarningThresholdInMB":4294967295,
         "WarningThresholdInPercent":4294967295
      }
   },
   "Nics":{
      "Failover":{
         "Hypervisor":{
            "DnsSuffix":"String content",
            "IpConfig":{
               "Gateway":"String content",
               "IsDhcp":true,
               "PrimaryDns":"String content",
               "SecondaryDns":"String content",
               "StaticIp":"String content",
               "SubnetMask":"String content"
            },
            "NetworkIdentifier":"String content",
            "ShouldReplaceMacAddress":true
         }
      },
      "FailoverTest":{
         "Hypervisor":{
            "DnsSuffix":"String content",
            "IpConfig":{
               "Gateway":"String content",
               "IsDhcp":true,
               "PrimaryDns":"String content",
               "SecondaryDns":"String content",
               "StaticIp":"String content",
               "SubnetMask":"String content"
            },
            "NetworkIdentifier":"String content",
            "ShouldReplaceMacAddress":true
         }
      },
      "NicIdentifier":"String content"
   }
},

"Recovery":{
   "DatastoreClusterIdentifier":"String content",
   "DatastoreIdentifier":"String content",
   "FolderIdentifier":"String content",
   "HostClusterIdentifier":"String content",
   "HostIdentifier":"String content",
   "ResourcePoolIdentifier":"String content"
}
```

The following is an example request Json body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/nics/{nicIdentifier}.

```json
{
  "VmIdentifier":"String content",
  "Volumes":[
    "Datastore":{
      "DatastoreClusterIdentifier":"String content",
      "DatastoreIdentifier":"String content",
      "IsThin":true
    },
    "ExistingVolume":{
      "DatastoreIdentifier":"String content",
      "ExistingVmIdentifier":"String content",
      "Mode":"String content",
      "Path":"String content"
    },
    "IsSwap":true,
    "VolumeIdentifier":"String content"
  }
}
```

The following is an example request Json body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/volumes/{volumeId}.

```json
{
  "Failover":{
    "Hypervisor":{
      "DnsSuffix":"String content",
      "IpConfig":{
        "Gateway":"String content",
        "IsDhcp":true,
        "PrimaryDns":"String content",
        "SecondaryDns":"String content",
        "StaticIp":"String content",
        "SubnetMask":"String content"
      },
      "NetworkIdentifier":"String content",
      "ShouldReplaceMacAddress":true
    }
  },
  "FailoverTest":{
    "Hypervisor":{
      "DnsSuffix":"String content",
      "IpConfig":{
        "Gateway":"String content",
        "IsDhcp":true,
        "PrimaryDns":"String content",
        "SecondaryDns":"String content",
        "StaticIp":"String content",
        "SubnetMask":"String content"
      },
      "NetworkIdentifier":"String content",
      "ShouldReplaceMacAddress":true
    }
  },
  "NicIdentifier":"String content"
}
```
Request Values

Update a VPG settings object Request values for https://zvm_ip:port/v1/vpgSettings.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Backup</td>
<td>Information related to offsite backup.</td>
</tr>
<tr>
<td>RepositoryIdentifier</td>
<td>The identifier of the repository where offsite backups will be written.</td>
</tr>
<tr>
<td>RetentionPeriod</td>
<td>The length of time to keep offsite backups, up to a maximum of 12 months.</td>
</tr>
<tr>
<td></td>
<td>Over time, Zerto reduces the number of stored offsite backups to save space.</td>
</tr>
<tr>
<td></td>
<td><strong>VALID VALUES FOR RetentionPeriod</strong></td>
</tr>
<tr>
<td></td>
<td><strong># OF BACKUPS SAVED WHEN RUN DAILY</strong></td>
</tr>
<tr>
<td></td>
<td><strong># OF BACKUPS SAVED WHEN RUN WEEKLY</strong></td>
</tr>
<tr>
<td>OneWeek</td>
<td>7</td>
</tr>
<tr>
<td>OneMonth</td>
<td>11</td>
</tr>
<tr>
<td>ThreeMonths</td>
<td>13</td>
</tr>
<tr>
<td>SixMonths</td>
<td>16</td>
</tr>
<tr>
<td>NineMonths</td>
<td>19</td>
</tr>
<tr>
<td>OneYear</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td><strong>The SchedulerPeriod parameter defines whether backups are created daily or</strong></td>
</tr>
<tr>
<td></td>
<td><strong>weekly.</strong></td>
</tr>
<tr>
<td></td>
<td><strong>For an explanation of how Zerto reduces the number of offsite backups, see</strong></td>
</tr>
<tr>
<td></td>
<td><strong>the Zerto Virtual Replication Administration Guide.</strong></td>
</tr>
<tr>
<td>Retry</td>
<td>Information about backup retries.</td>
</tr>
<tr>
<td>IntervalInMinutes</td>
<td>How much time to wait, in minutes, after a backup job fails before running the</td>
</tr>
<tr>
<td></td>
<td>backup job again.</td>
</tr>
<tr>
<td>Number</td>
<td>The number of retries that will be attempted.</td>
</tr>
<tr>
<td>Retry</td>
<td><strong>True:</strong> The offsite backup job will rerun automatically if it fails.</td>
</tr>
<tr>
<td></td>
<td><strong>False:</strong> The offsite backup job will not rerun automatically if it fails.</td>
</tr>
<tr>
<td>Scheduler</td>
<td>Offsite backup schedule settings.</td>
</tr>
<tr>
<td>DayOfWeek</td>
<td>The day of the week that the offsite backups will run.</td>
</tr>
<tr>
<td>SchedulerPeriod</td>
<td><strong>Daily:</strong> The offsite backups will run every day.</td>
</tr>
<tr>
<td></td>
<td><strong>Weekly:</strong> The offsite backups will run once a week.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>TimeOfDay</td>
<td>The time of day when offsite backup jobs will run. The time is based on a 24-hour clock.</td>
</tr>
<tr>
<td>Basic</td>
<td>Basic VPG settings.</td>
</tr>
<tr>
<td>JournalHistoryInHours</td>
<td>The time that all write commands are saved in the journal. The value is between 1 and 336 (14 days).</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the VPG.</td>
</tr>
<tr>
<td>Priority</td>
<td>The priority specified for the VPG. Possible values are:</td>
</tr>
<tr>
<td></td>
<td>Low: The VPG has a low priority for transferring data.</td>
</tr>
<tr>
<td></td>
<td>Medium: The VPG has a medium priority for transferring data.</td>
</tr>
<tr>
<td></td>
<td>High: The VPG has a high priority for transferring data.</td>
</tr>
<tr>
<td>ProtectedSiteIdentifier</td>
<td>The identifier of the source site where the VPG virtual machines will be protected. This is the site where the API runs.</td>
</tr>
<tr>
<td>RecoverySiteIdentifier</td>
<td>The identifier of the target site where the VPG virtual machines will be recovered.</td>
</tr>
<tr>
<td>RpoInSeconds</td>
<td>The maximum desired time between each automatic checkpoint being written to the journal before an alert is issued.</td>
</tr>
<tr>
<td>ServiceProfileIdentifier</td>
<td>The identifier of the service profile to use for the VPG when a Zerto Cloud Manager is used.</td>
</tr>
<tr>
<td>TestIntervalInMinutes</td>
<td>The time, in minutes, recommended between testing the integrity of the VPG. A warning is issued if a test is not done within this time frame. Possible values are:</td>
</tr>
<tr>
<td></td>
<td>0: No testing is expected.</td>
</tr>
<tr>
<td></td>
<td>43200 or null: Testing is expected monthly.</td>
</tr>
<tr>
<td></td>
<td>131040: Testing is expected every three months.</td>
</tr>
<tr>
<td></td>
<td>262080: Testing is expected every six months.</td>
</tr>
<tr>
<td></td>
<td>394560: Testing is expected every nine months.</td>
</tr>
<tr>
<td></td>
<td>525600: Testing is expected every twelve months.</td>
</tr>
<tr>
<td>UseWanCompression</td>
<td>True: Data will be compressed before sending it to the recovery site.</td>
</tr>
<tr>
<td></td>
<td>False: Data will not be compressed before sending it to the recovery site.</td>
</tr>
<tr>
<td>ZorgIdentifier</td>
<td>The identifier of the ZORG, Zerto organization, defined in the Zerto Cloud Manager.</td>
</tr>
<tr>
<td>BootGroups</td>
<td>Information about boot groups.</td>
</tr>
<tr>
<td>BootGroups</td>
<td>-</td>
</tr>
<tr>
<td>BootDelayInSeconds</td>
<td>Specifies the delay, in seconds, between starting up the virtual machines in this group and starting up the virtual machines in the next group.</td>
</tr>
<tr>
<td>BootGroupIdentifier</td>
<td>The identifier of a boot group.</td>
</tr>
<tr>
<td>Name</td>
<td>The name of a boot group.</td>
</tr>
<tr>
<td>Journal</td>
<td>Information about the journal.</td>
</tr>
<tr>
<td>DatastoreIdentifier</td>
<td>The identifier of the storage used by the journal. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>Limitation</td>
<td>Information about the journal limitations.</td>
</tr>
<tr>
<td>HardLimitInMB</td>
<td>The maximum journal size in MBs. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>HardLimitInPercent</td>
<td>The percentage of the virtual machine volume size the journal can grow to. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>WarningThresholdInMB</td>
<td>The journal size, in MBs, that generates a warning that the journal is nearing its hard limit. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>WarningThresholdInPercent</td>
<td>The percentage of the virtual machine volume size that generates a warning. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Networks</td>
<td>Information about the networks that connect the protected and recovery sites.</td>
</tr>
<tr>
<td>Failover</td>
<td>Information about the networks used for failover.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td></td>
</tr>
<tr>
<td>DefaultNetworkIdentifier</td>
<td>The network identifier of the network to use during a failover or move operation in which the recovered virtual machines will run.</td>
</tr>
<tr>
<td>FailoverTest</td>
<td>Information about the networks used for testing failover.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td></td>
</tr>
<tr>
<td>DefaultNetworkIdentifier</td>
<td>The network identifier of the network to use when testing the failover of virtual machines in the recovery site.</td>
</tr>
<tr>
<td>Recovery</td>
<td>The identifier of the storage cluster of the VRA virtual machine.</td>
</tr>
<tr>
<td>DefaultDatastoreClusterIdentifier</td>
<td>The identifier of the default datastore cluster used in the recovery site.</td>
</tr>
<tr>
<td></td>
<td>Note: Only when the recovery site is a vSphere site.</td>
</tr>
<tr>
<td>DefaultDatastoreIdentifier</td>
<td>The identifier of the default storage where the metadata files for the virtual machines are stored, such as the vmx or vhdx files. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultFolderIdentifier</td>
<td>The identifier of the default host cluster that handles the replicated data. The identifier comprises the server identifier and the folder moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultHostClusterIdentifier</td>
<td>The identifier of the default host that handles the replicated data. The identifier comprises the server identifier and the host cluster moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultHostIdentifier</td>
<td>The identifier of the resource pool for the recovered virtual machines. The identifier comprises the server identifier and the host moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>ResourcePoolIdentifier</td>
<td>The identifier of the storage cluster of the VRA virtual machine. The identifier comprises the server identifier and the resource pool moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>Scripting</td>
<td>Information about the scripts to run, either before or after recovery operation, or after an offsite backup is run.</td>
</tr>
<tr>
<td>PostBackup</td>
<td>Information about scripts that are run after an offsite backup is performed.</td>
</tr>
<tr>
<td>Command</td>
<td>The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site. &lt;br&gt;<strong>Note:</strong> Two backslash characters (\) must be specified instead of a single backslash character in every place in the path. For example, specify a path similar to the following: C:\ZertoScripts\MyScript.ps1 and not C:\ZertoScripts\MyScript.ps1.</td>
</tr>
<tr>
<td>Parameters</td>
<td>Parameters to pass to the script.</td>
</tr>
<tr>
<td>TimeoutInSeconds</td>
<td>The time-out, in seconds, for the script to run.</td>
</tr>
<tr>
<td>PostRecovery</td>
<td>Information about scripts that are run after a recovery operation is performed.</td>
</tr>
<tr>
<td>Command</td>
<td>The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site.</td>
</tr>
<tr>
<td>Parameters</td>
<td>Parameters to pass to the script.</td>
</tr>
<tr>
<td>TimeoutInSeconds</td>
<td>The time-out, in seconds, for the script to run.</td>
</tr>
<tr>
<td>PreRecovery</td>
<td>Information about scripts that are run before a recovery operation is performed.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Command</td>
<td>The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site.</td>
</tr>
<tr>
<td>Parameters</td>
<td>Parameters to pass to the script.</td>
</tr>
<tr>
<td>TimeoutInSeconds</td>
<td>The time-out, in seconds, for the script to run.</td>
</tr>
<tr>
<td>Vms</td>
<td>Information about the virtual machines in a VPG.</td>
</tr>
<tr>
<td>BootGroupIdentifier</td>
<td>The boot group identifier of a virtual machine.</td>
</tr>
<tr>
<td>Journal</td>
<td>Journal information of the virtual machine</td>
</tr>
<tr>
<td>DatastoreIdentifier</td>
<td>The identifier of the storage used by the journal for the virtual machine. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>Limitation</td>
<td>Information about the journal limitations.</td>
</tr>
<tr>
<td>HardLimitInMB</td>
<td>The maximum journal size, in MBs, for this virtual machine. O means unlimited. Integer values.</td>
</tr>
<tr>
<td>HardLimitInPercent</td>
<td>The percentage of the virtual machine volume size the journal can grow to. O means unlimited. Integer values.</td>
</tr>
<tr>
<td>WarningThresholdInMB</td>
<td>The journal size, in MBs, that generates a warning. O means unlimited. Integer values.</td>
</tr>
<tr>
<td>WarningThresholdInPercent</td>
<td>The percentage of the virtual machine volume size that generates a warning. O means unlimited. Integer values.</td>
</tr>
<tr>
<td>Nics</td>
<td>Information about NICs used by the virtual machine in the VPG.</td>
</tr>
<tr>
<td>Failover</td>
<td>Information about the networks used for failover by this virtual machine.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td>-</td>
</tr>
<tr>
<td>DnsSuffix</td>
<td>The DNS name excluding the host.</td>
</tr>
<tr>
<td>IpConfig</td>
<td>Information about the IP configuration of the recovery site used for failovers.</td>
</tr>
<tr>
<td>Gateway</td>
<td>The mask for the network.</td>
</tr>
<tr>
<td>IsDhcp</td>
<td>True: DHCP will be used for the virtual machine on the recovery site. False: A static IP address will be used for the virtual machine on the recovery site.</td>
</tr>
<tr>
<td>PrimaryDns</td>
<td>The IP address of the primary DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>SecondaryDns</td>
<td>The IP address of the alternate, secondary, DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>StaticIp</td>
<td>The IP of the restored virtual machine.</td>
</tr>
<tr>
<td>SubnetMask</td>
<td>The subnet mask for the network.</td>
</tr>
<tr>
<td>NetworkIdentifier</td>
<td>The network identifier of the network used for failover or move by this virtual machine.</td>
</tr>
<tr>
<td>ShouldReplaceMacAddress</td>
<td>True: The Media Access Control (MAC) address used on the protected site will be copied to the recovery site. False: The Media Access Control (MAC) address used on the protected site will not be copied to the recovery site.</td>
</tr>
<tr>
<td>FailoverTest</td>
<td>Information about the networks used for testing failover by this virtual machine.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td>-</td>
</tr>
<tr>
<td>DnsSuffix</td>
<td>The DNS name excluding the host.</td>
</tr>
<tr>
<td>IpConfig</td>
<td>Information about the IP configuration of the recovery site used for testing failovers.</td>
</tr>
<tr>
<td>Gateway</td>
<td>The mask for the network.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| IsDhcp                        | **True**: DHCP will be used for the virtual machine on the recovery site.  
|                               | **False**: A static IP address will be used for the virtual machine on the recovery site.  
| PrimaryDns                    | The IP address of the primary DNS server that handles Internet protocol mapping.                                                                 |
| SecondaryDns                  | The IP address of the alternate, secondary, DNS server that handles Internet protocol mapping.                                            |
| StaticIp                      | The IP of the restored virtual machine.                                                                                                                                                         |
| SubnetMask                    | The subnet mask for the network.                                                                                                                                                                  |
| NetworkIdentifier             | The network identifier of the network used for testing failover by this virtual machine.                                                                                                          |
| ShouldReplaceMacAddress       | **True**: The Media Access Control (MAC) address used on the protected site will be copied to the recovery site.  
|                               | **False**: The Media Access Control (MAC) address used on the protected site will not be copied to the recovery site.                                                                 |
| NicIdentifier                 | The identifier of the NIC for which settings are returned.                                                                                                                                       |
| Recovery                      | Information about the recovery site.                                                                                                                                                              |
| DatastoreClusterIdentifier    | The identifier of the datastore cluster used in the recovery site for the VM. Note: Only when the recovery site is a vSphere site.                                                               |
| DatastoreIdentifier           | The identifier of the storage. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.                                                        |
| FolderIdentifier              | The identifier of the folder used for recovery by the virtual machine. The identifier comprises the server identifier and the folder moref, with the format, serverid.moref. |
| HostClusterIdentifier         | The identifier of the host cluster that handles the replicated data. The identifier comprises the server identifier and the host cluster moref, with the format, serverid.moref. |
| HostIdentifier                | The identifier of the host that handles the replicated data. The identifier comprises the server identifier and the host moref, with the format, serverid.moref. |
| ResourcePoolIdentifier        | The identifier of the resource pool for the recovered virtual machines. The identifier comprises the server identifier and the resource pool moref, with the format, serverid.moref. |
| VmIdentifier                  | The identifier of the virtual machine. The identifier comprises the server identifier and the virtual machine moref, with the format, serverid.moref. |
| Volumes                       | Information about the volumes used by the virtual machine.                                                                                                                                 |
| Datastore                     | Information about the datastore used by the virtual machine.                                                                                                                                     |
| DatastoreClusterIdentifier    | The identifier of the datastore cluster used in the recovery site for the volume. Note: Only when the recovery site is a vSphere site.                                                               |
| DatastoreIdentifier           | The identifier of the storage. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.                                                        |
| IsThin                        | **True**: The recovery volumes are thin-provisioned.  
|                               | **False**: The recovery volumes are not thin-provisioned.                                                                                                                                       |
| ExistingVolume                | For future use.                                                                                                                                                                                  |
| DatastoreIdentifier           | For future use.                                                                                                                                                                                  |
| ExistedVmIdentifier           | For future use.                                                                                                                                                                                  |
Update the backup settings in a VPG settings object Request values for `https://zvm_ip:port/v1/vpgSettings/backup`.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mode</td>
<td>For future use.</td>
</tr>
<tr>
<td>Path</td>
<td>For future use.</td>
</tr>
<tr>
<td>IsSwap</td>
<td>True: The recovery disk is marked as a temp data disk. False: The recovery disk is not marked as a temp data disk.</td>
</tr>
<tr>
<td>VolumeIdentifier</td>
<td>The identifier of the volume.</td>
</tr>
</tbody>
</table>

The `RepositoryIdentifier` parameter is the identifier of the repository where offsite backups will be written.

The `RetentionPeriod` parameter is the length of time to keep offsite backups, up to a maximum of 12 months. Over time, Zerto reduces the number of stored offsite backups to save space.

<table>
<thead>
<tr>
<th>VALID VALUES FOR RetentionPeriod</th>
<th># OF BACKUPS SAVED WHEN RUN DAILY</th>
<th># OF BACKUPS SAVED WHEN RUN WEEKLY</th>
</tr>
</thead>
<tbody>
<tr>
<td>OneWeek</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>OneMonth</td>
<td>11</td>
<td>5</td>
</tr>
<tr>
<td>ThreeMonths</td>
<td>13</td>
<td>7</td>
</tr>
<tr>
<td>SixMonths</td>
<td>16</td>
<td>10</td>
</tr>
<tr>
<td>NineMonths</td>
<td>19</td>
<td>13</td>
</tr>
<tr>
<td>OneYear</td>
<td>22</td>
<td>16</td>
</tr>
</tbody>
</table>

The `SchedulerPeriod` parameter defines whether backups are created daily or weekly.

For an explanation of how Zerto reduces the number of offsite backups, see the Zerto Virtual Replication Administration Guide.

The `Retry` parameter is information about backup retries.

- `IntervalInMinutes`: How much time to wait, in minutes, after a backup job fails before running the backup job again.
- `Number`: The number of retries that will be attempted.
- `Retry`: True: The offsite backup job will rerun automatically if it fails. False: The offsite backup job will not rerun automatically if it fails.

The `Scheduler` parameter is offsite backup schedule settings.

- `DayOfWeek`: The day of the week that the offsite backups will run.
- `SchedulerPeriod`: Daily: The offsite backups will run every day. Weekly: The offsite backups will run once a week.
- `TimeOfDay`: The time of day when offsite backup jobs will run. The time is based on a 24-hour clock.

Update the basic settings in a VPG settings object Request values for `https://zvm_ip:port/v1/vpgSettings/basic`.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>JournalHistoryInHours</td>
<td>The time that all write commands are saved in the journal. The value is between 1 and 336 (14 days).</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the VPG.</td>
</tr>
</tbody>
</table>
### VPG Management API

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority</td>
<td>The priority specified for the VPG. Possible values are:</td>
</tr>
<tr>
<td></td>
<td><strong>Low</strong>: The VPG has a low priority for transferring data.</td>
</tr>
<tr>
<td></td>
<td><strong>Medium</strong>: The VPG has a medium priority for transferring data.</td>
</tr>
<tr>
<td></td>
<td><strong>High</strong>: The VPG has a high priority for transferring data.</td>
</tr>
<tr>
<td>ProtectedSiteIdentifier</td>
<td>The identifier of the source site where the VPG virtual machines will be protected. This is the site where the API runs.</td>
</tr>
<tr>
<td>RecoverySiteIdentifier</td>
<td>The identifier of the target site where the VPG virtual machines will be recovered.</td>
</tr>
<tr>
<td>RpoInSeconds</td>
<td>The maximum desired time between each automatic checkpoint being written to the journal before an alert is issued.</td>
</tr>
<tr>
<td>ServiceProfileIdentifier</td>
<td>The identifier of the service profile to use for the VPG when a Zerto Cloud Manager is used.</td>
</tr>
<tr>
<td>TestIntervalInMinutes</td>
<td>The time, in minutes, recommended between testing the integrity of the VPG. A warning is issued if a test is not done within this time frame. Possible values are:</td>
</tr>
<tr>
<td></td>
<td><strong>0</strong>: No testing is expected.</td>
</tr>
<tr>
<td></td>
<td><strong>43200</strong> or <strong>null</strong>: Testing is expected monthly.</td>
</tr>
<tr>
<td></td>
<td><strong>131040</strong>: Testing is expected every three months.</td>
</tr>
<tr>
<td></td>
<td><strong>262080</strong>: Testing is expected every six months.</td>
</tr>
<tr>
<td></td>
<td><strong>394560</strong>: Testing is expected every nine months.</td>
</tr>
<tr>
<td></td>
<td><strong>525600</strong>: Testing is expected every twelve months.</td>
</tr>
<tr>
<td>UseWanCompression</td>
<td>True: Data will be compressed before sending it to the recovery site.</td>
</tr>
<tr>
<td></td>
<td>False: Data will not be compressed before sending it to the recovery site.</td>
</tr>
<tr>
<td>ZorgIdentifier</td>
<td>The identifier of the ZORG, Zerto organization, defined in the Zerto Cloud Manager.</td>
</tr>
</tbody>
</table>

### Update the bootgroup settings in a VPG settings object

Request values for `https://zvm_ip:port/v1/vpgSettings/bootgroup`.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>BootGroups</td>
<td>-</td>
</tr>
<tr>
<td>BootDelayInSeconds</td>
<td>Specifies the delay, in seconds, between starting up the virtual machines in this group and starting up the virtual machines in the next group.</td>
</tr>
<tr>
<td>BootGroupIdentifier</td>
<td>The identifier of a boot group.</td>
</tr>
<tr>
<td>Name</td>
<td>The name of a boot group.</td>
</tr>
</tbody>
</table>

### Update the journal settings in a VPG settings object

Request values for `https://zvm_ip:port/v1/vpgSettings/journal`.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>DatastoreIdentifier</td>
<td>The identifier of the storage used by the journal. The identifier comprises the server identifier and the storage moref, with the format, <code>serverid.moref</code>.</td>
</tr>
<tr>
<td>Limitation</td>
<td>Information about the journal limitations.</td>
</tr>
<tr>
<td>HardLimitInMB</td>
<td>The maximum journal size in MBs. O means unlimited. Integer values.</td>
</tr>
<tr>
<td>HardLimitInPercent</td>
<td>The percentage of the virtual machine volume size the journal can grow to. O means unlimited. Integer values.</td>
</tr>
<tr>
<td>WarningThresholdInMB</td>
<td>The journal size, in MBs, that generates a warning that the journal is nearing its hard limit. O means unlimited. Integer values.</td>
</tr>
<tr>
<td>WarningThresholdInPercent</td>
<td>The percentage of the virtual machine volume size that generates a warning, O means unlimited. Integer values.</td>
</tr>
</tbody>
</table>
Update the network settings in a VPG settings object Request values for https://zvm_ip:port/v1/vpgSettings/networks.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Failover</td>
<td>Information about the networks used for failover.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td></td>
</tr>
<tr>
<td>DefaultNetworkIdentifier</td>
<td>The network identifier of the network to use during a failover or move operation in which the recovered virtual machines will run.</td>
</tr>
<tr>
<td>FailoverTest</td>
<td>Information about the networks used for testing failover.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td></td>
</tr>
<tr>
<td>DefaultNetworkIdentifier</td>
<td>The network identifier of the network to use when testing the failover of virtual machines in the recovery site.</td>
</tr>
</tbody>
</table>

Update the recovery settings in a VPG settings object Request values for https://zvm_ip:port/v1/vpgSettings/recovery.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>DefaultDatastoreClusterIdentifier</td>
<td>The identifier of the default datastore cluster used in the recovery site. Note: Only when the recovery site is a vSphere site.</td>
</tr>
<tr>
<td>DefaultDatastoreIdentifier</td>
<td>The identifier of the default storage where the metadata files for the virtual machines are stored, such as the vmx or vhdx files. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultFolderIdentifier</td>
<td>The identifier of the default host cluster that handles the replicated data. The identifier comprises the server identifier and the folder moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultHostClusterIdentifier</td>
<td>The identifier of the default host that handles the replicated data. The identifier comprises the server identifier and the host cluster moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultHostIdentifier</td>
<td>The identifier of the resource pool for the recovered virtual machines. The identifier comprises the server identifier and the host moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>ResourcePoolIdentifier</td>
<td>The identifier of the storage cluster of the VRA virtual machine. The identifier comprises the server identifier and the resource pool moref, with the format, serverid.moref.</td>
</tr>
</tbody>
</table>

Update the scripting settings in a VPG settings object Request values for https://zvm_ip:port/v1/vpgSettings/scripting.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>PostBackup</td>
<td>Information about scripts that are run after an offsite backup is performed.</td>
</tr>
<tr>
<td>Command</td>
<td>The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site.</td>
</tr>
<tr>
<td>Parameters</td>
<td>Parameters to pass to the script.</td>
</tr>
<tr>
<td>TimeoutInSeconds</td>
<td>The time-out, in seconds, for the script to run.</td>
</tr>
<tr>
<td>PostRecovery</td>
<td>Information about scripts that are run after a recovery operation is performed.</td>
</tr>
<tr>
<td>Command</td>
<td>The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site.</td>
</tr>
<tr>
<td>Parameters</td>
<td>Parameters to pass to the script.</td>
</tr>
<tr>
<td>TimeoutInSeconds</td>
<td>The time-out, in seconds, for the script to run.</td>
</tr>
<tr>
<td>PreRecovery</td>
<td>Information about scripts that are run before a recovery operation is performed.</td>
</tr>
</tbody>
</table>
**Update the virtual machine settings in a VPG settings object** Request values for https://zvm_ip_port/v1/vpgSettings/vms/{vmIdentifier}/{vmIdentifier}.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BootGroupIdentifier</strong></td>
<td>The boot group identifier of a virtual machine.</td>
</tr>
<tr>
<td><strong>Journal</strong></td>
<td>Journal information of the virtual machine</td>
</tr>
<tr>
<td><strong>Dat astoreIdentifier</strong></td>
<td>The identifier of the storage used by the journal for the virtual machine. The identifier comprises the server identifier and the storage moref, with the format, <em>serverid.moref</em>.</td>
</tr>
<tr>
<td><strong>Limitation</strong></td>
<td>Information about the journal limitations.</td>
</tr>
<tr>
<td><strong>HardLimitInMB</strong></td>
<td>The maximum journal size, in MBs, for this virtual machine. O means unlimited. Integer values.</td>
</tr>
<tr>
<td><strong>HardLimitInPercent</strong></td>
<td>The percentage of the virtual machine volume size the journal can grow to. O means unlimited. Integer values.</td>
</tr>
<tr>
<td><strong>WarningThresholdInMB</strong></td>
<td>The journal size, in MBs, that generates a warning. O means unlimited. Integer values.</td>
</tr>
<tr>
<td><strong>WarningThresholdInPercent</strong></td>
<td>The percentage of the virtual machine volume size that generates a warning. O means unlimited. Integer values.</td>
</tr>
<tr>
<td><strong>Nics</strong></td>
<td>Information about NICs used by the virtual machine in the VPG.</td>
</tr>
<tr>
<td><strong>Failover</strong></td>
<td>Information about the networks used for failover by this virtual machine.</td>
</tr>
<tr>
<td><strong>Hypervisor</strong></td>
<td>-</td>
</tr>
<tr>
<td><strong>Dns Suffix</strong></td>
<td>The DNS name excluding the host.</td>
</tr>
<tr>
<td><strong>Ip Config</strong></td>
<td>Information about the IP configuration of the recovery site used for failovers.</td>
</tr>
<tr>
<td><strong>Gateway</strong></td>
<td>The mask for the network.</td>
</tr>
<tr>
<td><strong>Is Dhcp</strong></td>
<td>True: DHCP will be used for the virtual machine on the recovery site.</td>
</tr>
<tr>
<td><strong>Primary Dns</strong></td>
<td>False: A static IP address will be used for the virtual machine on the recovery site.</td>
</tr>
<tr>
<td><strong>Secondary Dns</strong></td>
<td>The IP address of the primary DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td><strong>Static Ip</strong></td>
<td>The IP of the restored virtual machine.</td>
</tr>
<tr>
<td><strong>Subnet Mask</strong></td>
<td>The subnet mask for the network.</td>
</tr>
<tr>
<td><strong>Network Identifier</strong></td>
<td>The network identifier of the network used for failover or move by this virtual machine.</td>
</tr>
<tr>
<td><strong>Should Replace Mac Address</strong></td>
<td>True: The Media Access Control (MAC) address used on the protected site will be copied to the recovery site.</td>
</tr>
<tr>
<td><strong>Failover Test</strong></td>
<td>False: The Media Access Control (MAC) address used on the protected site will not be copied to the recovery site.</td>
</tr>
<tr>
<td><strong>Hypervisor</strong></td>
<td>-</td>
</tr>
<tr>
<td><strong>Dns Suffix</strong></td>
<td>The DNS name excluding the host.</td>
</tr>
<tr>
<td><strong>Ip Config</strong></td>
<td>Information about the IP configuration of the recovery site used for failovers.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Gateway</td>
<td>The mask for the network.</td>
</tr>
<tr>
<td>IsDhcp</td>
<td><strong>True</strong>: DHCP will be used for the virtual machine on the recovery site. <strong>False</strong>: A static IP address will be used for the virtual machine on the recovery site.</td>
</tr>
<tr>
<td>PrimaryDns</td>
<td>The IP address of the primary DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>SecondaryDns</td>
<td>The IP address of the alternate, secondary, DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>StaticIp</td>
<td>The IP of the restored virtual machine.</td>
</tr>
<tr>
<td>SubnetMask</td>
<td>The subnet mask for the network.</td>
</tr>
<tr>
<td>NetworkIdentifier</td>
<td>The network identifier of the network used for testing failover by this virtual machine.</td>
</tr>
<tr>
<td>ShouldReplaceMacAddress</td>
<td><strong>True</strong>: The Media Access Control (MAC) address used on the protected site will be copied to the recovery site. <strong>False</strong>: The Media Access Control (MAC) address used on the protected site will not be copied to the recovery site.</td>
</tr>
<tr>
<td>NicIdentifier</td>
<td>The identifier of the NIC for which settings are returned.</td>
</tr>
<tr>
<td>Recovery</td>
<td>Information about the recovery site.</td>
</tr>
</tbody>
</table>
| DatastoreClusterIdentifier | The identifier of the datastore cluster used in the recovery site for the VM.  
Note: Only when the recovery site is a vSphere site.                                                                                       |
| DatastoreIdentifier | The identifier of the storage. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.                                                                     |
| FolderIdentifier  | The identifier of the folder used for recovery by the virtual machine. The identifier comprises the server identifier and the folder moref, with the format, serverid.moref.                      |
| HostClusterIdentifier | The identifier of the host cluster that handles the replicated data. The identifier comprises the server identifier and the host cluster moref, with the format, serverid.moref. |
| HostIdentifier    | The identifier of the host that handles the replicated data. The identifier comprises the server identifier and the host moref, with the format, serverid.moref.                                             |
| ResourcePoolIdentifier | The identifier of the resource pool for the recovered virtual machines. The identifier comprises the server identifier and the resource pool moref, with the format, serverid.moref.                           |
| VmIdentifier      | The identifier of the virtual machine. The identifier comprises the server identifier and the virtual machine moref, with the format, serverid.moref.                                                      |
| Volumes           | Information about the volumes used by the virtual machine.                                                                                                                                                   |
| Datastore         | Information about the datastore used by the virtual machine.                                                                                                                                                 |
| DatastoreClusterIdentifier | The identifier of the datastore cluster used in the recovery site for the volume.  
Note: Only when the recovery site is a vSphere site.                                                                                           |
| DatastoreIdentifier | The identifier of the storage. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.                                                                    |
| IsThin            | **True**: The recovery volumes are thin-provisioned. **False**: The recovery volumes are not thin-provisioned.                                                                                           |
| ExistingVolume    | For future use.                                                                                                                                                                                              |
| DatastoreIdentifier | For future use.                                                                                                                                                                                           |
Update the virtual machine settings in a VPG settings object

Request values for

```
https://zvm_ip:port/v1/vpgSettings/vms/{vmIdentifier}/{vmIdentifier}/nics/{nicIdentifier}
```

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExistedVmIdentifier</td>
<td>For future use.</td>
</tr>
<tr>
<td>Mode</td>
<td>For future use.</td>
</tr>
<tr>
<td>Path</td>
<td>For future use.</td>
</tr>
<tr>
<td>IsSwap</td>
<td><strong>True</strong>: The recovery disk is marked as a temp data disk. <strong>False</strong>: The recovery disk is not marked as a temp data disk.</td>
</tr>
<tr>
<td>VolumeIdentifier</td>
<td>The identifier of the volume.</td>
</tr>
</tbody>
</table>

**PARAMETER**

**DESCRIPTION**

- **Failover**: Information about the networks used for failover by this virtual machine.
- **Hypervisor**: The DNS name excluding the host.
- **DnsSuffix**: The DNS name excluding the host.
- **IpConfig**: Information about the IP configuration of the recovery site used for failovers.
- **Gateway**: The mask for the network.
- **IsDhcp**: **True**: DHCP will be used for the virtual machine on the recovery site. **False**: A static IP address will be used for the virtual machine on the recovery site.
- **PrimaryDns**: The IP address of the primary DNS server that handles Internet protocol mapping.
- **SecondaryDns**: The IP address of the alternate, secondary, DNS server that handles Internet protocol mapping.
- **StaticIp**: The IP of the restored virtual machine.
- **SubnetMask**: The subnet mask for the network.
- **NetworkIdentifier**: The network identifier of the network used for failover or move by this virtual machine.
- **ShouldReplaceMacAddress**: **True**: The Media Access Control (MAC) address used on the protected site will be copied to the recovery site. **False**: The Media Access Control (MAC) address used on the protected site will not be copied to the recovery site.
- **FailoverTest**: Information about the networks used for testing failover by this virtual machine.
- **Hypervisor**: The DNS name excluding the host.
- **DnsSuffix**: The DNS name excluding the host.
- **IpConfig**: Information about the IP configuration of the recovery site used for testing failovers.
- **Gateway**: The mask for the network.
- **IsDhcp**: **True**: DHCP will be used for the virtual machine on the recovery site. **False**: A static IP address will be used for the virtual machine on the recovery site.
- **PrimaryDns**: The IP address of the primary DNS server that handles Internet protocol mapping.
- **SecondaryDns**: The IP address of the alternate, secondary, DNS server that handles Internet protocol mapping.
- **StaticIp**: The IP of the restored virtual machine.
- **SubnetMask**: The subnet mask for the network.
- **NetworkIdentifier**: The network identifier of the network used for testing failover by this virtual machine.
Update the virtual machine settings in a VPG settings object Request values for
https://zvm_ip:port/v1/vpgSettings/vms/{vmIdentifier}/volumes/{volumeId}.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ShouldReplaceMacAddress</td>
<td><strong>True</strong>: The Media Access Control (MAC) address used on the protected site will be copied to the recovery site. <strong>False</strong>: The Media Access Control (MAC) address used on the protected site will not be copied to the recovery site.</td>
</tr>
<tr>
<td>NicIdentifier</td>
<td>The identifier of the NIC for which settings are returned.</td>
</tr>
<tr>
<td><strong>PARAMETER</strong></td>
<td><strong>DESCRIPTION</strong></td>
</tr>
<tr>
<td><strong>Datastore</strong></td>
<td>Information about the datastore used by the virtual machine.</td>
</tr>
<tr>
<td>DatastoreClusterIdentifier</td>
<td>The identifier of the datastore cluster used in the recovery site for the volume. Note: Only when the recovery site is a vSphere site.</td>
</tr>
<tr>
<td>DatastoreIdentifier</td>
<td>The identifier of the storage. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>IsThin</td>
<td><strong>True</strong>: The recovery volumes are thin-provisioned. <strong>False</strong>: The recovery volumes are not thin-provisioned.</td>
</tr>
<tr>
<td>ExistingVolume</td>
<td>For future use.</td>
</tr>
<tr>
<td>DatastoreIdentifier</td>
<td>For future use.</td>
</tr>
<tr>
<td>ExistedVmIdentifier</td>
<td>For future use.</td>
</tr>
<tr>
<td>Mode</td>
<td>For future use.</td>
</tr>
<tr>
<td>Path</td>
<td>For future use.</td>
</tr>
<tr>
<td>IsSwap</td>
<td><strong>True</strong>: The recovery disk is marked as a temp data disk. <strong>False</strong>: The recovery disk is not marked as a temp data disk.</td>
</tr>
<tr>
<td>VolumeIdentifier</td>
<td>The identifier of the volume.</td>
</tr>
</tbody>
</table>

Response Format

The response bodies are empty.

**VPG Settings: DELETE**

Deletes all or part of the VPG settings object.

See also:

- “VPG Settings: GET”, on page 119
- “VPG Settings: POST”, on page 142
- “VPG Settings: PUT”, on page 155
URL

Delete a VPG settings object  https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}
Delete backup settings from a VPG  https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/backup
Delete basic settings from a VPG  https://zvm_ip:port/v1/{vpgSettingsIdentifier}/basic
Delete boot settings from a VPG  https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/bootgroup
Delete journal settings from a VPG  https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/journal
Delete network settings from a VPG  https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/networks
Delete recovery settings from a VPG  https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/recovery
Delete scripts from a VPG  https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/scripting
Delete a VM from a VPG  https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}
Reset NIC settings object to Default  https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/nics/{nicIdentifier}

Where:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>zvm_ip</td>
<td>The IP address of the Zerto Virtual Manager where the API is run.</td>
</tr>
<tr>
<td>port</td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
<tr>
<td>vpgSettingsIdentifier</td>
<td>The identifier of the VPG settings object.</td>
</tr>
</tbody>
</table>

Request Format

The request bodies are empty.

Response Format

The response bodies are empty.

Code Examples

For complete code examples, see “Code Samples”, on page 22. For specific use of the vpgSettings API, see “Managing VPGs”, on page 12.

VRAs API

/v1/vras returns information about VRAs or installs or performs actions on a specific VRA. The following API are available:

- “VRAs: GET”, below
- “VRAs: POST”, on page 181
- “VRAs: PUT”, on page 183
- “VRAs: DELETE”, on page 185

<table>
<thead>
<tr>
<th>PURPOSE</th>
<th>METHOD</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Information for all VRAs</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/vras</td>
</tr>
<tr>
<td>Information for one VRA</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/vras/{vraIdentifier}</td>
</tr>
<tr>
<td>PURPOSE</td>
<td>METHOD</td>
<td>URL</td>
</tr>
<tr>
<td>------------------------------</td>
<td>--------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>Valid VRA IP Configuration Types</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/vras/ipconfigurationtypes</td>
</tr>
<tr>
<td>Valid VRA Statuses</td>
<td>GET</td>
<td>https://zvm_ip:port/v1/vras/statuses</td>
</tr>
<tr>
<td>Install a VRA</td>
<td>POST</td>
<td>https://zvm_ip:port/v1/vras</td>
</tr>
<tr>
<td>Upgrade a Group of VRAs</td>
<td>POST</td>
<td>https://zvm_ip:port/v1/vras/upgrade</td>
</tr>
<tr>
<td>Upgrade a Specific VRA</td>
<td>POST</td>
<td>https://zvm_ip:port/v1/vras/{vraIdentifier}/upgrade</td>
</tr>
<tr>
<td>Edit a VRA</td>
<td>PUT</td>
<td>https://zvm_ip:port/v1/vras/{vraIdentifier}</td>
</tr>
<tr>
<td>Delete a VRA</td>
<td>DELETE</td>
<td>https://zvm_ip:port/v1/vras/{vraIdentifier}</td>
</tr>
</tbody>
</table>

**HTTP Methods**

GET, POST, PUT, DELETE

**Security**

The API is exposed over HTTPS. Client code must use the `x-zerto-session` HTTP authorization header.

**See Also**

Starting a session: Session: POST

**Format**

Json, XML

**VRAs: GET**

Returns information about VRAs.

**URL**

All VRAs  
https://zvm_ip:port/v1/vras

Filtered VRAs  
https://zvm_ip:port/v1/vras?vraName={VRANAME}&status={STATUS}&vraVersion={VRAVERSION}&hostVersion={HOSTVERSION}&ipAddress={IPADDRESS}&vraGroup={VRAGROUP}&datastoreName={DATASTORENAME}&datastoreClusterName={DATASTORECLUSTERNAME}&networkName={NETWORKNAME}

Single VRA  
https://zvm_ip:port/v1/vras/{vraIdentifier}

Valid VRA IP configuration types Checkpoints  
https://zvm_ip:port/v1/vras/ipconfigurationtypes

Valid VRA statuses  
https://zvm_ip:port/v1/vras/statuses

Where:

<table>
<thead>
<tr>
<th>zvm_ip</th>
<th>The IP address of the Zerto Virtual Manager where the API is run.</th>
</tr>
</thead>
<tbody>
<tr>
<td>port</td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
<tr>
<td>vraIdentifier</td>
<td>The identifier of the VRA for which information is returned.</td>
</tr>
</tbody>
</table>
Filters

Filters are optional and any combination of filters is valid. When no filter is specified, all VRAs are returned. Filters are not case-sensitive.

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vraName</td>
<td>The name of the VRA.</td>
</tr>
<tr>
<td>status</td>
<td>The status of the VRA.</td>
</tr>
<tr>
<td>vraVersion</td>
<td>The version of the VRA.</td>
</tr>
<tr>
<td>hostVersion</td>
<td>The version of the host where the VRA is installed.</td>
</tr>
<tr>
<td>ipAddress</td>
<td>The IP address of the VRA.</td>
</tr>
<tr>
<td>vraGroup</td>
<td>The group the VRA belongs to. VRAs can be grouped together when VRAs use different networks so they can be grouped by network, for example when the same hypervisor manager supports two datacenters with separate networks and you are replicating from one datacenter to the second datacenter.</td>
</tr>
<tr>
<td>datastoreName</td>
<td>The name of the datastore the VRA uses for recovery data.</td>
</tr>
<tr>
<td>datastoreClusterName</td>
<td>The name of the datastore cluster the VRA uses for recovery data.</td>
</tr>
<tr>
<td>networkName</td>
<td>The network the VRA belongs to.</td>
</tr>
</tbody>
</table>

Request Format

The request body is empty.

Json Response Format

The following is an example response Json body for https://zvm_ip:port/v1/vras and, without the array statement, for https://zvm_ip:port/v1/vras/{vraIdentifier}.

```json
{
    "DatastoreClusterName": "String content",
    "DatastoreIdentifier": "String content",
    "DatastoreName": "String content",
    "HostIdentifier": "String content",
    "HostVersion": "String content",
    "IpAddress": "String content",
    "Link": {
        "href": "String content",
        "identifier": "String content",
        "rel": "String content",
        "type": "String content"
    },
    "Link__x007B_0_x007D_": {
        "href": "String content",
        "rel": "String content",
        "type": "String content"
    },
    "MemoryInGB": 2147483647,
    "NetworkIdentifier": "String content",
    "NetworkName": "String content",
    "Progress": 2147483647,
    "ProtectedCounters": {
        "Vms": 2147483647,
        "Volumes": 2147483647,
        "Vpgs": 2147483647
    }
}
```
The following is an example response JSON body for https://zvm_ip:port/v1/vras/ipconfigurationtypes and for https://zvm_ip:port/v1/vras/statuses.

```json
{"RecoveryCounters": {
    "Vms":2147483647,
    "Volumes":2147483647,
    "Vpgs":2147483647
},

"SelfProtectedVpgs":2147483647,
"Status":0,
"VraGroup": "String content",
"VraIdentifier":18446744073709551615,
"VraName": "String content",

"VraNetworkDataApi": {
    "DefaultGateway": "String content",
    "SubnetMask": "String content",
    "VraIPAddress": "String content",
    "VraIPConfigurationTypeApi": "String content"
},

"VraVersion": "String content"}
```

### XML Response Format

For the XML response format, see “VRAs API GET Method Response Formats”, on page 273.

### Response Values


<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>DatastoreClusterName</td>
<td>The name of the datastore cluster of the VRA virtual machine.</td>
</tr>
<tr>
<td>DatastoreIdentifier</td>
<td>The identifier of the vSphere datastore/Hyper-V storage of the VRA virtual machine. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DatastoreName</td>
<td>The name of the vSphere datastore/Hyper-V storage of the VRA virtual machine.</td>
</tr>
<tr>
<td>HostVersion</td>
<td>The version of the host on which the VRA is installed.</td>
</tr>
<tr>
<td>IpAddress</td>
<td>The IP address of the VRA.</td>
</tr>
<tr>
<td>Link</td>
<td>The link details.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used.</td>
</tr>
<tr>
<td>identifier</td>
<td>The VRA identifier.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level of the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>Link</td>
<td>The link details.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level of the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>MemoryInGB</td>
<td>The amount of reserved memory allocated to the VRA when it was installed.</td>
</tr>
<tr>
<td>NetworkIdentifier</td>
<td>The identifier of the network used to access the VRA.</td>
</tr>
<tr>
<td>NetworkName</td>
<td>The name of the network used to access the VRA.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>--------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Progress</td>
<td>The percentage progress when installing the VRA.</td>
</tr>
<tr>
<td>ProtectionCounters</td>
<td>Details for a VRA on the protection site.</td>
</tr>
<tr>
<td>Vms</td>
<td>The number of VMs protected on the host with the VRA installed.</td>
</tr>
<tr>
<td>Volumes</td>
<td>The volumes used by the VRA.</td>
</tr>
<tr>
<td>Vpgs</td>
<td>The number of VPGs using the VRA.</td>
</tr>
<tr>
<td>RecoveryCounters</td>
<td>Details for a VRA on the recovery site.</td>
</tr>
<tr>
<td>Vms</td>
<td>The number of VMs recovered to the host with the VRA installed.</td>
</tr>
<tr>
<td>Volumes</td>
<td>The volumes used by the VRA.</td>
</tr>
<tr>
<td>Vpgs</td>
<td>The number of VPGs using the VRA.</td>
</tr>
<tr>
<td>SelfProtectedVpgs</td>
<td>The number of VPGs being protected and recovered on the same site.</td>
</tr>
<tr>
<td>Status</td>
<td>The status of the VRA. Possible values are (Json/XML):</td>
</tr>
<tr>
<td></td>
<td><strong>Installed:</strong> The VRA is installed.</td>
</tr>
<tr>
<td></td>
<td><strong>UnsupportedEsxVersion:</strong> The VRA cannot be installed on the ESX/ESXi host as the host version is not supported.</td>
</tr>
<tr>
<td></td>
<td><strong>NotInstalled:</strong> A VRA is not installed.</td>
</tr>
<tr>
<td></td>
<td><strong>Installing:</strong> The VRA is being installed.</td>
</tr>
<tr>
<td></td>
<td><strong>Removing:</strong> The VRA is being removed.</td>
</tr>
<tr>
<td></td>
<td><strong>InstallationError:</strong> The installation of the VRA failed.</td>
</tr>
<tr>
<td></td>
<td><strong>HostPasswordChanged:</strong> The password used to access the host has changed.</td>
</tr>
<tr>
<td></td>
<td><strong>UpdatingIpSettings:</strong> The IP settings of the VRA are updated.</td>
</tr>
<tr>
<td></td>
<td><strong>DuringChangeHost:</strong> The host of the VRA disks is being changed.</td>
</tr>
<tr>
<td>VraGroup</td>
<td>The group the VRA belongs to, used when VRAs use different networks so they can be grouped by network, for example when the same vCenter Server supports two datacenters with separate networks and you are replicating from one datacenter to the second datacenter.</td>
</tr>
<tr>
<td>VralIdentifier</td>
<td>The internal VRA identifier.</td>
</tr>
<tr>
<td>VraName</td>
<td>The name used to identify the VRA in the UI.</td>
</tr>
<tr>
<td>VraNetworkDataApi</td>
<td>The VRA network details.</td>
</tr>
<tr>
<td>DefaultGateway</td>
<td>The default gateway for the VRA network.</td>
</tr>
<tr>
<td>SubnetMask</td>
<td>The subnet mask for the VRA network. The default value is 255.255.255.0.</td>
</tr>
<tr>
<td>VralIPAddress</td>
<td>The IP address for the VRA.</td>
</tr>
<tr>
<td>VralIPConfigurationTypeApi</td>
<td>The IP configuration used for the VRA. Possible values are:</td>
</tr>
<tr>
<td></td>
<td><strong>Dhcp:</strong> The VRA IP address is allocated via a DHCP server.</td>
</tr>
<tr>
<td></td>
<td><strong>Static:</strong> The VRA is installed with a static IP that was assigned when the VRA was installed.</td>
</tr>
<tr>
<td>VraVersion</td>
<td>The storage provisioned for the virtual machine in the recovery site.</td>
</tr>
</tbody>
</table>

**Response: Description**

Possible statuses for a VRA:

- **Installed**: The VRA is installed.
- **UnsupportedEsxVersion**: The VRA cannot be installed on the ESX/ESXi host as the host version is not supported.
- **NotInstalled**: A VRA is not installed.
- **Installing**: The VRA is being installed.
- **Removing**: The VRA is being removed.
- **InstallationError**: The installation of the VRA failed.
- **HostPasswordChanged**: The password used to access the host has changed.
- **UpdatingIpSettings**: The IP settings of the VRA are updated.
- **DuringChangeHost**: The host of the VRA disks is being changed.


**Response: Description**

Possible IP configurations for the VRA:

- **Dhcp**: The VRA IP address is allocated via a DHCP server.
- **Static**: The VRA is installed with a static IP that was assigned when the VRA was installed.

**VRAs: POST**

Installs a VRA or upgrades a specific VRA.

**URL**

- **Install VRA**
  - https://zvm_ip:port/v1/vras
- **Upgrade a Group of VRAs**
  - https://zvm_ip:port/v1/vras/upgrade
- **Upgrade a Specific VRA**
  - https://zvm_ip:port/v1/vras/{vraIdentifier}/upgrade

Where:

- **zvm_ip**: The IP address of the Zerto Virtual Manager where the API is run.
- **port**: The port to access the Zerto Virtual Manager. The default port is 9669.
- **vraIdentifier**: The identifier of the VRA to be upgraded.

**Json Request Format**

The following is an example request Json body for https://zvm_ip:port/v1/vras.

```json
{
  "DatastoreIdentifier": "String content",
  "GroupName": "String content",
  "HostIdentifier": "String content",
  "HostRootPassword": "String content",
  "MemoryInGb": 2147483647,
  "NetworkIdentifier": "String content",
  "UsePublicKeyInsteadOfCredentials": Boolean,
  "VraNetworkDataApi": {
    "DefaultGateway": "String content",
    "SubnetMask": "String content",
    "VraIPAddress": "String content",
    "VraIPConfigurationTypeApi": "String content"
  }
}
```
The following is an example request Json body for https://zvm_ip:port/v1/vras/upgrade.

```json
{
    "VraIdentifiers":
    [
        "String content",
        "String content",
        ...
        "String content"
    ]
}
```

The request body for https://zvm_ip:port/v1/vras/{vraIdentifier}/upgrade is empty.

**XML Request Format**

For the XML request format, see “VRAs API POST Method Request and Response Formats”, on page 274.

**Request Values**

**Install a VRA**  Request values for https://zvm_ip:port/v1/vras

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>DatastoreIdentifier</td>
<td>The identifier of the vSphere datastore/Hyper-V storage for the VRA virtual machine. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>GroupName</td>
<td>Optional: The group the VRA belongs to, used when VRAs use different networks so they can be grouped by network, for example when the same vCenter Server supports two datacenters with separate networks and you are replicating from one datacenter to the second datacenter. If you want to create a new VRA group, enter the name of the new group.</td>
</tr>
<tr>
<td>HostIdentifier</td>
<td>The identifier of the host on which the VRA will be installed.</td>
</tr>
<tr>
<td>HostRootPassword</td>
<td>vSphere environments: The password used to access the host for the root user. This field is only required for VMware ESXi 4.x and 5.x hosts. A value is required if UsePublicKeyInsteadOfCredentials is set to false.</td>
</tr>
<tr>
<td>MemoryInGB</td>
<td>The amount of memory to allocate to the VRA. The amount determines the maximum buffer size for the VRA for buffering IOs written by the protected virtual machines, before the writes are sent over the network to the recovery VRA. The recovery VRA also buffers the incoming IOs until they are written to the journal. If a buffer becomes full, a Bitmap Sync is performed after space is freed up in the buffer. The value can be in 1-16GB.</td>
</tr>
<tr>
<td>NetworkIdentifier</td>
<td>The identifier of the network to use to access the VRA.</td>
</tr>
</tbody>
</table>
| UsePublicKeyInsteadOfCredentials | vSphere environments: Optional. The default is true.  
**True**: A password is not needed to access the host. From ESXi 5.5, by default, Zerto Virtual Manager uses a vSphere Installation Bundle, VIB, to connect to the host, which does not require a password.  
**False**: A password is needed to access the host. For ESXi/i versions earlier than 5.5, this parameter must be set to false and a value specified for HostRootPassword. |
| VraNetworkDataApi             | The VRA network details. |
| DefaultGateway                | The default gateway for the VRA network when VraIPConfigurationTypeApi is set to Static. |
| SubnetMask                    | The subnet mask for the VRA network when VraIPConfigurationTypeApi is set to Static. |
| VraIPAddress                  | The IP address for the VRA when VraIPConfigurationTypeApi is set to Static. |
| VraIPConfigurationTypeApi     | The IP configuration used for the VRA. Possible values are:  
**Dhcp**: The VRA IP address is allocated via a DHCP server.  
**Static**: The VRA is installed with a static IP that was assigned when the VRA was installed. |
**VRAs: PUT**

Enables editing a specific VRA.

**URL**

```
Edit VRA  https://zvm_ip:port/v1/vras/{vraIdentifier}
```

Where:

- **zvm_ip**: The IP address of the Zerto Virtual Manager where the API is run.
- **port**: The port to access the Zerto Virtual Manager. The default port is 9669.
- **vraIdentifier**: The identifier of the VRA to be edited.
Json Request Format

The following is an example request Json body for `https://zvm_ip:port/v1/vras`.

```json
{
  "GroupName": "String content",
  "HostRootPassword": "String content",
  "UsePublicKeyInsteadOfCredentials": Boolean,
  "VraNetworkDataApi": {
    "DefaultGateway": "String content",
    "SubnetMask": "String content",
    "VraIPAddress": "String content",
    "VraIPConfigurationTypeApi": "String content"
  }
}
```

XML Request Format

For the XML request format, see “VRAs API PUT Method Request and Response Formats”, on page 275.

Request Values

Request values for `https://zvm_ip:port/v1/vras`

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>GroupName</td>
<td>Optional: The group the VRA belongs to, used when VRAs use different networks so they can be grouped by network, for example when the same vCenter Server supports two datacenters with separate networks and you are replicating from one datacenter to the second datacenter. If you want to create a new VRA group, enter the name of the new group.</td>
</tr>
<tr>
<td>HostRootPassword</td>
<td>vSphere environments: The password used to access the host for the root user. This field is only required for VMware ESXi 4.x and 5.x hosts. A value is required if <code>UsePublicKeyInsteadOfCredentials</code> is set to <code>false</code>.</td>
</tr>
<tr>
<td>UsePublicKeyInsteadOfCredentials</td>
<td>vSphere environments: Optional. The default is <code>true</code>. <strong>True</strong>: A password is not needed to access the host. From ESXi 5.5, by default, Zerto Virtual Manager uses a vSphere Installation Bundle, VIB, to connect to the host, which does not require a password. <strong>False</strong>: A password is needed to access the host. For ESX/i versions earlier than 5.5, this parameter must be set to <code>false</code> and a value specified for <code>HostRootPassword</code>.</td>
</tr>
<tr>
<td>VraNetworkDataApi</td>
<td>Optional: The VRA network details.</td>
</tr>
<tr>
<td>DefaultGateway</td>
<td>Optional: The default gateway for the VRA network when <code>VraIPConfigurationTypeApi</code> is set to <code>Static</code>.</td>
</tr>
<tr>
<td>SubnetMask</td>
<td>Optional: The subnet mask for the VRA network when <code>VraIPConfigurationTypeApi</code> is set to <code>Static</code>.</td>
</tr>
<tr>
<td>VraIPAddress</td>
<td>Optional: The IP address for the VRA when <code>VraIPConfigurationTypeApi</code> is set to <code>Static</code>.</td>
</tr>
<tr>
<td>VraIPConfigurationTypeApi</td>
<td>Optional: The IP configuration used for the VRA. Possible values are: <strong>Dhcp</strong>: The VRA IP address is allocated via a DHCP server. <strong>Static</strong>: The VRA is installed with a static IP that was assigned when the VRA was installed.</td>
</tr>
</tbody>
</table>

Note: If a value is not specified in the request, a null value is used. Nulls are specified in Json as `null`, all lowercase.

Json Response Format

The following is an example response Json body for `https://zvm_ip:port/v1/vras/{vraIdentifier}`.

"String content"
XML Response Format

For the XML response format, see “VRAs API PUT Method Request and Response Formats”, on page 275.

Response Values

The response value is the task identifier which can be used with the Tasks API to monitor the action.

**VRAs: DELETE**

Delete a specified VRA.

**URL**

Delete VRA  
https://zvm_ip:port/v1/vras/{vraIdentifier}

Where:

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>zvm_ip</strong></td>
<td>The IP address of the Zerto Virtual Manager where the API is run.</td>
</tr>
<tr>
<td><strong>port</strong></td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
<tr>
<td><strong>vraIdentifier</strong></td>
<td>The identifier of the VRA to be deleted.</td>
</tr>
</tbody>
</table>

**Request Format**

The request body is empty.

**Json Response Format**

The following is an example response Json body for https://zvm_ip:port/v1/vras/{vraIdentifier}.

"String content"

**XML Response Format**

For the XML response format, see “VRAs API DELETE Method Response Formats”, on page 275.

**Response Values**

The response value is the task identifier which can be used with the Tasks API to monitor the delete action.

**Code Examples**

For complete code examples, see “Code Samples”, on page 22.

/v1/vras cURL Code Example

Retrieve information about the VRAs in this site.

```
```

For more code examples, see “cURL Code”, on page 22.
ZORGs API

/v1/vzorgs returns information about Zerto organizations, ZORGs, defined in the Zerto Cloud Manager that is connected to the site where the API runs.

URL

All ZORGs  
https://zvm_ip:port/v1/zorgs

Single ZORG  
https://zvm_ip:port/v1/zorgs/{zorgIdentifier}

Where:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>zvm_ip</td>
<td>The IP address of the Zerto Virtual Manager where the API is run.</td>
</tr>
<tr>
<td>port</td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
<tr>
<td>zorgIdentifier</td>
<td>The identifier of the ZORG for which information is returned.</td>
</tr>
</tbody>
</table>

HTTP Method

GET

Security

The API is exposed over HTTPS. Client code must use the x-zerto-session HTTP authorization header.

See Also

Starting a session: Session: POST

Format

Json, XML

Request Format

The request body is empty.

Json Response Format

The following is an example response Json body for https://zvm_ip:port/v1/zorgs and for https://zvm_ip:port/v1/zorgs/{zorgIdentifier}.

```json
[{
    "Link": {
        "href": "String content",
        "identifier": "String content",
        "rel": "String content",
        "type": "String content"
    },
    "ZorgIdentifier": "String content",
    "ZorgName": "String content"
}]
```
Zerto Virtual Replication RESTful API Reference Guide - Version 5.5

Zerto Virtual Replication APIs

XML Response Format

For the XML response format, see “ZORG API XML Response Format”, on page 276.

Response Values


<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Link</td>
<td>The link details.</td>
</tr>
<tr>
<td>href</td>
<td>The URL used.</td>
</tr>
<tr>
<td>identifier</td>
<td>The internal ZORG identifier.</td>
</tr>
<tr>
<td>rel</td>
<td>The next path level for the API relative to the current path.</td>
</tr>
<tr>
<td>type</td>
<td>The API interface service.</td>
</tr>
<tr>
<td>ZorgIdentifier</td>
<td>The identifier of the ZORG, Zerto organization, defined in the Zerto Cloud Manager.</td>
</tr>
<tr>
<td>ZorgName</td>
<td>The name of the organization set up in Zerto Cloud Manager that uses a cloud service provider for recovery.</td>
</tr>
</tbody>
</table>

Code Examples

For complete code examples, see “Code Samples”, on page 22.

/v1/zorgs cURL Code Example

Retrieve information about the ZORGs defined in the Zerto Cloud Manager that manages this site.


For more code examples, see “cURL Code”, on page 22.

Resources Report API

/zvmservice/resourcesreport generates resource information about the virtual machines being protected to a recovery site. This information is collected at fixed times as defined in the Resource Reports section of the Advanced Settings dialog. Information for the report is saved for 90 days when the sampling period is hourly and for one year when the sampling period is daily. The report is run on the Zerto Virtual Manager at the recovery site.

URL

Basic filtering  https://zvm_ip:port/ZvmService/ResourcesReport/getSamples?fromTimeString={fromTimeString}&toTimeString={toTimeString}&startIndex={startIndex}&count={count}

Advanced filtering https://zvm_ip:port/ZvmService/ResourcesReport/getSamplesWithFilter?fromTimeString={fromTimeString}&toTimeString={toTimeString}&startIndex={startIndex}&count={count}&filter={filter}

Where:

<table>
<thead>
<tr>
<th>zvm_ip</th>
<th>The IP address of the Zerto Virtual Manager where the API is run.</th>
</tr>
</thead>
<tbody>
<tr>
<td>port</td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
</tbody>
</table>
**Filters**

Any combination of filters is valid. Filters are not case-sensitive.

**Optional Filters**

The following filter is optional when using both the `getSamples` and `getSamplesWithFilter` options:

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>startIndex</td>
<td>The first record within the time frame to write to the report. The value is zero-based and the default starting point is the first record, <code>startIndex=0</code>.</td>
</tr>
</tbody>
</table>

**Mandatory Filters**

The following filters are mandatory when using both the `getSamples` and `getSamplesWithFilter` options:

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>fromTimeString</td>
<td>The starting time of the report, supplied as a date and time with the format of the Zerto Virtual Manager where the API runs, with the format <code>yyyy-mm-dd hh:mm:ss</code>. If the time portion is not specified, the starting time of the report is 00:00:00.</td>
</tr>
<tr>
<td>toTimeString</td>
<td>The end time of the report, supplied as a date and time with the format of the Zerto Virtual Manager where the API runs, with the format <code>yyyy-mm-dd hh:mm:ss</code>. If the time portion is not specified, the starting time for the report is 00:00:00.</td>
</tr>
<tr>
<td>count</td>
<td>The number of records to write to the report. The maximum value is 500 and values over 500 default to 500. If a value is not specified, no records are written in the report. To return more than 500 records, see &quot;Perl Code Example: Retrieving All the Records&quot;, on page 194.</td>
</tr>
</tbody>
</table>

The filter parameter is mandatory when using the `getSamplesWithFilter` option:

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
</table>
| filter | Enables advanced filtering of the results according to specific criteria. For example:  
  
  `filter=('SourceSite='Las Vegas')`  
  
  The filter can include `=, >, >=, <, <=, <>, the “or” and the “and” logical operators. For example:  
  
  `(SourceSite='New York' or SourceSite='London') and (ThroughputInBytes>5.6 or ThroughputInBytes<2.6 or ThroughputInBytes=3) and (NumberOfvCpu>=2)`  
  
  Passing an empty filter is equivalent to not specifying any advanced filtering. |

<table>
<thead>
<tr>
<th>Class</th>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ActiveGuestMemoryInMB</td>
<td></td>
<td>The active memory of the virtual machine.</td>
</tr>
<tr>
<td>BandwidthInBytes</td>
<td></td>
<td>The average bandwidth used between two consecutive samples.</td>
</tr>
<tr>
<td>ConsumedHostMemoryInMB</td>
<td></td>
<td>The amount of host memory consumed by the virtual machine.</td>
</tr>
<tr>
<td>CpuLimitInMhz</td>
<td></td>
<td>The maximum MHz available for the CPUs in the virtual machine.</td>
</tr>
<tr>
<td>CpuReservedInMhz</td>
<td></td>
<td>The MHz reserved for use by the CPUs in the virtual machine.</td>
</tr>
<tr>
<td>CpuUsedInMhz</td>
<td></td>
<td>The MHz used by the CPUs in the virtual machine.</td>
</tr>
<tr>
<td>CrmId</td>
<td></td>
<td>The CRM identifier specified in Zerto Cloud Manager for an organization that uses a cloud service provider for recovery.</td>
</tr>
<tr>
<td>MemoryInMB</td>
<td></td>
<td>The virtual machine defined memory.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>--------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>MemoryLimitInMB</td>
<td>The upper limit for this virtual machine's memory allocation.</td>
<td></td>
</tr>
<tr>
<td>MemoryReservedInMB</td>
<td>The guaranteed memory allocation for this virtual machine.</td>
<td></td>
</tr>
<tr>
<td>NumberOfvCpu</td>
<td>The number of CPUs for the virtual machine.</td>
<td></td>
</tr>
<tr>
<td>NumberOfVolumes</td>
<td>The number of volumes attached to the virtual machine.</td>
<td></td>
</tr>
<tr>
<td>RecoveryJournalProvisionedStorageInGB</td>
<td>The amount of provisioned journal storage for the virtual machine.</td>
<td></td>
</tr>
<tr>
<td>RecoveryJournalUsedStorageInGB</td>
<td>The amount of used journal storage for the virtual machine.</td>
<td></td>
</tr>
<tr>
<td>RecoveryVolumesProvisionedStorageInGB</td>
<td>The amount of provisioned storage for the virtual machine in the target site. This value is the sum of volumes' provisioned size.</td>
<td></td>
</tr>
<tr>
<td>RecoveryVolumesUsedStorageInGB</td>
<td>The amount of used storage for the virtual machine in the target site.</td>
<td></td>
</tr>
<tr>
<td>ServiceProfile</td>
<td>The service profile used by the VPG.</td>
<td></td>
</tr>
<tr>
<td>SourceCluster</td>
<td>The source cluster name hosting the virtual machine.</td>
<td></td>
</tr>
<tr>
<td>SourceHost</td>
<td>The source ESX/ESXi name hosting the virtual machine.</td>
<td></td>
</tr>
<tr>
<td>SourceOrgVDC</td>
<td>The name of the source vDC Org.</td>
<td></td>
</tr>
<tr>
<td>SourceResourcePool</td>
<td>The source resource pool name hosting the virtual machine.</td>
<td></td>
</tr>
<tr>
<td>SourceSite</td>
<td>The source protected site name, defined in the Zerto User Interface.</td>
<td></td>
</tr>
<tr>
<td>SourceVCDOrg</td>
<td>The name of the source vCD Org.</td>
<td></td>
</tr>
<tr>
<td>SourceVolumesProvisionedStorageInGB</td>
<td>The amount of provisioned storage for the virtual machine in the source site. This value is the sum of volumes' provisioned size.</td>
<td></td>
</tr>
<tr>
<td>SourceVolumesUsedStorageInGB</td>
<td>The amount of used storage for the virtual machine in the source site.      This value is the sum of the volumes' used size.</td>
<td></td>
</tr>
<tr>
<td>SourceVraName</td>
<td>The name of the source VRA used to send data to the recovery site.</td>
<td></td>
</tr>
<tr>
<td>StorageProfile</td>
<td>The target vCD storage profile used.</td>
<td></td>
</tr>
<tr>
<td>TargetCluster</td>
<td>The target cluster name hosting the virtual machine.</td>
<td></td>
</tr>
<tr>
<td>TargetDatastores</td>
<td>The target datastore used for the virtual machine if it is recovered.</td>
<td></td>
</tr>
<tr>
<td>TargetHost</td>
<td>The target ESX/ESXi name hosting the virtual machine when it is recovered.</td>
<td></td>
</tr>
<tr>
<td>TargetOrgVDC</td>
<td>The name of the target vDC Org</td>
<td></td>
</tr>
<tr>
<td>TargetResourcePool</td>
<td>The target resource pool name where the virtual machine will be recovered.</td>
<td></td>
</tr>
<tr>
<td>TargetSite</td>
<td>The target site name, defined in the Zerto User Interface.</td>
<td></td>
</tr>
<tr>
<td>TargetVCDOrg</td>
<td>The name of the target vCD Org</td>
<td></td>
</tr>
<tr>
<td>TargetVraName</td>
<td>The name of the recovery VRA managing the recovery.</td>
<td></td>
</tr>
<tr>
<td>ThroughputInBytes</td>
<td>The average throughput used between two consecutive samples.</td>
<td></td>
</tr>
<tr>
<td>Timestamp</td>
<td>The date and time the resource information was collected.</td>
<td></td>
</tr>
<tr>
<td>VmHardwareVersion</td>
<td>The VMware hardware version.</td>
<td></td>
</tr>
</tbody>
</table>
### HTTP Method

GET

### Security

The API is exposed over HTTPS and does not require basic authentication.

### See Also

Perl Code Example: Retrieving the First 100 Records
Perl Code Example: Retrieving All the Records

### Response Format

Example XML response body:

```xml
<ArrayOfVmResourcesInfoStorageObject
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <VmResourcesInfoStorageObject>
    <ActiveGuestMemoryInMB>20</ActiveGuestMemoryInMB>
    <BandwidthInBytes>-1</BandwidthInBytes>
    <ConsumedHostMemoryInMB>153</ConsumedHostMemoryInMB>
    <CpuLimitInMhz>-1</CpuLimitInMhz>
    <CpuReservedInMhz>0</CpuReservedInMhz>
    <CpuUsedInMhz>89</CpuUsedInMhz>
    <CrmId/>
    <MemoryInMB>2048</MemoryInMB>
    <MemoryLimitInMB>-1</MemoryLimitInMB>
    <MemoryReservedInMB>0</MemoryReservedInMB>
    <NumberOfVolumes>1</NumberOfVolumes>
    <NumberOfvCpu>1</NumberOfvCpu>
    <RecoveryJournalProvisionedStorageInGB>16</RecoveryJournalProvisionedStorageInGB>
    <RecoveryJournalUsedStorageInGB>0.584</RecoveryJournalUsedStorageInGB>
    <RecoveryVolumesProvisionedStorageInGB>5</RecoveryVolumesProvisionedStorageInGB>
    <RecoveryVolumesUsedStorageInGB>1.106</RecoveryVolumesUsedStorageInGB>
  </VmResourcesInfoStorageObject>
</ArrayOfVmResourcesInfoStorageObject>
```
<ServiceProfile>System Service Profile</ServiceProfile>
<SourceCluster/>
<SourceHost>987.65.432.1</SourceHost>
<SourceOrgVDC/>
<SourceResourcePool/>
<SourceSite>Dallas</SourceSite>
<SourceVCDOrg/>
<SourceVolumesProvisionedStorageInGB>5</SourceVolumesProvisionedStorageInGB>
<SourceVolumesUsedStorageInGB>1.106</SourceVolumesUsedStorageInGB>
<SourceVraName>2-VRA-987.65.432.1</SourceVraName>
<StorageProfile/>

<TargetCluster>London Cluster</TargetCluster>
<TargetDatastores>datastore2</TargetDatastores>
<TargetHost>123.45.678.9</TargetHost>
<TargetOrgVDC/>
<TargetResourcePool/>
<TargetSite>London</TargetSite>
<TargetVCDOrg/>
<TargetVraName>2-VRA-123.45.678.9</TargetVraName>
<ThroughputInBytes>-1</ThroughputInBytes>
<Timestamp>2014-02-04T01:00:23</Timestamp>

<VmHardwareVersion>vmx-07</VmHardwareVersion>
    <a:InternalVmName>vm-518</a:InternalVmName>
    <a:ServerIdentifier>
        <a:ServerGuid>cbaab5ca-3915-4df7-8223-1ac851021f5a</a:ServerGuid>
    </a:ServerIdentifier>
</VmIdentifier>

<VmName>Billing</VmName>
<VpgName>New VPG</VpgName>
<VpgType>VC2VC</VpgType>
<Zorg/>

Response Values

Response values for https://zvm_ip:port/ZvmService/ResourcesReport/{filters}

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ActiveGuestMemoryInMB</td>
<td>The active memory of the virtual machine.</td>
</tr>
<tr>
<td>BandwidthInBytes</td>
<td>The average bandwidth used between two consecutive samples.</td>
</tr>
<tr>
<td>ConsumedHostMemoryInMB</td>
<td>The amount of host memory consumed by the virtual machine.</td>
</tr>
<tr>
<td>CpuLimitInMhz</td>
<td>The maximum MHz available for the CPUs in the virtual machine.</td>
</tr>
<tr>
<td>CpuReservedInMhz</td>
<td>The MHz reserved for use by the CPUs in the virtual machine.</td>
</tr>
<tr>
<td>CpuUsedInMhz</td>
<td>The MHz used by the CPUs in the virtual machine.</td>
</tr>
<tr>
<td>CrmId</td>
<td>The CRM identifier specified in Zerto Cloud Manager for an organization that uses a cloud service provider for recovery.</td>
</tr>
<tr>
<td>MemoryInMB</td>
<td>The virtual machine defined memory.</td>
</tr>
<tr>
<td>MemoryLimitInMB</td>
<td>The upper limit for this virtual machine's memory allocation.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>MemoryReservedInMB</td>
<td>The guaranteed memory allocation for this virtual machine.</td>
</tr>
<tr>
<td>NumberOfvCpus</td>
<td>The number of CPUs for the virtual machine.</td>
</tr>
<tr>
<td>NumberOfVolumes</td>
<td>The number of volumes attached to the virtual machine.</td>
</tr>
<tr>
<td>RecoveryJournalProvisionedStorageInGB</td>
<td>The amount of provisioned journal storage for the virtual machine.</td>
</tr>
<tr>
<td>RecoveryJournalUsedStorageInGB</td>
<td>The amount of used journal storage for the virtual machine.</td>
</tr>
<tr>
<td>RecoveryVolumesProvisionedStorageInGB</td>
<td>The amount of provisioned storage for the virtual machine in the target site. This value is the sum of volumes’ provisioned size.</td>
</tr>
<tr>
<td>RecoveryVolumesUsedStorageInGB</td>
<td>The amount of used storage for the virtual machine in the target site.</td>
</tr>
<tr>
<td>ServiceProfile</td>
<td>The service profile used by the VPG.</td>
</tr>
<tr>
<td>SourceCluster</td>
<td>The source cluster name hosting the virtual machine.</td>
</tr>
<tr>
<td>SourceHost</td>
<td>The source ESX/ESXi name hosting the virtual machine.</td>
</tr>
<tr>
<td>SourceOrgVDC</td>
<td>The name of the source vDC Org.</td>
</tr>
<tr>
<td>SourceResourcePool</td>
<td>The source resource pool name hosting the virtual machine.</td>
</tr>
<tr>
<td>SourceSite</td>
<td>The source protected site name, defined in the Zerto User Interface.</td>
</tr>
<tr>
<td>SourceVCDOrg</td>
<td>The name of the source vCD Org.</td>
</tr>
<tr>
<td>SourceVolumesProvisionedStorageInGB</td>
<td>The amount of provisioned storage for the virtual machine in the source site. This value is the sum of volumes’ provisioned size.</td>
</tr>
<tr>
<td>SourceVolumesUsedStorageInGB</td>
<td>The amount of used storage for the virtual machine in the source site.</td>
</tr>
<tr>
<td>SourceVraName</td>
<td>The name of the source VRA used to send data to the recovery site.</td>
</tr>
<tr>
<td>StorageProfile</td>
<td>The target vCD storage profile used.</td>
</tr>
<tr>
<td>TargetCluster</td>
<td>The target cluster name hosting the virtual machine.</td>
</tr>
<tr>
<td>TargetDatastores</td>
<td>The target datastore used for the virtual machine if it is recovered.</td>
</tr>
<tr>
<td>TargetHost</td>
<td>The target ESX/ESXi name hosting the virtual machine when it is recovered.</td>
</tr>
<tr>
<td>TargetOrgVDC</td>
<td>The name of the target vDC Org</td>
</tr>
<tr>
<td>TargetResourcePool</td>
<td>The target resource pool name where the virtual machine will be recovered.</td>
</tr>
<tr>
<td>TargetSite</td>
<td>The target site name, defined in the Zerto User Interface.</td>
</tr>
<tr>
<td>TargetVCDOrg</td>
<td>The name of the target vCD Org</td>
</tr>
<tr>
<td>TargetVraName</td>
<td>The name of the recovery VRA managing the recovery.</td>
</tr>
<tr>
<td>ThroughputInBytes</td>
<td>The average throughput used between two consecutive samples.</td>
</tr>
<tr>
<td>Timestamp</td>
<td>The date and time the resource information was collected. The value can be converted to an understandable date using code similar to the following: var date = new Date(jsonDate); or code similar to the Perl code example, jsonDateToString($), in “Perl Code Example: Retrieving the First 100 Records”, on page 193</td>
</tr>
<tr>
<td>VmHardwareVersion</td>
<td>The VMware hardware version.</td>
</tr>
<tr>
<td>VmIdentifier</td>
<td>The internal virtual machine identifier.</td>
</tr>
<tr>
<td>VmName</td>
<td>The name of the virtual machine.</td>
</tr>
<tr>
<td>VpgName</td>
<td>The name of the VPG.</td>
</tr>
</tbody>
</table>
Perl Code Example: Retrieving the First 100 Records

The following example uses Perl and Json1 and can be run on both Windows and Linux operating systems.

```perl
#!/usr/bin/perl -w
use strict;
use LWP::UserAgent;
use JSON;

main();
sub main {
    print "Retrieving $url\n";
    my $userAgent = LWP::UserAgent->new;
    my $req = HTTP::Request->new(GET => $url);
    $req->header(Accept => 'application/json');
    my $res = $userAgent->request($req);
    if ($res->is_success) {
        my $json = new JSON;
        my @json_array = @{$json->decode($res->content)};
        my $last_timestamp = "";
        for my $elem (@json_array) {
            my $timestamp = jsonDateToString($elem->{Timestamp});
            if ($timestamp ne $last_timestamp) {
                print \nTime: $timestamp\n;
                $last_timestamp = $timestamp;
            }
            print "VPG: $elem->{VpgName}, VM: $elem->{VmName}, Bandwidth (Bps): $elem->{BandwidthInBytes}\n";
        }
    } else {
        print $res->status_line, "\n";
    }
}

sub jsonDateToString($) {
    my $json_date = shift;
    if ($json_date =~ m{  (\d+) \b ([+-]\d\d\d\d)? }x ) {
        my ( $epoch_milliseconds, $time_zone ) = ( $1, $2 );
        return localtime($epoch_milliseconds / 1000);
    }
    return $json_date;
}
```

### PARAMETER | DESCRIPTION

| VpgType | The VPG Type. Possible values are: VC2VC: vCenter to vCenter replication. VC2VCD: vCenter to vCloud Director replication. VCD2VCD: vCloud Director to vCloud Director replication. VCD2VC: vCloud Director to vCenter replication. |
| ZORG | An organization set up in the Zerto Cloud Manager using a cloud service provider for recovery. |
The following output is based on each VPG containing only one virtual machine.

Retrieving

Time: Thu Jun 20 01:00:00 2013
VPG: HR Dallas, VM: HR Dallas, Bandwidth (Bps): 639.437
VPG: HR NY, VM: HR New York, Bandwidth (Bps): 206.335
VPG: HR London, VM: HR London, Bandwidth (Bps): 1261.204
VPG: Reporting, VM: NY Reporting, Bandwidth (Bps): 2998.503
VPG: BO Recs, VM: Reconciliation, Bandwidth (Bps): 7.988
VPG: FX Report, VM: FX, Bandwidth (Bps): 1397.885

Time: Fri Jun 21 01:00:00 2013
VPG: HR Dallas, VM: HR Dallas, Bandwidth (Bps): 833.524
VPG: HR NY, VM: HR New York, Bandwidth (Bps): 183.923
VPG: HR London, VM: HR London, Bandwidth (Bps): 1282.578
VPG: Reporting, VM: NY Reporting, Bandwidth (Bps): 4429.341
VPG: BO Recs, VM: Reconciliation, Bandwidth (Bps): 7.994
VPG: DC Zertolab.local, VM: DC Zertolab.local, Bandwidth (Bps): 872.078

Time: Sat Jun 22 01:00:00 2013
VPG: HR Dallas, VM: HR Dallas, Bandwidth (Bps): 870.926
VPG: HR NY, VM: HR New York, Bandwidth (Bps): 218.696
VPG: HR London, VM: HR London, Bandwidth (Bps): 1245.913
VPG: Reporting, VM: NY Reporting, Bandwidth (Bps): 3022.375
VPG: BO Recs, VM: Reconciliation, Bandwidth (Bps): 7.994
VPG: FX Report, VM: FX, Bandwidth (Bps): 942.535

Code Notes

The example assumes Json and Crypt::SSLeay Perl packages are installed.¹

The API returns the first 100 records.

The code uses the default SSL_VERIFY_NONE verification mode for clients. The most recent Perl versions will still run this code although they will report the verification as an error.

jsonDateToString($) converts the date from its epoch format to a readable date.

Perl Code Example: Retrieving All the Records

The following example uses Perl and Json² and can be run on both Windows and Linux operating systems.

```perl
#!/usr/bin/perl -w
use strict;
use LWP::UserAgent;
use JSON;
```

---

¹ If you prefer to use XML, you can convert the Json to XML.
² On a Windows operating system, ActiveState Perl was used.
main();
sub main {
    my $zvm = "1.2.3.4";
    my $startIndex = 0;
    my $count = 500;
    my $url;
    my $last_timestamp = "";

    my $json = new JSON;
    my $userAgent = LWP::UserAgent->new;
    my $num_results_from_last_query = 0;

    do {
2015-01-01&startIndex=$startIndex&count=$count";
        print "Retrieving $url\n";
        my $req = HTTP::Request->new(GET => $url);
        $req->header(Accept => 'application/json');
        my $res = $userAgent->request($req);

        if ($res->is_success) {
            my @json_array = @{$json->decode($res->content)};
            $num_results_from_last_query = @json_array;
            for my $elem (@json_array) {
                my $timestamp = jsonDateToString($elem->{Timestamp});
                if ($timestamp ne $last_timestamp) {
                    print "\nTime: $timestamp\n";
                    $last_timestamp = $timestamp;
                }
                print "VPG: $elem->{VpgName}, VM: $elem->{VmName}, Bandwidth (Bps): $elem->{BandwidthInBytes}\n";
            }
            $startIndex += $count;
        } else {
            print "Error: ", $res->status_line, "\n";
            last;
        }
    } while ($num_results_from_last_query == $count);
}

sub jsonDateToString($) {
    my $json_date = shift;
    if ($json_date =~ m{ \b (\d+) \b \([+-]\d\d\d\d\)? \}x ) {
        my ($epoch_milliseconds, $time_zone) = ( $1, $2 );
        return localtime($epoch_milliseconds / 1000);
    } else {
        return $json_date;
    }
}

1. The example assumes Json and Crypt::SSLeay Perl packages are installed.

2. If you prefer to use XML, you can convert the Json to XML.
Managing vCD APIs

Following are APIs that can be run in a vCD environment:

**vCD Virtualization Sites APIs, on page 196**

**VPG Management APIs, on page 200**

**Note:** The protected machines are protected as a vCD vApp in the recovery site vCD. To review which settings are retained, see *Settings Retained when Replicating from a Protected Site vCloud Director to a Recovery Site vCloud Director.*

### vCD Virtualization Sites APIs

In this section you can get information about Org vDCs and unprotected vCD vApps in a site.

**Note:** The API is exposed over HTTPS. Client code must use the `x-zerto-session` HTTP authorization header.

The following APIs are available:

- **List Org vDCs in a Site**
- **List Networks of an Org vDC in a Site**
- **List Storage Profiles of an Org vDC in a Site**
- **List Unprotected vCD vApps in a Site**

#### List Org vDCs in a Site

Use this API to get a list of Org vDCs on a specific site.

**Request**

**Response**

**Request**

<table>
<thead>
<tr>
<th>METHOD</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/orgvdcso</td>
</tr>
</tbody>
</table>

Where:

- **zvm_ip** The IP address of the Zerto Virtual Manager where the API is run.
- **port** The port to access the Zerto Virtual Manager. The default port is 9669.
- **siteIdentifier** The identifier of the site on which the Org vDC resides.

The request Json body is empty.

**Response**

The following is an example response Json body.
List Networks of an Org vDC in a Site

Use the following API to get a list of networks existing in an Org vDC on a specific site.

**Request**

**Response**

**Request**

<table>
<thead>
<tr>
<th>METHOD</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>[https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/orgvdc/{orgvdcidentifier}/networks](https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/orgvdc/{orgvdcidentifier}/networks)</td>
</tr>
</tbody>
</table>

Where:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>zvm_ip</td>
<td>The IP address of the Zerto Virtual Manager where the API is run.</td>
</tr>
<tr>
<td>port</td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
<tr>
<td>siteIdentifier</td>
<td>The identifier of the site on which the Org vDC resides.</td>
</tr>
<tr>
<td>orgvdcidentifier</td>
<td>The identifier of an Org vDC.</td>
</tr>
</tbody>
</table>

The request Json body is empty.

**Response**

The following is an example response Json body.
List Storage Profiles of an Org vDC in a Site

Use this API to get a list of available storage profiles in an Org vDC.

**Request**

**Response**

**Request**

**PARAMETER** | **DESCRIPTION**
--- | ---
NetworkIdentifier | The internal identification of the Org vDC network.
VirtualizationNetworkName | The orgVDC network name.

**METHOD** | **URL**
--- | ---
GET | https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/orgvdcs/{orgvdcidentifier}/storageprofiles

Where:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>zvm_ip</td>
<td>The IP address of the Zerto Virtual Manager where the API is run.</td>
</tr>
<tr>
<td>port</td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
<tr>
<td>siteIdentifier</td>
<td>The identifier of the site on which the Org vDC resides.</td>
</tr>
<tr>
<td>orgvdcidentifier</td>
<td>The identifier of an Org vDC.</td>
</tr>
</tbody>
</table>

The request Json body is empty.

**Response**

The following is an example response Json body.
List Unprotected vCD vApps in a Site

Use this API to get a list of vCD vApps that are not protected in a site.

**Request**

**METHOD**

<table>
<thead>
<tr>
<th>URL</th>
<th>GET</th>
</tr>
</thead>
<tbody>
<tr>
<td>https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/vcdvapps</td>
<td></td>
</tr>
</tbody>
</table>

**Where:**

- **zvm_ip**
  - The IP address of the Zerto Virtual Manager where the API is run.
- **port**
  - The port to access the Zerto Virtual Manager. The default port is 9669.
- **siteIdentifier**
  - The identifier of the site on which the Org vDC resides.

The request Json body is empty.

**Response**

The following is an example response Json body.

```json
[{
   "VCDName": "String content",
   "VCDVappIdentifier": "String content"
},
{
   "VCDName": "String content",
   "VCDVappIdentifier": "String content"
}]
```
VPG Management APIs

Using the VpgSettings APIs, you can view existing vCD VPGs, create, update and delete existing vCD VPGs.

There are two ways of creating new vCD VPGs:

- “Multi-Step Operation for Creating an Empty VPG Template”, on page 200
- “Two Step Operation”, on page 221

Following is an explanation for both methods.

Multi-Step Operation for Creating an Empty VPG Template

Using this method, you run the following APIs:

1. Get a VPG Settings Identifier
2. Get an Empty VPG Template
3. Add Values in the Empty VPG Template
4. Get a Full VPG Template

Get a VPG Settings Identifier

Run this API to get a VPG settings identifier. With the VPG Settings Identifier, you can request for an empty VPG template.

Request

Response

<table>
<thead>
<tr>
<th>METHOD</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>https://zvm_ip:port/v1/vpgSettings</td>
</tr>
</tbody>
</table>

Where:

- **zvm_ip** The IP address of the Zerto Virtual Manager where the API is run.
- **port** The port to access the Zerto Virtual Manager. The default port is 9669.

Request

The request Json body is empty.

Response

The following is an example response.

```json
{
    "VpgSettingsIdentifier": "String content"
}
```
Get an Empty VPG Template

Run this API to get an empty VPG template. The empty template includes mandatory fields for creating a VPG.

### Request

The request body is empty.

### Response

The following is an example response.

```json
[
  {
    "Backup": "String Content",
    "Basic": {
      "JournalHistoryInHours": number,
      "Name": "String Content",
      "Priority": "String Content",
      "ProtectedSiteIdentifier": "String Content",
      "RecoverySiteIdentifier": "String Content",
      "RpoInSeconds": Number,
      "ServiceProfileIdentifier": "String Content",
      "TestIntervalInMinutes": Number,
      "UseWanCompression": Boolean,
      "ZorgIdentifier": "String Content"
    },
    "BootGroups": {
      "BootGroups": [
        {
          "BootDelayInSeconds": Number,
          "BootGroupIdentifier": "String Content",
          "Name": "String Content"
        }
      ]
    }
  }
]
```
"Journal": {
  "DatastoreIdentifier": "String Content",
  "Limitation": {
    "HardLimitInMB": Number,
    "HardLimitInPercent": Number,
    "WarningThresholdInMB": Number,
    "WarningThresholdInPercent": Number
  }
},

"Networks": {
  "Failover": {
    "Hypervisor": {
      "DefaultNetworkIdentifier": "String Content"
    },
    "VCD": "String Content"
  },
  "FailoverTest": {
    "Hypervisor": {
      "DefaultNetworkIdentifier": "String Content"
    },
    "VCD": "String Content"
  }
},

"Protected": {
  "VCD": {
    "VCDVappIdentifier": "String Content"
  }
},

"Recovery": {
  "DefaultDatastoreClusterIdentifier": "String Content",
  "DefaultDatastoreIdentifier": "String Content",
  "DefaultFolderIdentifier": "String Content",
  "DefaultHostClusterIdentifier": "String Content",
  "DefaultHostIdentifier": "String Content",
  "ResourcePoolIdentifier": "String Content",
  "VCD": {
    "OrgVcdIdentifier": "String Content"
  }
},

"Scripting": {
  "PostBackup": "String Content",
  "PostRecovery": {
    "Command": "String Content",
    "Parameters": null,
    "TimeoutInSeconds": Number
  },
  "PreRecovery": {
    "Command": "String Content",
    "Parameters": "String Content",
    "TimeoutInSeconds": Number
  }
},

"Vms": [],
"VpgIdentifier": "String Content",
"VpgSettingsIdentifier": "String Content"}]}
<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Backup</td>
<td>Information related to offsite backup.</td>
</tr>
<tr>
<td>Basic</td>
<td>Basic VPG settings.</td>
</tr>
<tr>
<td>JournalHistoryInHours</td>
<td>The time that all write commands are saved in the journal. The value is between 1 and 336 (14 days).</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the VPG.</td>
</tr>
<tr>
<td>Priority</td>
<td>The priority specified for the VPG. Possible values are:</td>
</tr>
<tr>
<td></td>
<td><strong>Low</strong>: The VPG has a low priority for transferring data.</td>
</tr>
<tr>
<td></td>
<td><strong>Medium</strong>: The VPG has a medium priority for transferring data.</td>
</tr>
<tr>
<td></td>
<td><strong>High</strong>: The VPG has a high priority for transferring data.</td>
</tr>
<tr>
<td>ProtectedSiteIdentifier</td>
<td>The identifier of the source site where the VPG virtual machines will be protected. This is the site where the API runs.</td>
</tr>
<tr>
<td>RecoverySiteIdentifier</td>
<td>The identifier of the target site where the VPG virtual machines will be recovered.</td>
</tr>
<tr>
<td>RpoInSeconds</td>
<td>The maximum desired time between each automatic checkpoint being written to the journal before an alert is issued.</td>
</tr>
<tr>
<td>ServiceProfileIdentifier</td>
<td>The identifier of the service profile to use for the VPG when a Zerto Cloud Manager is used.</td>
</tr>
<tr>
<td>TestIntervalInMinutes</td>
<td>The time, in minutes, recommended between testing the integrity of the VPG. A warning is issued if a test is not done within this time frame. Possible values are:</td>
</tr>
<tr>
<td></td>
<td><strong>0</strong>: No testing is expected.</td>
</tr>
<tr>
<td></td>
<td><strong>43200</strong> or <strong>null</strong>: Testing is expected monthly.</td>
</tr>
<tr>
<td></td>
<td><strong>131040</strong>: Testing is expected every three months.</td>
</tr>
<tr>
<td></td>
<td><strong>262080</strong>: Testing is expected every six months.</td>
</tr>
<tr>
<td></td>
<td><strong>394560</strong>: Testing is expected every nine months.</td>
</tr>
<tr>
<td></td>
<td><strong>525600</strong>: Testing is expected every twelve months.</td>
</tr>
<tr>
<td>UseWanCompression</td>
<td>True: Data will be compressed before sending it to the recovery site.</td>
</tr>
<tr>
<td></td>
<td>False: Data will not be compressed before sending it to the recovery site.</td>
</tr>
<tr>
<td>ZorgIdentifier</td>
<td>The identifier of the ZORG, Zerto organization, defined in the Zerto Cloud Manager.</td>
</tr>
<tr>
<td>BootGroups</td>
<td>Information about boot groups.</td>
</tr>
<tr>
<td>BootGroups</td>
<td>-</td>
</tr>
<tr>
<td>BootDelayInSeconds</td>
<td>Specifies the delay, in seconds, between starting up the virtual machines in this group and starting up the virtual machines in the next group.</td>
</tr>
<tr>
<td>BootGroupIdIdentifier</td>
<td>The identifier of a boot group.</td>
</tr>
<tr>
<td>Name</td>
<td>The name of a boot group.</td>
</tr>
<tr>
<td>Journal</td>
<td>Information about the journal.</td>
</tr>
<tr>
<td>DatastoreIdentifier</td>
<td>The identifier of the storage used by the journal. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>Limitation</td>
<td>Information about the journal limitations.</td>
</tr>
<tr>
<td>HardLimitInMB</td>
<td>The maximum journal size in MBs. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>HardLimitInPercent</td>
<td>The percentage of the virtual machine volume size the journal can grow to. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>WarningThresholdInMB</td>
<td>The journal size, in MBs, that generates a warning that the journal is nearing its hard limit. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>WarningThresholdInPercent</td>
<td>The percentage of the virtual machine volume size that generates a warning. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>Networks</td>
<td>Information about the networks that connect the protected and recovery sites.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Failover</td>
<td>Information about the networks used for failover.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td>Information related to networks used in hypervisors.</td>
</tr>
<tr>
<td>DefaultNetworkIdentifier</td>
<td>The network identifier of the network to use during a failover or move operation in which the recovered virtual machines will run.</td>
</tr>
<tr>
<td>VCD</td>
<td>Information related to networks used in vCDs.</td>
</tr>
<tr>
<td>FailoverTest</td>
<td>Information about the networks used for testing failover.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td>Information related to networks used in hypervisors.</td>
</tr>
<tr>
<td>DefaultNetworkIdentifier</td>
<td>The network identifier of the network to use when testing the failover of virtual machines in the recovery site.</td>
</tr>
<tr>
<td>VCD</td>
<td>Information related to networks used in a vCD environment.</td>
</tr>
<tr>
<td>Protected</td>
<td>Information about the protected site.</td>
</tr>
<tr>
<td>VCD</td>
<td>Null</td>
</tr>
<tr>
<td>Recovery</td>
<td>Information about the recovery.</td>
</tr>
<tr>
<td>DefaultDatastoreClusterIdentifier</td>
<td>The identifier of the default datastore cluster used in the recovery site.</td>
</tr>
<tr>
<td></td>
<td>Note: Only when the recovery site is a vSphere site.</td>
</tr>
<tr>
<td>DefaultDatastoreIdentifier</td>
<td>The identifier of the default storage where the metadata files for the virtual machines are stored, such as the vmx or vhdx files. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultFolderIdentifier</td>
<td>The identifier of the default folder used for recovery. The identifier comprises the server identifier and the folder moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultHostClusterIdentifier</td>
<td>The identifier of the default host cluster that handles the replicated data. The identifier comprises the server identifier and the host cluster moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultHostIdentifier</td>
<td>The identifier of the default host that handles the replicated data. The identifier comprises the server identifier and the host moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>ResourcePoolIdentifier</td>
<td>The identifier of the resource pool for the recovered virtual machines. The identifier comprises the server identifier and the resource pool moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>VCD</td>
<td>Null. Information about the vCD recovery vApp.</td>
</tr>
<tr>
<td>Scripting</td>
<td>Information about the scripts to run, either before or after recovery operation, or after an offsite backup is run.</td>
</tr>
<tr>
<td>PostBackup</td>
<td>Information about scripts that are run after an offsite backup is performed.</td>
</tr>
<tr>
<td>PostRecovery</td>
<td>Information about scripts that are run after a recovery operation is performed.</td>
</tr>
<tr>
<td>Command</td>
<td>The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site.</td>
</tr>
<tr>
<td>Parameters</td>
<td>Parameters to pass to the script.</td>
</tr>
<tr>
<td>TimeoutInSeconds</td>
<td>The time-out, in seconds, for the script to run.</td>
</tr>
<tr>
<td>PreRecovery</td>
<td>Information about scripts that are run before a recovery operation is performed.</td>
</tr>
<tr>
<td>Command</td>
<td>The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site.</td>
</tr>
<tr>
<td>Parameters</td>
<td>Parameters to pass to the script.</td>
</tr>
<tr>
<td>TimeoutInSeconds</td>
<td>The time-out, in seconds, for the script to run.</td>
</tr>
</tbody>
</table>
Add Values in the Empty VPG Template

Once you get the empty template, add values for the new VPG.

Some parameters appear as stand alone with the value null, even though they have sub-parameters. See for example the parameter OrgVcdIdentifier:

```json
"Recovery": {
  "DefaultDatastoreIdentifier": null,
  "DefaultFolderIdentifier": null,
  "DefaultHostClusterIdentifier": null,
  "DefaultHostIdentifier": null,
  "ResourcePoolIdentifier": null
  "VCD": {
    "OrgVcdIdentifier": null
  }
}
```

By running this API with a value in OrgVcdIdentifier, you are defining the recovery site as a vCD site. After adding the values in the empty template, you run a GET API to receive the full template, with the newly defined parameters. If you added a value to the OrgVcdIdentifier parameter, the Networks section expands from the original Networks section:

```json
"Networks": {
  "Failover": {
    "Hypervisor": {
      "DefaultNetworkIdentifier": null
    }
    "VCD": null
  },
  "FailoverTest": {
    "Hypervisor": {
      "DefaultNetworkIdentifier": null
    }
    "VCD": null
  }
}
```
"Networks": {
    "Failover": {
        "Hypervisor": {
            "DefaultNetworkIdentifier": null
        }
    },
    "VCD": {
        "CopyNatRules": "String content",
        "IsEnableGuestCustomization": "true",
        "DefaultRecoveryOrgVdcNetworkIdentifier": "String content",
        "Mapping": {
            "ProtectedOrgVdcNetworkIdentifier": "String content",
            "RecoveryOrgVdcNetworkIdentifier": "String content",
            "ReverseTestOrgVdcNetworkIdentifier": "String content"
        }
    },
    "FailoverTest": {
        "Hypervisor": {
            "DefaultNetworkIdentifier": null
        }
    },
    "VCD": {
        "CopyNatRules": "String content",
        "IsEnableGuestCustomization": "true",
        "DefaultRecoveryOrgVdcNetworkIdentifier": "String content",
        "Mapping": {
            "ProtectedOrgVdcNetworkIdentifier": "String content",
            "RecoveryOrgVdcNetworkIdentifier": "String content",
            "ReverseTestOrgVdcNetworkIdentifier": "String content"
        }
    }
}

Note: Not all parameters with a value of *null*, are expanded after they are filled.

<table>
<thead>
<tr>
<th>METHOD</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUT</td>
<td>https://zvm_ip:port/v1/vpgSettings/VpgSettingsIdentifier</td>
</tr>
</tbody>
</table>

Where:

- **zvm_ip** The IP address of the Zerto Virtual Manager where the API is run.
- **port** The port to access the Zerto Virtual Manager. The default port is 9669.
- **VpgSettingsIdentifier** The identifier of the session.
The following is an example vCD request Json body.

```json
{
  "Backup": "String Content",
  "Basic": {
    "JournalHistoryInHours": number,
    "Name": "String Content",
    "Priority": "String Content",
    "ProtectedSiteIdentifier": "String Content",
    "RecoverySiteIdentifier": "String Content",
    "RpoInSeconds": Number,
    "ServiceProfileIdentifier": "String Content",
    "TestIntervalInMinutes": Number,
    "UseWanCompression": Boolean,
    "ZorgIdentifier": "String Content"
  },
  "BootGroups": {
    "BootGroups": {
      "BootDelayInSeconds": Number,
      "BootGroupIdentifier": "String Content",
      "Name": "String Content"
    }
  },
  "Journal": {
    "DatastoreIdentifier": "String Content",
    "Limitation": {
      "HardLimitInMB": Number,
      "HardLimitInPercent": Number,
      "WarningThresholdInMB": Number,
      "WarningThresholdInPercent": Number
    }
  },
  "Networks": {
    "Failover": {
      "Hypervisor": {
        "DefaultNetworkIdentifier": "String Content"
      }
    },
    "FailoverTest": {
      "Hypervisor": {
        "DefaultNetworkIdentifier": "String Content"
      }
    }
  },
  "Protected": {
    "VCD": {
      "VCDVappIdentifier": "String Content"
    }
  }
}```
"Recovery": {
  "DefaultDatastoreClusterIdentifier": "String Content",
  "DefaultDatastoreIdentifier": "String Content",
  "DefaultFolderIdentifier": "String Content",
  "DefaultHostClusterIdentifier": "String Content",
  "DefaultHostIdentifier": "String Content",
  "ResourcePoolIdentifier": "String Content"
},
"VCD": {
  "OrgVcdIdentifier": "String Content"
},

"Scripting": {
  "PostBackup": "String Content",
  "PostRecovery": {
    "Command": "String Content",
    "Parameters": null,
    "TimeoutInSeconds": Number
  },
  "PreRecovery": {
    "Command": "String Content",
    "Parameters": "String Content",
    "TimeoutInSeconds": Number
  }
},

"Vms": [],
"VpgIdentifier": "String Content",
"VpgSettingsIdentifier": "String Content"
}

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Backup</td>
<td>Information related to offsite backup.</td>
</tr>
<tr>
<td>Basic</td>
<td>Basic VPG settings.</td>
</tr>
<tr>
<td>JournalHistoryInHours</td>
<td>The time that all write commands are saved in the journal. The value is between 1 and 336 (14 days).</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the VPG.</td>
</tr>
<tr>
<td>Priority</td>
<td>The priority specified for the VPG. Possible values are:</td>
</tr>
<tr>
<td></td>
<td>Low: The VPG has a low priority for transferring data.</td>
</tr>
<tr>
<td></td>
<td>Medium: The VPG has a medium priority for transferring data.</td>
</tr>
<tr>
<td></td>
<td>High: The VPG has a high priority for transferring data.</td>
</tr>
<tr>
<td>ProtectedSiteIdentifier</td>
<td>The identifier of the source site where the VPG virtual machines will be protected. This is the site where the API runs.</td>
</tr>
<tr>
<td>RecoverySiteIdentifier</td>
<td>The identifier of the target site where the VPG virtual machines will be recovered.</td>
</tr>
<tr>
<td>RpoinSeconds</td>
<td>The maximum desired time between each automatic checkpoint being written to the journal before an alert is issued.</td>
</tr>
<tr>
<td>ServiceProfileIdentifier</td>
<td>The identifier of the service profile to use for the VPG when a Zerto Cloud Manager is used.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| TestIntervalInMinutes     | The time, in minutes, recommended between testing the integrity of the VPG. A warning is issued if a test is not done within this time frame. Possible values are:  
0: No testing is expected.  
43200 or null: Testing is expected monthly.  
131040: Testing is expected every three months.  
262080: Testing is expected every six months.  
394560: Testing is expected every nine months.  
525600: Testing is expected every twelve months. |
| UseWanCompression         | True: Data will be compressed before sending it to the recovery site.  
False: Data will not be compressed before sending it to the recovery site. |
| ZorgIdentifier            | The identifier of the ZORG, Zerto organization, defined in the Zerto Cloud Manager. |
| BootGroups                | Information about boot groups.                                             |
| BootGroups                | -                                                                           |
| BootDelayInSeconds        | Specifies the delay, in seconds, between starting up the virtual machines in this group and starting up the virtual machines in the next group. |
| BootGroupIdentifier       | The identifier of a boot group.                                            |
| Name                      | The name of a boot group.                                                  |
| Journal                   | Information about the journal.                                            |
| DatastoreIdentifier       | The identifier of the storage used by the journal. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref. |
| Limitation                | Information about the journal limitations.                                 |
| HardLimitInMB             | The maximum journal size in MBs. 0 means unlimited. Integer values.         |
| HardLimitInPercent        | The percentage of the virtual machine volume size the journal can grow to. 0 means unlimited. Integer values. |
| WarningThresholdInMB      | The journal size, in MBs, that generates a warning that the journal is nearing its hard limit. 0 means unlimited. Integer values. |
| WarningThresholdInPercent | The percentage of the virtual machine volume size that generates a warning. 0 means unlimited. Integer values. |
| Networks                  | Information about the networks that connect the protected and recovery sites. |
| Failover                  | Information about the networks used for failover.                          |
| Hypervisor                | Information related to networks used in hypervisors.                       |
| DefaultNetworkIdentifier  | The network identifier of the network to use during a failover or move operation in which the recovered virtual machines will run. |
| VCD                       | Information related to networks used in vCDs.                              |
| FailoverTest              | Information about the networks used for testing failover.                  |
| Hypervisor                | Information related to networks used in hypervisors.                       |
| DefaultNetworkIdentifier  | The network identifier of the network to use when testing the failover of virtual machines in the recovery site. |
| VCD                       | Information related to networks used in a vCD environment.                 |
| Protected                 | Information about the protected site.                                      |
| VCD                       | Null when the protected site is hypervisor.                                |
| Recovery                  | Information about the recovery.                                           |
## DefaultDatastoreClusterIdentifier
The identifier of the default datastore cluster used in the recovery site.

*Note: Only when the recovery site is a vSphere site.*

## DefaultDatastoreIdentifier
The identifier of the default storage where the metadata files for the virtual machines are stored, such as the vmx or vhdx files. The identifier comprises the server identifier and the storage moref, with the format, `serverid.moref`.

## DefaultFolderIdentifier
The identifier of the default folder used for recovery. The identifier comprises the server identifier and the folder moref, with the format, `serverid.moref`.

## DefaultHostClusterIdentifier
The identifier of the default host cluster that handles the replicated data. The identifier comprises the server identifier and the host cluster moref, with the format, `serverid.moref`.

## DefaultHostIdentifier
The identifier of the default host that handles the replicated data. The identifier comprises the server identifier and the host moref, with the format, `serverid.moref`.

## ResourcePoolIdentifier
The identifier of the resource pool for the recovered virtual machines. The identifier comprises the server identifier and the resource pool moref, with the format, `serverid.moref`.

## VCD
Null. Information about the vCD recovery vApp.

### Scripting
Information about the scripts to run, either before or after recovery operation, or after an offsite backup is run.

#### PreRecovery
Information about scripts that are run before a recovery operation is performed.

**Command**
The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site.

**Parameters**
Parameters to pass to the script.

**TimeoutInSeconds**
The time-out, in seconds, for the script to run.

---

### PostBackup
Information about scripts that are run after an offsite backup is performed.

**PostRecovery**
Information about scripts that are run after a recovery operation is performed.

**Command**
The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site.

**Parameters**
Parameters to pass to the script.

**TimeoutInSeconds**
The time-out, in seconds, for the script to run.

---

### Vms
Information about the virtual machines in a VPG.

**VpgIdentifier**
The VPG identifier will be specified if a VPG was already created in a previous session. The session is identified by the VpgSettingsIdentifier, and must be stated when running the current API.

**VpgSettingsIdentifier**
The identifier received after running the following POST API:

```plaintext
https://zvm_ip:port/v1/vpgSettings
```
Get a Full VPG Template

After you add the values to the basic parameters, run this API to receive a more detailed template. The detailed template includes sub-parameters of the parameters to which you added values, as explained in the previous API. See Add Values in the Empty VPG Template.

<table>
<thead>
<tr>
<th>METHOD</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>https://zvm_ip:port/v1/vpgSettings/VpgSettingsIdentifier</td>
</tr>
</tbody>
</table>

Where:

- **zvm_ip**: The IP address of the Zerto Virtual Manager where the API is run.
- **port**: The port to access the Zerto Virtual Manager. The default port is 9669.
- **VpgSettingsIdentifier**: The identifier of the session.

**Request**

**Response**

The request body is empty.

**Response**

The following is an example vCD response Json body.

```json
{
  "Backup": null,
  "RepositoryIdentifier": "String content",
  "RetentionPeriod": "String content",
  "Retry": {
    "IntervalInMinutes": 2147483647,
    "Number": 2147483647,
    "Retry": Boolean
  },
  "Scheduler": {
    "DayOfWeek": "String content",
    "SchedulerPeriod": "String content",
    "TimeOfDay": "String content"
  },

  "Basic": {
    "JournalHistoryInHours": 2147483647,
    "Name": "String content",
    "Priority": "String content",
    "ProtectedSiteIdentifier": "String content",
    "RecoverySiteIdentifier": "String content",
    "RpoInSeconds": 4294967295,
    "ServiceProfileIdentifier": "String content",
    "TestIntervalInMinutes": 2147483647,
    "UseWanCompression": Boolean,
    "ZorgIdentifier": "String content"
  }
}
```
"BootGroups": [{
  "BootGroups": [{
    "BootDelayInSeconds": 4294967295,
    "BootGroupIdentifier": "String content",
    "Name": "String content"
  }
}]
,"Journal": {
  "DatastoreIdentifier": "String content",
  "Limitation": {
    "HardLimitInMB": 2147483647,
    "HardLimitInPercent": 2147483647,
    "WarningThresholdInMB": 2147483647,
    "WarningThresholdInPercent": 2147483647
  }
},
"Networks": [
  "Failover": {
    "Hypervisor": {
      "DefaultNetworkIdentifier": null
    }
  },
  "VCD": {
    "CopyNatRules": "String content",
    "IsEnableGuestCustomization": "true",
    "DefaultRecoveryOrgVdcNetworkIdentifier": "String content",
    "Mapping": {
      "ProtectedOrgVdcNetworkIdentifier": "String content",
      "RecoveryOrgVdcNetworkIdentifier": "String content",
      "ReverseTestOrgVdcNetworkIdentifier": "String content"
    }
  }
},
  "FailoverTest": {
    "Hypervisor": {
      "DefaultNetworkIdentifier": null
    }
  },
  "VCD": {
    "CopyNatRules": "String content",
    "IsEnableGuestCustomization": "true",
    "DefaultRecoveryOrgVdcNetworkIdentifier": "String content",
    "Mapping": {
      "ProtectedOrgVdcNetworkIdentifier": "String content",
      "RecoveryOrgVdcNetworkIdentifier": "String content",
      "ReverseTestOrgVdcNetworkIdentifier": "String content"
    }
  }
},
"Protected": {
  "VCD": {
    "VCDVappIdentifier": null
  }
}
"Recovery": {
  "DefaultDatastoreClusterIdentifier": "String content",
  "DefaultDatastoreIdentifier": "String content",
  "DefaultFolderIdentifier": "String content",
  "DefaultHostClusterIdentifier": "String content",
  "DefaultHostIdentifier": "String content",
  "ResourcePoolIdentifier": "String content"
}
"VCD": {
  "OrgVcdIdentifier": null
}
"Scripting": {
  "PostBackup": {
    "PostRecovery": {
      "Command": null,
      "Parameters": null
    },
    "PreRecovery": {
      "Command": null,
      "Parameters": null,
      "TimeoutInSeconds": 300
    }
  }
}
"Vms": [{
  "BootGroupIdentifier": "String content",
  "Journal": {
    "DatastoreIdentifier": "String content",
    "Limitation": {
      "HardLimitInMB": 4294967295,
      "HardLimitInPercent": 4294967295,
      "WarningThresholdInMB": 4294967295,
      "WarningThresholdInPercent": 4294967295
    }
  }
}]}
"Nics": [{
  "Failover": {
    "Hypervisor": null,
    "VCD": {
      "IpAddress": "String content",
      "IpMode": "String content",
      "IsConnected": Boolean,
      "IsPrimary": Boolean,
      "RecoveryOrgVdcNetworkIdentifier": "String content",
      "ShouldReplaceMacAddress": "String content"
    }
  },
  "FailoverTest": {
    "Hypervisor": null,
    "VCD": "String",
    "IpAddress": "String content",
    "IpMode": "String content",
    "IsConnected": Boolean,
    "IsPrimary": Boolean,
    "RecoveryOrgVdcNetworkIdentifier": "String content",
    "ShouldReplaceMacAddress": "String content"
  }
},
"NicIdentifier": "String content"
},
"Recovery": {
  "DefaultDatastoreClusterIdentifier": "String content",
  "DefaultDatastoreIdentifier": "String content",
  "DefaultFolderIdentifier": "String content",
  "DefaultHostClusterIdentifier": "String content",
  "DefaultHostIdentifier": "String content",
  "ResourcePoolIdentifier": "String content",
  "VCD": {
    "StorageProfileIdentifier": "String content"
  }
},
"VmIdentifier": "String content",
"Volumes": [{
  "Datastore": null,
  "IsSwap": Boolean,
  "VCD": {
    "IsThin": Boolean
  }
},
"VolumeIdentifier": String
}],
"VpgIdentifier": null,
"VpgSettingsIdentifier": "String content"
}]

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Backup</td>
<td>Information related to offsite backup.</td>
</tr>
<tr>
<td>RepositoryIdentifier</td>
<td>The identifier of the repository where offsite backups will be written.</td>
</tr>
</tbody>
</table>
RetentionPeriod

The length of time to keep offsite backups, up to a maximum of 12 months. Over time, Zerto reduces the number of stored offsite backups to save space.

<table>
<thead>
<tr>
<th>VALID VALUES FOR RetentionPeriod</th>
<th># OF BACKUPS SAVED WHEN RUN DAILY</th>
<th># OF BACKUPS SAVED WHEN RUN WEEKLY</th>
</tr>
</thead>
<tbody>
<tr>
<td>OneWeek</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>OneMonth</td>
<td>11</td>
<td>5</td>
</tr>
<tr>
<td>ThreeMonths</td>
<td>13</td>
<td>7</td>
</tr>
<tr>
<td>SixMonths</td>
<td>16</td>
<td>10</td>
</tr>
<tr>
<td>NineMonths</td>
<td>19</td>
<td>13</td>
</tr>
<tr>
<td>OneYear</td>
<td>22</td>
<td>16</td>
</tr>
</tbody>
</table>

The SchedulerPeriod parameter defines whether backups are created daily or weekly.

For an explanation of how Zerto reduces the number of offsite backups, see the Zerto Virtual Replication Administration Guide.

Retry

Information about backup retries.

IntervalInMinutes

How much time to wait, in minutes, after a backup job fails before running the backup job again.

Number

The number of retries that will be attempted.

Retry

True: The offsite backup job will rerun automatically if it fails.
False: The offsite backup job will not rerun automatically if it fails.

Scheduler

Offsite backup schedule settings.

DayOfWeek

The day of the week that the offsite backups will run.

SchedulerPeriod

How often are backups created. The options are:
Daily: The offsite backups will run every day.
Weekly: The offsite backups will run once a week.

TimeOfDay

The time of day when offsite backup jobs will run. The time is based on a 24-hour clock.

Basic

Basic VPG settings.

JournalHistoryInHours

The time that all write commands are saved in the journal. The value is between 1 and 336 (14 days).

Name

The name of the VPG.

Priority

The priority specified for the VPG. Possible values are:
Low: The VPG has a low priority for transferring data.
Medium: The VPG has a medium priority for transferring data.
High: The VPG has a high priority for transferring data.

ProtectedSiteIdentifier

The identifier of the source site where the VPG virtual machines will be protected. This is the site where the API runs.

RecoverySiteIdentifier

The identifier of the target site where the VPG virtual machines will be recovered.

RpoInSeconds

The maximum desired time between each automatic checkpoint being written to the journal before an alert is issued.

ServiceProfileIdentifier

The identifier of the service profile to use for the VPG when a Zerto Cloud Manager is used.
### TestIntervalInMinutes
The time, in minutes, recommended between testing the integrity of the VPG. A warning is issued if a test is not done within this time frame. Possible values are:
- **0**: No testing is expected.
- **43200** or **null**: Testing is expected monthly.
- **131040**: Testing is expected every three months.
- **262080**: Testing is expected every six months.
- **394560**: Testing is expected every nine months.
- **525600**: Testing is expected every twelve months.

### UseWanCompression
- **True**: Data will be compressed before sending it to the recovery site.
- **False**: Data will not be compressed before sending it to the recovery site.

### ZorgIdentifier
The identifier of the ZORG, Zerto organization, defined in the Zerto Cloud Manager.

### BootGroups
Information about boot groups.

- **BootDelayInSeconds**: Specifies the delay, in seconds, between starting up the virtual machines in this group and starting up the virtual machines in the next group.
- **BootGroupIdentifier**: The identifier of a boot group.
- **Name**: The name of a boot group.

### Journal
Information about the journal.

- **DatastoreIdentifier**: The identifier of the storage used by the journal. The identifier comprises the server identifier and the storage moref, with the format, `serverid.moref`.
- **Limitation**: Information about the journal limitations.
- **HardLimitInMB**: The maximum journal size in MBs. 0 means unlimited. Integer values.
- **HardLimitInPercent**: The percentage of the virtual machine volume size the journal can grow to. 0 means unlimited. Integer values.
- **WarningThresholdInMB**: The journal size, in MBs, that generates a warning that the journal is nearing its hard limit. 0 means unlimited. Integer values.
- **WarningThresholdInPercent**: The percentage of the virtual machine volume size that generates a warning. 0 means unlimited. Integer values.

### Networks
Information about the networks that connect the protected and recovery sites.

- **Failover**: Information about the networks used for failover.
- **Hypervisor**: Information about the hypervisor protected site.
- **DefaultNetworkIdentifier**: The network identifier of the network to use during a failover or move operation in which the recovered virtual machines will run.
- **VCD**: Information about the vCD protected site.
- **CopyNatRules**: Whether to copy the NAT rules on source vApp networks to the recovery vApp during recovery. Possible values are:
  - **AutoAllocatedByRecovery**
  - **KeepSourceExternalIP**
  - **NoCopy**

  The automatic setting is applied as automatic and the manual setting is applied as manual using the IPs on the source.

  When both the protected and recovery sites are vCD environments, the NAT rules on source vApp networks are automatically copied to the recovery vApp during recovery.
IsEnableGuestCustomization | True: The computer name and network settings configured for this virtual machine are applied to its Guest OS when the virtual machine is powered on. Use this option to enable re-IPing the recovered virtual machines.  
False: The computer name and network settings configured for this virtual machine are not applied to its Guest OS when the virtual machine is powered on.  
Re-IPing is not available.

DefaultRecoveryOrgVdcNetworkIdentifier | The default recovery Org vDC network to use in the recovery site.

Mapping

ProtectedOrgVdcNetworkIdentifier | The OrgVdc network used on the protected site.

RecoveryOrgVdcNetworkIdentifier | The OrgVdc network used on the recovery site.

ReverseTestOrgVdcNetworkIdentifier

FailoverTest | Information about the networks used for testing failover.

Hypervisor | Information about the hypervisor recovery site.

DefaultNetworkIdentifier | The network identifier of the network to use when testing the failover of virtual machines in the recovery site.

VCD

CopyNatRules | Whether to copy the NAT rules on source vApp networks to the recovery vApp during recovery. Possible values are:
- AutoAllocatedByRecovery
- KeepSourceExternalIP
- NoCopy
The automatic setting is applied as automatic and the manual setting is applied as manual using the IPs on the source.

When both the protected and recovery sites are vCD environments, the NAT rules on source vApp networks are automatically copied to the recovery vApp during recovery.

The automatic setting is applied as automatic and the manual setting is applied as manual using the IPs on the source.

IsEnableGuestCustomization | True: The computer name and network settings configured for this virtual machine are applied to its Guest OS when the virtual machine is powered on. Use this option to enable re-IPing the recovered virtual machines.

False: The computer name and network settings configured for this virtual machine are not applied to its Guest OS when the virtual machine is powered on.
Re-IPing is not available.

DefaultRecoveryOrgVdcNetworkIdentifier | The default recovery Org vDC network to use in the recovery site.

Mapping

ProtectedOrgVdcNetworkIdentifier | The OrgVdc network used on the protected site.

RecoveryOrgVdcNetworkIdentifier | The OrgVdc network used on the recovery site.

ReverseTestOrgVdcNetworkIdentifier

Protected

VCD

VCDVappIdentifier | The identifier of the vCD vApp to protect. When protecting virtual machines in a vCenter Server, this value is null. Get the identifier using the API List Unprotected vCD vApps in a Site.
<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recovery</td>
<td>The identifier of the storage cluster of the VRA virtual machine.</td>
</tr>
<tr>
<td>DefaultDatastoreClusterId</td>
<td>The identifier of the default datastore cluster used in the recovery site. Note: Only when the recovery site is a vSphere site.</td>
</tr>
<tr>
<td>DefaultDatastoreIdentifier</td>
<td>The identifier of the default storage where the metadata files for the virtual machines are stored, such as the vmx or vhdx files. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultFolderIdentifier</td>
<td>The identifier of the default folder used for recovery. The identifier comprises the server identifier and the folder moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultHostClusterIdentifier</td>
<td>The identifier of the default host cluster that handles the replicated data. The identifier comprises the server identifier and the host cluster moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultHostIdentifier</td>
<td>The identifier of the default host that handles the replicated data. The identifier comprises the server identifier and the host moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>ResourcePoolIdentifier</td>
<td>The identifier of the resource pool for the recovered virtual machines. The identifier comprises the server identifier and the resource pool moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>VCD</td>
<td></td>
</tr>
<tr>
<td>OrgVcdIdentifier</td>
<td>The OrgvCD identifier when recovering to vCD. Get the identifier using VMware only: Resource Pools.</td>
</tr>
<tr>
<td>Scripting</td>
<td>Information about the scripts to run, either before or after recovery operation, or after an offsite backup is run.</td>
</tr>
<tr>
<td>PostBackup: null</td>
<td>Information about scripts that are run after an offsite backup is performed.</td>
</tr>
<tr>
<td>PostRecovery</td>
<td>Information about scripts that are run after a recovery operation is performed.</td>
</tr>
<tr>
<td>Command</td>
<td>The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site.</td>
</tr>
<tr>
<td>Parameters</td>
<td>Parameters to pass to the script.</td>
</tr>
<tr>
<td>TimeoutInSeconds</td>
<td>The time-out, in seconds, for the script to run.</td>
</tr>
<tr>
<td>PreRecovery</td>
<td>Information about scripts that are run before a recovery operation is performed.</td>
</tr>
<tr>
<td>Command</td>
<td>The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site.</td>
</tr>
<tr>
<td>Parameters</td>
<td>Parameters to pass to the script.</td>
</tr>
<tr>
<td>TimeoutInSeconds</td>
<td>The time-out, in seconds, for the script to run.</td>
</tr>
<tr>
<td>Vms</td>
<td>Information about the virtual machines in a VPG.</td>
</tr>
<tr>
<td>BootGroupIdentifier</td>
<td>The boot group identifier of a virtual machine.</td>
</tr>
<tr>
<td>Journal</td>
<td>Journal information of the virtual machine</td>
</tr>
<tr>
<td>DatastoreIdentifier</td>
<td>The identifier of the storage used by the journal for the virtual machine. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref</td>
</tr>
<tr>
<td>Limitation</td>
<td>Information about the journal limitations.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>HardLimitInMB</td>
<td>The maximum journal size in MBs. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>HardLimitInPercent</td>
<td>The percentage of the virtual machine volume size the journal can grow to. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>WarningThresholdInMB</td>
<td>The journal size, in MBs, that generates a warning that the journal is nearing its hard limit. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>WarningThresholdInPercent</td>
<td>The percentage of the virtual machine volume size that generates a warning. 0 means unlimited. Integer values.</td>
</tr>
<tr>
<td>Nics</td>
<td>Information about NICs used by the virtual machine in the VPG.</td>
</tr>
<tr>
<td>Failover</td>
<td>Information about the networks used for failover by this virtual machine.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td>Information about the networks used for failover in a hypervisor.</td>
</tr>
<tr>
<td>VCD</td>
<td></td>
</tr>
<tr>
<td>IpAddress</td>
<td>The IP address of the vCD.</td>
</tr>
<tr>
<td>IpMode</td>
<td>The mask for the network.</td>
</tr>
<tr>
<td>IsConnected</td>
<td>True: DHCP will be used for the virtual machine on the recovery site. False: A static IP address will be used for the virtual machine on the recovery site.</td>
</tr>
<tr>
<td>IsPrimary</td>
<td>True: the DNS server that handles Internet protocol mapping is the primary. False: the DNS server that handles Internet protocol mapping is not the primary.</td>
</tr>
<tr>
<td>RecoveryOrgVdcNetworkIdentifier</td>
<td>The OrgVdc network used on the recovery site.</td>
</tr>
<tr>
<td>ShouldReplaceMacAddress</td>
<td>True: The Media Access Control (MAC) address used on the protected site will be copied to the recovery site. False: The Media Access Control (MAC) address used on the protected site will not be copied to the recovery site.</td>
</tr>
<tr>
<td>FailoverTest</td>
<td>Information about the networks used for testing failover by this virtual machine.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td></td>
</tr>
<tr>
<td>VCD</td>
<td></td>
</tr>
<tr>
<td>IpAddress</td>
<td>The IP address of the vCD.</td>
</tr>
<tr>
<td>IpMode</td>
<td>The mask for the network.</td>
</tr>
<tr>
<td>IsConnected</td>
<td>True: DHCP will be used for the virtual machine on the recovery site. False: A static IP address will be used for the virtual machine on the recovery site.</td>
</tr>
<tr>
<td>IsPrimary</td>
<td>True: the DNS server that handles Internet protocol mapping is the primary. False: the DNS server that handles Internet protocol mapping is not the primary.</td>
</tr>
<tr>
<td>RecoveryOrgVdcNetworkIdentifier</td>
<td>The OrgVdc network used on the recovery site.</td>
</tr>
<tr>
<td>ShouldReplaceMacAddress</td>
<td>True: The Media Access Control (MAC) address used on the protected site will be copied to the recovery site. False: The Media Access Control (MAC) address used on the protected site will not be copied to the recovery site.</td>
</tr>
<tr>
<td>NicIdentifier</td>
<td>The identifier of the NIC for which settings are returned.</td>
</tr>
<tr>
<td>Recovery</td>
<td>Information about the recovery site.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>DatastoreClusterIdentifier</td>
<td>The identifier of the datastore cluster used in the recovery site for the VM. Note: Only when the recovery site is a vSphere site.</td>
</tr>
<tr>
<td>DefaultDatastoreIdentifier</td>
<td>The identifier of the storage. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultFolderIdentifier</td>
<td>The identifier of the folder used for recovery by the virtual machine. The identifier comprises the server identifier and the folder moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultHostClusterIdentifier</td>
<td>The identifier of the host cluster that handles the replicated data. The identifier comprises the server identifier and the host cluster moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultHostIdentifier</td>
<td>The identifier of the host that handles the replicated data. The identifier comprises the server identifier and the host moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultResourcePoolIdentifier</td>
<td>The identifier of the resource pool for the recovered virtual machines. The identifier comprises the server identifier and the resource pool moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>VmIdentifier</td>
<td>The identifier of the virtual machine. The identifier comprises the server identifier and the virtual machine moref, with the format, serverid.more.</td>
</tr>
<tr>
<td>Volumes</td>
<td>Information about the volumes used by the virtual machine.</td>
</tr>
<tr>
<td>Datastore</td>
<td>Information about the datastore used by the virtual machine.</td>
</tr>
<tr>
<td>DatastoreClusterIdentifier</td>
<td>The identifier of the datastore cluster used in the recovery site for the volume. Note: Only when the recovery site is a vSphere site.</td>
</tr>
<tr>
<td>IsSwap</td>
<td>True: The recovery disk is marked as a temp data disk. False: The recovery disk is not marked as a temp data disk.</td>
</tr>
<tr>
<td>VCD</td>
<td>IsThin True: The recovery volumes are thin-provisioned. False: The recovery volumes are not thin-provisioned.</td>
</tr>
<tr>
<td>ExistingVolume</td>
<td>For future use.</td>
</tr>
<tr>
<td>DatastoreIdentifier</td>
<td>For future use.</td>
</tr>
<tr>
<td>ExistedVmIdentifier</td>
<td>For future use.</td>
</tr>
<tr>
<td>Mode</td>
<td>For future use.</td>
</tr>
<tr>
<td>Path</td>
<td>For future use.</td>
</tr>
<tr>
<td>VolumelIdentifier</td>
<td>The identifier of the volume.</td>
</tr>
<tr>
<td>VpgIdentifier</td>
<td>The VPG identifier will be specified if a VPG was already created in a previous session.</td>
</tr>
<tr>
<td>VpgSettingsIdentifier</td>
<td>The identifier received after running the following POST API: https://zvm_ip:port/v1/vpgSettings</td>
</tr>
</tbody>
</table>
Two Step Operation

Using this method to create a VPG, you run the following APIs:
1. “Create an Empty VPG Template”, on page 221
2. “Add Values in the Full VPG Template”, on page 221

Create an Empty VPG Template

Run this API to get a VPG settings identifier. This action creates an empty VPG template.

Request
Response

<table>
<thead>
<tr>
<th>METHOD</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>https://zvm_ip:port/v1/vpgSettings</td>
</tr>
</tbody>
</table>

Where:

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>zvm_ip</td>
<td>The IP address of the Zerto Virtual Manager where the API is run.</td>
</tr>
<tr>
<td>port</td>
<td>The port to access the Zerto Virtual Manager. The default port is 9669.</td>
</tr>
</tbody>
</table>

Request
The request body is empty.

Response
The following is an example response.

```json
{
    "VpgSettingsIdentifier": "f9e631d3-9bfb-463a-9958-2fa2c6ac7640"
}
```

Add Values in the Full VPG Template

Request
Response

<table>
<thead>
<tr>
<th>METHOD</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>https://zvm_ip:port/v1/vpgSettings</td>
</tr>
</tbody>
</table>

Where:
### Request

The following is an example vCD request JSON body.

```json
[
    {
        "Backup": {
            "RepositoryIdentifier": "String Content",
            "RetentionPeriod": ,
            "Retry": {
                "IntervalInMinutes":2147483647,
                "Number":2147483647,
                "Retry": Boolean
            },
            "Scheduler": {
                "DayOfWeek": "String Content",
                "SchedulerPeriod": "String Content",
                "TimeOfDay": "String Content"
            }
        },
        "Basic": {
            "JournalHistoryInHours":2147483647,
            "Name": "String Content",
            "Priority": "String Content",
            "ProtectedSiteIdentifier": "String Content",
            "RecoverySiteIdentifier": "String Content",
            "RpoInSeconds":4294967295,
            "ServiceProfileIdentifier": "String Content",
            "TestIntervalInMinutes":2147483647,
            "UseWanCompression": Boolean,
            "ZorgIdentifier": "String Content"
        },
        "BootGroups": {
            "BootGroups":[
                {
                    "BootDelayInSeconds":4294967295,
                    "BootGroupIdentifier": "String Content",
                    "Name": "String Content"
                }
            ]
        },
        "Journal": {
            "DatastoreIdentifier": "String Content",
            "Limitation": {
                "HardLimitInMB":2147483647,
                "HardLimitInPercent":2147483647,
                "WarningThresholdInMB":2147483647,
                "WarningThresholdInPercent":2147483647
            }
        }
    }
]
```
"Networks": {
  "Failover": {
    "Hypervisor": {
      "DefaultNetworkIdentifier": "String Content"
    }
  },
  "VCD": {
    "CopyNatRules": "String Content",
    "IsEnableGuestCustomization": Boolean,
    "DefaultRecoveryOrgVdcNetworkIdentifier": "String Content"
    "Mapping": {
      "ProtectedOrgVdcNetworkIdentifier": "String Content",
      "RecoveryOrgVdcNetworkIdentifier": "String Content",
      "ReverseTestOrgVdcNetworkIdentifier": "String Content"
    }
  }
},
"FailoverTest": {
  "Hypervisor": {
    "DefaultNetworkIdentifier": "String Content"
  },
  "VCD": {
    "CopyNatRules": "String Content",
    "IsEnableGuestCustomization": Boolean,
    "DefaultRecoveryOrgVdcNetworkIdentifier": "String Content",
    "Mapping": {
      "ProtectedOrgVdcNetworkIdentifier": "String Content",
      "RecoveryOrgVdcNetworkIdentifier": "String Content",
      "ReverseTestOrgVdcNetworkIdentifier": "String Content"
    }
  }
},
"Protected": {
  "VCD": {
    "VCDVappIdentifier": "String Content"
  }
}
"Recovery": {
  "DefaultDatastoreClusterIdentifier": "String Content",
  "DefaultDatastoreIdentifier": "String Content",
  "DefaultFolderIdentifier": "String Content",
  "DefaultHostClusterIdentifier": "String Content",
  "DefaultHostIdentifier": "String Content",
  "ResourcePoolIdentifier": "String Content"
  "VCD": {
    "OrgVcdIdentifier": "String Content"
  }
}
"Scripting": {
  "PostBackup": {
    "Command": "String Content",
    "Parameters": "String Content",
    "TimeoutInSeconds": Number
  },
  "PreRecovery": {
    "Command": "String Content",
    "Parameters": "String Content",
    "TimeoutInSeconds": Number
  }
},

"Vms": {
  "BootGroupIdentifier": ,
  "Journal": {
    "DatastoreIdentifier": "String Content",
    "Limitation": {
      "HardLimitInMB": Number,
      "HardLimitInPercent": Number,
      "WarningThresholdInMB": Number,
      "WarningThresholdInPercent": Number
    }
  },
  "Nics": {
    "Failover": {
      "Hypervisor": "String Content",
      "VCD": {
        "IpAddress": "String Content",
        "IpMode": "String Content",
        "IsConnected": Boolean,
        "IsPrimary": Boolean,
        "RecoveryOrgVdcNetworkIdentifier": "String Content",
        "ShouldReplaceMacAddress": Boolean
      }
    },
    "FailoverTest": {
      "Hypervisor": "String Content",
      "VCD": {
        "IpAddress": "String Content",
        "IpMode": "String Content",
        "IsConnected": Boolean,
        "IsPrimary": Boolean
        "RecoveryOrgVdcNetworkIdentifier": "String Content",
        "ShouldReplaceMacAddress": Boolean
      }
    },
    "NicIdentifier": "String Content"
  }
}
"Recovery": {  
  "DefaultDatastoreClusterIdentifier": "String content",  
  "DefaultDatastoreIdentifier": "String content",  
  "DefaultFolderIdentifier": "String content",  
  "DefaultHostClusterIdentifier": "String content",  
  "DefaultHostIdentifier": "String content",  
  "ResourcePoolIdentifier": "String Content"  
  "VCD": {  
    "StorageProfileIdentifier": "String content"  
  }  
},  
"VmIdentifier": "String content",  
"Volumes": {  
  "Datastore": "String Content",  
  "IsSwap": Boolean  
  "VCD": {  
    "IsThin": Boolean  
  }  
  "VolumeIdentifier": "String content"  
  
}  
"VpgIdentifier": "String Content",  
"VpgSettingsIdentifier": "String content" ]

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Backup</td>
<td>Information related to offsite backup.</td>
</tr>
<tr>
<td>RepositoryIdentifier</td>
<td>The identifier of the repository where offsite backups will be written.</td>
</tr>
<tr>
<td>RetentionPeriod</td>
<td>The length of time to keep offsite backups, up to a maximum of 12 months. Over time, Zerto reduces the number of stored offsite backups to save space.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>VALID VALUES FOR RetentionPeriod</th>
<th># OF BACKUPS SAVED WHEN RUN DAILY</th>
<th># OF BACKUPS SAVED WHEN RUN WEEKLY</th>
</tr>
</thead>
<tbody>
<tr>
<td>OneWeek</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>OneMonth</td>
<td>11</td>
<td>5</td>
</tr>
<tr>
<td>ThreeMonths</td>
<td>13</td>
<td>7</td>
</tr>
<tr>
<td>SixMonths</td>
<td>16</td>
<td>10</td>
</tr>
<tr>
<td>NineMonths</td>
<td>19</td>
<td>13</td>
</tr>
<tr>
<td>OneYear</td>
<td>22</td>
<td>16</td>
</tr>
</tbody>
</table>

The SchedulerPeriod parameter defines whether backups are created daily or weekly.

For an explanation of how Zerto reduces the number of offsite backups, see the Zerto Virtual Replication Administration Guide.

<table>
<thead>
<tr>
<th>Retry</th>
<th>Information about backup retries.</th>
</tr>
</thead>
<tbody>
<tr>
<td>IntervallnMinutes</td>
<td>How much time to wait, in minutes, after a backup job fails before running the backup job again.</td>
</tr>
<tr>
<td>Number</td>
<td>The number of retries that will be attempted.</td>
</tr>
<tr>
<td>Retry</td>
<td>True: The offsite backup job will rerun automatically if it fails. False: The offsite backup job will not rerun automatically if it fails.</td>
</tr>
<tr>
<td>Scheduler</td>
<td>Offsite backup schedule settings.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>DayOfWeek</td>
<td>The day of the week that the offsite backups will run.</td>
</tr>
<tr>
<td>SchedulerPeriod</td>
<td><strong>Daily</strong>: The offsite backups will run every day.</td>
</tr>
<tr>
<td></td>
<td><strong>Weekly</strong>: The offsite backups will run once a week.</td>
</tr>
<tr>
<td>TimeOfDay</td>
<td>The time of day when offsite backup jobs will run. The time is based on a 24-hour clock.</td>
</tr>
<tr>
<td>Basic</td>
<td>Basic VPG settings.</td>
</tr>
<tr>
<td>JournalHistoryInHours</td>
<td>The time that all write commands are saved in the journal. The value is between 1 and 336 (14 days).</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the VPG.</td>
</tr>
<tr>
<td>Priority</td>
<td>The priority specified for the VPG. Possible values are:</td>
</tr>
<tr>
<td></td>
<td><strong>Low</strong>: The VPG has a low priority for transferring data.</td>
</tr>
<tr>
<td></td>
<td><strong>Medium</strong>: The VPG has a medium priority for transferring data.</td>
</tr>
<tr>
<td></td>
<td><strong>High</strong>: The VPG has a high priority for transferring data.</td>
</tr>
<tr>
<td>ProtectedSiteIdentifier</td>
<td>The identifier of the source site where the VPG virtual machines will be protected. This is the site where the API runs.</td>
</tr>
<tr>
<td>RecoverySiteIdentifier</td>
<td>The identifier of the target site where the VPG virtual machines will be recovered.</td>
</tr>
<tr>
<td>RpOilInSeconds</td>
<td>The maximum desired time between each automatic checkpoint being written to the journal before an alert is issued.</td>
</tr>
<tr>
<td></td>
<td>Default appears if default service profile is selected.</td>
</tr>
<tr>
<td>ServiceProfileIdentifier</td>
<td>The identifier of the service profile to use for the VPG when a Zerto Cloud Manager is used.</td>
</tr>
<tr>
<td>TestIntervalInMinutes</td>
<td>The time, in minutes, recommended between testing the integrity of the VPG. A warning is issued if a test is not done within this time frame.</td>
</tr>
<tr>
<td></td>
<td>Possible values are:</td>
</tr>
<tr>
<td></td>
<td><strong>0</strong>: No testing is expected.</td>
</tr>
<tr>
<td></td>
<td><strong>43200</strong> or <strong>null</strong>: Testing is expected monthly.</td>
</tr>
<tr>
<td></td>
<td><strong>131040</strong>: Testing is expected every three months.</td>
</tr>
<tr>
<td></td>
<td><strong>262080</strong>: Testing is expected every six months.</td>
</tr>
<tr>
<td></td>
<td><strong>394560</strong>: Testing is expected every nine months.</td>
</tr>
<tr>
<td></td>
<td><strong>525600</strong>: Testing is expected every twelve months.</td>
</tr>
<tr>
<td>UseWanCompression</td>
<td><strong>True</strong>: Data will be compressed before sending it to the recovery site.</td>
</tr>
<tr>
<td></td>
<td><strong>False</strong>: Data will not be compressed before sending it to the recovery site.</td>
</tr>
<tr>
<td>ZorgIdentifier</td>
<td>The identifier of the ZORG, Zerto organization, defined in the Zerto Cloud Manager.</td>
</tr>
<tr>
<td>BootGroups</td>
<td>Information about boot groups.</td>
</tr>
<tr>
<td>BootGroups</td>
<td></td>
</tr>
<tr>
<td>BootDelayInSeconds</td>
<td>Specifies the delay, in seconds, between starting up the virtual machines in this group and starting up the virtual machines in the next group.</td>
</tr>
<tr>
<td>Name</td>
<td>The identifier of a boot group.</td>
</tr>
<tr>
<td>Journal</td>
<td>Information about the journal.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>DatastoreIdentifier</td>
<td>The identifier of the storage used by the journal. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>Limitation</td>
<td>Information about the journal limitations.</td>
</tr>
<tr>
<td>HardLimitInMB</td>
<td>The maximum journal size in MBs. 0 means unlimited. Can be either in MB or in percentage.</td>
</tr>
<tr>
<td>HardLimitInPercent</td>
<td>The percentage of the virtual machine volume size the journal can grow to. 0 means unlimited. Can be either in MB or in percentage.</td>
</tr>
<tr>
<td>WarningThresholdInMB</td>
<td>The journal size, in MBs, that generates a warning that the journal is nearing its hard limit. 0 means unlimited. Can be either in MB or in percentage.</td>
</tr>
<tr>
<td>WarningThresholdInPercent</td>
<td>The percentage of the virtual machine volume size that generates a warning. 0 means unlimited. Integer values. Can be either in MB or in percentage.</td>
</tr>
<tr>
<td>Networks</td>
<td>Information about the networks that connect the protected and recovery sites.</td>
</tr>
<tr>
<td>Failover</td>
<td>Information about the networks used for failover.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td>Information about the hypervisor protected site.</td>
</tr>
<tr>
<td>DefaultNetworkIdentifier</td>
<td>The network identifier of the network to use during a failover or move operation in which the recovered virtual machines will run.</td>
</tr>
<tr>
<td>VCD</td>
<td>Information about the vCD protected site.</td>
</tr>
<tr>
<td>CopyNatRules</td>
<td>Whether to copy the NAT rules on source vApp networks to the recovery vApp during recovery. Possible values are:</td>
</tr>
<tr>
<td></td>
<td>auto</td>
</tr>
<tr>
<td></td>
<td>source</td>
</tr>
<tr>
<td></td>
<td>dont</td>
</tr>
<tr>
<td></td>
<td>The automatic setting is applied as automatic and the manual setting is applied as manual using the IPs on the source.</td>
</tr>
<tr>
<td></td>
<td>When both the protected and recovery sites are vCD environments, the NAT rules on source vApp networks are automatically copied to the recovery vApp during recovery.</td>
</tr>
<tr>
<td>IsEnableGuestCustomization</td>
<td>True: The computer name and network settings configured for this virtual machine are applied to its Guest OS when the virtual machine is powered on. Use this option to enable re-IPing the recovered virtual machines.</td>
</tr>
<tr>
<td></td>
<td>False: The computer name and network settings configured for this virtual machine are not applied to its Guest OS when the virtual machine is powered on. Re-IPing is not available.</td>
</tr>
<tr>
<td>DefaultRecoveryOrgVdcNetworkIdentifier</td>
<td>The default recovery Org vDC network to use in the recovery site.</td>
</tr>
<tr>
<td>Mapping</td>
<td>The OrgVdc network used on the protected site.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-----------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>RecoveryOrgVdcNetworkIdentifier</td>
<td>The OrgVdc network used on the recovery site. Should this be added: ...recovery site, for failover/move.</td>
</tr>
<tr>
<td>ReverseTestOrgVdcNetworkIdentifier</td>
<td>The OrgVdc network used on the recovery site for when testing failover.</td>
</tr>
<tr>
<td>FailoverTest</td>
<td>Information about the networks used for testing failover.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td>-</td>
</tr>
<tr>
<td>DefaultNetworkIdentifier</td>
<td>The network identifier of the network to use when testing the failover of virtual machines in the recovery site.</td>
</tr>
<tr>
<td>VCD</td>
<td></td>
</tr>
<tr>
<td>CopyNatRules</td>
<td>Whether to copy the NAT rules on source vApp networks to the recovery vApp during recovery. Possible values are: auto, source, dont. The automatic setting is applied as automatic and the manual setting is applied as manual using the IPs on the source. When both the protected and recovery sites are vCD environments, the NAT rules on source vApp networks are automatically copied to the recovery vApp during recovery. The automatic setting is applied as automatic and the manual setting is applied as manual using the IPs on the source.</td>
</tr>
<tr>
<td>IsEnableGuestCustomization</td>
<td>True: The computer name and network settings configured for this virtual machine are applied to its Guest OS when the virtual machine is powered on. Use this option to enable re-IPing the recovered virtual machines. False: The computer name and network settings configured for this virtual machine are not applied to its Guest OS when the virtual machine is powered on. Re-IPing is not available.</td>
</tr>
<tr>
<td>DefaultRecoveryOrgVdcNetworkIdentifier</td>
<td>The default recovery Org vDC network to use in the recovery site.</td>
</tr>
<tr>
<td>Mapping</td>
<td></td>
</tr>
<tr>
<td>ProtectedOrgVdcNetworkIdentifier</td>
<td>The OrgVdc network used on the protected site.</td>
</tr>
<tr>
<td>RecoveryOrgVdcNetworkIdentifier</td>
<td>The OrgVdc network used on the recovery site.</td>
</tr>
<tr>
<td>ReverseTestOrgVdcNetworkIdentifier</td>
<td>The OrgVdc network used on the recovery site for when testing failover.</td>
</tr>
<tr>
<td>Protected</td>
<td></td>
</tr>
<tr>
<td>VCD</td>
<td></td>
</tr>
<tr>
<td>VCDVappId</td>
<td>The identifier of the protected vCD vApp.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Recovery</td>
<td>The identifier of the storage cluster of the VRA virtual machine.</td>
</tr>
<tr>
<td>DefaultDatastoreClusterIdentifier</td>
<td>The identifier of the default datastore cluster used in the recovery site. Note: Only when the recovery site is a vSphere site.</td>
</tr>
<tr>
<td>DefaultDatastoreIdentifier</td>
<td>The identifier of the default storage where the metadata files for the virtual machines are stored, such as the vmx or vhdx files. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultFolderIdentifier</td>
<td>The identifier of the default folder used for recovery. The identifier comprises the server identifier and the folder moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultHostClusterIdentifier</td>
<td>The identifier of the default host cluster that handles the replicated data. The identifier comprises the server identifier and the folder moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>DefaultHostIdentifier</td>
<td>The identifier of the default host that handles the replicated data. The identifier comprises the server identifier and the host cluster moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>ResourcePoolIdentifier</td>
<td>The identifier of the resource pool for the recovered virtual machines. The identifier comprises the server identifier and the host moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>OrgVcdIdentifier</td>
<td>The OrgvCD identifier when recovering to vCD. Get the identifier using VMware only: Resource Pools.</td>
</tr>
<tr>
<td>Scripting</td>
<td>Information about the scripts to run, either before or after recovery operation, or after an offsite backup is run.</td>
</tr>
<tr>
<td>PostBackup</td>
<td>Information about scripts that are run after an offsite backup is performed.</td>
</tr>
<tr>
<td>PostRecovery</td>
<td>Information about scripts that are run after a recovery operation is performed.</td>
</tr>
<tr>
<td>Command</td>
<td>The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site.</td>
</tr>
<tr>
<td>Parameters</td>
<td>Parameters to pass to the script.</td>
</tr>
<tr>
<td>TimeoutInSeconds</td>
<td>The time-out, in seconds, for the script to run.</td>
</tr>
<tr>
<td>PreRecovery</td>
<td>Information about scripts that are run before a recovery operation is performed.</td>
</tr>
<tr>
<td>Command</td>
<td>The full path of the script. The script must be located on the same machine as the Zerto Virtual Manager for the recovery site.</td>
</tr>
<tr>
<td>Parameters</td>
<td>Parameters to pass to the script.</td>
</tr>
<tr>
<td>TimeoutInSeconds</td>
<td>The time-out, in seconds, for the script to run.</td>
</tr>
<tr>
<td>Vms</td>
<td>Information about the virtual machines in a VPG.</td>
</tr>
<tr>
<td>BootGroupIdentifier</td>
<td>The boot group identifier of a virtual machine.</td>
</tr>
<tr>
<td>Journal</td>
<td>Journal information of the virtual machine</td>
</tr>
<tr>
<td>DatastoreIdentifier</td>
<td>The identifier of the storage used by the journal for the virtual machine. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>Limitation</td>
<td>Information about the journal limitations.</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>HardLimitInMB</td>
<td>The maximum journal size in MBs. 0 means unlimited. Integer values. Can be either in MB or in percentage.</td>
</tr>
<tr>
<td>HardLimitInPercent</td>
<td>The percentage of the virtual machine volume size the journal can grow to. 0 means unlimited. Integer values. Can be either in MB or in percentage.</td>
</tr>
<tr>
<td>WarningThresholdInMB</td>
<td>The journal size, in MBs, that generates a warning that the journal is nearing its hard limit. 0 means unlimited. Integer values. Can be either in MB or in percentage.</td>
</tr>
<tr>
<td>WarningThresholdInPercent</td>
<td>The percentage of the virtual machine volume size that generates a warning. 0 means unlimited. Integer values. Can be either in MB or in percentage.</td>
</tr>
<tr>
<td>Nics</td>
<td>Information about NICs used by the virtual machine in the VPG.</td>
</tr>
<tr>
<td>Failover</td>
<td>Information about the networks used for failover by this virtual machine.</td>
</tr>
<tr>
<td>Hypervisor</td>
<td>Not active when running a vCD API</td>
</tr>
<tr>
<td>VCD</td>
<td>The DNS name excluding the host.</td>
</tr>
<tr>
<td>IpAddress</td>
<td>Information about the IP configuration of the recovery site used for failovers.</td>
</tr>
<tr>
<td>IpMode</td>
<td>The mask for the network.</td>
</tr>
<tr>
<td>IsConnected</td>
<td>True: DHCP will be used for the virtual machine on the recovery site.</td>
</tr>
<tr>
<td></td>
<td>False: A static IP address will be used for the virtual machine on the recovery site.</td>
</tr>
<tr>
<td>IsPrimary</td>
<td>The IP address of the primary DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>RecoveryOrgVdcNetworkId</td>
<td>The IP address of the alternate, secondary, DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>ShouldReplaceMacAddress</td>
<td>True: The Media Access Control (MAC) address used on the protected site will be copied to the recovery site.</td>
</tr>
<tr>
<td></td>
<td>False: The Media Access Control (MAC) address used on the protected site will not be copied to the recovery site.</td>
</tr>
<tr>
<td>FailoverTest</td>
<td>Not active when running a vCD API</td>
</tr>
<tr>
<td>Hypervisor</td>
<td>The DNS name excluding the host.</td>
</tr>
<tr>
<td>VCD</td>
<td>Information about the IP configuration of the recovery site used for failovers.</td>
</tr>
<tr>
<td>IpMode</td>
<td>The mask for the network.</td>
</tr>
<tr>
<td>IsConnected</td>
<td>True: DHCP will be used for the virtual machine on the recovery site.</td>
</tr>
<tr>
<td></td>
<td>False: A static IP address will be used for the virtual machine on the recovery site.</td>
</tr>
<tr>
<td>IsPrimary</td>
<td>The IP address of the primary DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>RecoveryOrgVdcNetworkId</td>
<td>The IP address of the alternate, secondary, DNS server that handles Internet protocol mapping.</td>
</tr>
<tr>
<td>ShouldReplaceMacAddress</td>
<td>True: The Media Access Control (MAC) address used on the protected site will be copied to the recovery site.</td>
</tr>
<tr>
<td></td>
<td>False: The Media Access Control (MAC) address used on the protected site will not be copied to the recovery site.</td>
</tr>
</tbody>
</table>
## NicIdentifier
The identifier of the NIC for which settings are returned.

## Recovery
Information about the recovery site.

## DatastoreClusterIdentifier
The identifier of the datastore cluster used in the recovery site for the VM.

Note: Only when the recovery site is a vSphere site.

## DefaultDatastoreIdentifier
The identifier of the storage. The identifier comprises the server identifier and the storage moref, with the format, serverid.moref.

## DefaultFolderIdentifier
The identifier of the folder used for recovery by the virtual machine. The identifier comprises the server identifier and the folder moref, with the format, serverid.moref.

## DefaultHostClusterIdentifier
The identifier of the host cluster that handles the replicated data. The identifier comprises the server identifier and the host cluster moref, with the format, serverid.moref.

## DefaultHostIdentifier
The identifier of the host that handles the replicated data. The identifier comprises the server identifier and the host moref, with the format, serverid.moref.

## DefaultResourcePoolIdentifier
The identifier of the resource pool for the recovered virtual machines. The identifier comprises the server identifier and the resource pool moref, with the format, serverid.moref.

## ResourcePoolIdentifier
The identifier of the resource pool for the recovered virtual machines.

## VCD

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>StorageProfileIdentifier</td>
<td>The identifier of the virtual machine. The identifier comprises the server identifier and the virtual machine moref, with the format, serverid.moref.</td>
</tr>
<tr>
<td>VmIdentifier</td>
<td>Information about the volumes used by the virtual machine.</td>
</tr>
<tr>
<td>Volumes</td>
<td>Information about the datastore used by the virtual machine.</td>
</tr>
<tr>
<td>IsSwap</td>
<td><strong>True</strong>: The recovery disk is marked as a temp data disk. <strong>False</strong>: The recovery disk is not marked as a temp data disk.</td>
</tr>
<tr>
<td>VCD</td>
<td><strong>True</strong>: The recovery volumes are thin-provisioned. <strong>False</strong>: The recovery volumes are not thin-provisioned.</td>
</tr>
<tr>
<td>VmIdentifier</td>
<td>The identifier of the volume.</td>
</tr>
</tbody>
</table>

## Response
The response body is empty.

Back to [VPG Management APIs](#)
Back to [Managing vCD APIs](#)
Back to [All APIs](#)
## Settings Retained when Replicating from a Protected Site vCloud Director to a Recovery Site vCloud Director

The following tables display settings that are retained when replicating from a protected site vCloud Director to a recovery site vCloud Director:

- Edge Gateway Services, on page 232
- vApp Properties, on page 232
- Network, on page 232
- VM Properties, on page 232

### Edge Gateway Services

<table>
<thead>
<tr>
<th>SETTING</th>
<th>SETTING RETAINED?</th>
</tr>
</thead>
<tbody>
<tr>
<td>DHCP</td>
<td>No</td>
</tr>
<tr>
<td>Firewall</td>
<td>No</td>
</tr>
<tr>
<td>Static Routing</td>
<td>No</td>
</tr>
<tr>
<td>NAT</td>
<td>Configurable</td>
</tr>
</tbody>
</table>

### vApp Properties

<table>
<thead>
<tr>
<th>SETTING</th>
<th>SETTING RETAINED?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Leases (Runtime / Storage)</td>
<td>No</td>
</tr>
<tr>
<td>vApp Description</td>
<td>No</td>
</tr>
<tr>
<td>VM Start / Stop</td>
<td>Yes</td>
</tr>
<tr>
<td>Sharing</td>
<td>No</td>
</tr>
<tr>
<td>Metadata</td>
<td>Yes</td>
</tr>
</tbody>
</table>

### Network

<table>
<thead>
<tr>
<th>SETTING</th>
<th>SETTING RETAINED?</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORG Network</td>
<td>No</td>
</tr>
<tr>
<td>(protected ORG vDC networks need to be mapped to recovery ORG vDC networks)</td>
<td></td>
</tr>
<tr>
<td>Isolated vApp network</td>
<td>Yes</td>
</tr>
<tr>
<td>(same gateway address used)</td>
<td></td>
</tr>
<tr>
<td>Routed vApp Network</td>
<td>Yes</td>
</tr>
<tr>
<td>(same gateway address used, routed organization network mapped)</td>
<td></td>
</tr>
<tr>
<td>vApp Network Without NIC On It</td>
<td>Yes</td>
</tr>
<tr>
<td>(same gateway address used)</td>
<td></td>
</tr>
</tbody>
</table>

### VM Properties

<table>
<thead>
<tr>
<th>SETTING</th>
<th>SETTING RETAINED?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metadata and Description</td>
<td>No</td>
</tr>
</tbody>
</table>
CHAPTER 5: XML REQUEST AND RESPONSE FORMATS

This section provides examples of Zerto Virtual Replication RESTful API XML request and response formats.

- “V1/ API XML Response Format”, below
- “Alerts API XML Response Format”, on page 233
- “Events API XML Response Format”, on page 234
- “Local Site API XML Response Format”, on page 237
- “Peer Sites API XML Request and Response Formats”, on page 238
- “Service Profiles API XML Response Formats”, on page 239
- “Session API XML Request Format”, on page 240
- “Tasks API XML Response Format”, on page 240
- “Virtualization Sites API XML Response Format”, on page 242
- “VMs API XML Response Format”, on page 243
- “VPGs API XML Request and Response Formats”, on page 245
- “VPG Settings API XML Request and Response Formats”, on page 249
- “VRAs API XML Request and Response Formats”, on page 272
- “ZORG API XML Response Format”, on page 276

V1/ API XML Response Format

The following is an example response XML body for https://zvm_ip:port/v1/.

```
<ArrayOfLink_String xmlns="http://schemas.zerto.com/zvm/api">
    <Link_String>
        <href>String content</href>
        <rel>String content</rel>
        <type>String content</type>
    </Link_String>
    ...
</ArrayOfLink_String>
```

Alerts API XML Response Format

The following is an example response XML body for https://zvm_ip:port/v1/alerts and, without the array statement, for https://zvm_ip:port/v1/alerts/{alertId}.

```
<ArrayOfAlertApi xmlns="http://schemas.zerto.com/zvm/api">
    <AlertApi>
        <AffectedVpgs>
            <ResourceLink>
                <href>String content</href>
                <identifier>String content</identifier>
                <rel>String content</rel>
                <type>String content</type>
            </ResourceLink>
            ...
        </AffectedVpgs>
    </AlertApi>
</ArrayOfAlertApi>
```

```xml
<ArrayOfAlertApi xmlns="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
    <AlertApi>
        <AffectedZorgs>
            <ResourceLink>
                <href>String content</href>
                <identifier>String content</identifier>
                <rel>String content</rel>
                <type>String content</type>
            </ResourceLink>
            ...
        </AffectedZorgs>
        <Description>String content</Description>
        <Entity>String content</Entity>
        <HelpIdentifier>String content</HelpIdentifier>
        <IsDismissed>true</IsDismissed>
        <Level>String content</Level>
        <Link>
            <href>String content</href>
            <identifier>String content</identifier>
            <rel>String content</rel>
            <type>String content</type>
        </Link>
        <Site>
            <href>String content</href>
            <identifier>String content</identifier>
            <rel>String content</rel>
            <type>String content</type>
        </Site>
        <TurnedOn>1999-05-31T11:20:00</TurnedOn>
    </AlertApi>
    ...
</ArrayOfAlertApi>
```

The following is an example response XML body for https://zvm_ip:port/v1/events and, without the array statement, for https://zvm_ip:port/v1/events/{eventId}.

```xml
<ArrayOfEventApi xmlns="http://schemas.zerto.com/zvm/api">
    <EventApi>
        <Description>String content</Description>
        <EntityType>String content</EntityType>
        <EventCategory>String content</EventCategory>
        <EventCompletedSuccessfully>true</EventCompletedSuccessfully>
        <EventIdentifier>String content</EventIdentifier>
        <EventType>String content</EventType>
    </EventApi>
    ...
</ArrayOfEventApi>
```
<HelpLink>String content</HelpLink>
<Link>
  <href>String content</href>
  <identifier>String content</identifier>
  <rel>String content</rel>
  <type>String content</type>
</Link>
<OccurredOn>1999-05-31T11:20:00</OccurredOn>
<RelatedEntities>
  <Alerts>
    <ResourceLink>
      <href>String content</href>
      <identifier>String content</identifier>
      <rel>String content</rel>
      <type>String content</type>
    </ResourceLink>
  </Alerts>
  <FlrSessions>
    <ResourceLink>
      <href>String content</href>
      <identifier>String content</identifier>
      <rel>String content</rel>
      <type>String content</type>
    </ResourceLink>
  </FlrSessions>
  <Hosts>
    <ResourceLink>
      <href>String content</href>
      <identifier>String content</identifier>
      <rel>String content</rel>
      <type>String content</type>
    </ResourceLink>
  </Hosts>
  <Sites>
    <ResourceLink>
      <href>String content</href>
      <identifier>String content</identifier>
      <rel>String content</rel>
      <type>String content</type>
    </ResourceLink>
  </Sites>
  <Vpgs>
    <ResourceLink>
      <href>String content</href>
      <identifier>String content</identifier>
      <rel>String content</rel>
      <type>String content</type>
    </ResourceLink>
  </Vpgs>
  <Zorgs>
    <ResourceLink>
      <href>String content</href>
      <identifier>String content</identifier>
      <rel>String content</rel>
      <type>String content</type>
    </ResourceLink>
  </Zorgs>
</RelatedEntities>

<SiteIdentifier>String content</SiteIdentifier>
<SiteName>String content</SiteName>
<UserName>String content</UserName>
Local Site API XML Response Format


```xml
<Vpgs>
  <VpgApiLink>
    <Link _x007B_0_x007D_>
      <href>String content</href>
      <rel>String content</rel>
      <type>String content</type>
    </Link>
    <VpgIdentifier>String content</VpgIdentifier>
    <VpgName>String content</VpgName>
  </VpgApiLink>
  ...
</Vpgs>

<ZorgIdentifier>String content</ZorgIdentifier>
<ZorgName>String content</ZorgName>
</EventApi>
...
</ArrayOfEventApi>

The following is an example response XML body for https://zvm_ip:port/v1/localsite.

```xml
<LocalSiteApi xmlns="http://schemas.zerto.com/zvm/api">
  <ContactEmail>String content</ContactEmail>
  <ContactName>String content</ContactName>
  <ContactPhone>String content</ContactPhone>
  <IsReplicationToSelfEnabled>boolean</IsReplicationToSelfEnabled>
  <Link>
    <href>String content</href>
    <identifier>String content</identifier>
    <rel>String content</rel>
    <type>String content</type>
  </Link>
  <Location>String content</Location>
  <SiteIdentifier>String content</SiteIdentifier>
  <SiteName>String content</SiteName>
  <UtcOffsetInMinutes>2147483647</UtcOffsetInMinutes>
  <Version>String content</Version>
</LocalSiteApi>

The following is an example response XML body for https://zvm_ip:port/v1/localsite/pairingstatuses.

```xml
<ArrayOfstring xmlns="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
  <string>String content</string>
  ...
</ArrayOfstring>
Peer Sites API XML Request and Response Formats

The `https://zvm_ip:port/v1/peersites` API has GET and POST methods:
- “Peer Sites API GET Method Request and Response Formats”, below
- “Peer Sites API POST Method Request and Response Formats”, on page 238
- “Peer Sites API DELETE Method Request and Response Formats”, on page 239

Peer Sites API GET Method Request and Response Formats

The following is an example response XML body for `https://zvm_ip:port/v1/peersites` and, without the array statement, for `https://zvm_ip:port/v1/peersites/{siteIdentifier}`.

```
<ArrayOfPeerSiteApi xmlns="http://schemas.zerto.com/zvm/api">
  <ArrayOfstring xmlns="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
    <string>String content</string>
    ...
  </ArrayOfstring>
  <ArrayOfPeerSiteApi xmlns="http://schemas.zerto.com/zvm/api">
    <HostName>String content</HostName>
    <Port>9669</Port>
    ...
  </ArrayOfPeerSiteApi>
</ArrayOfPeerSiteApi>
```

The following is an example response XML body for `https://zvm_ip:port/v1/peersites/pairingstatuses`.

```
<ArrayOfstring xmlns="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
  <string>String content</string>
  ...
</ArrayOfstring>
```

Peer Sites API POST Method Request and Response Formats

The `https://zvm_ip:port/v1/peersites` POST API has both request and response bodies.

Request Body Format

The following is an example request XML body for `https://zvm_ip:port/v1/peersites`.

```
<ArrayOfPeerSiteApi xmlns="http://schemas.zerto.com/zvm/api">
  <HostName>String content</HostName>
  <Port>9669</Port>
</ArrayOfPeerSiteApi>
```
XML Request and Response Formats

Response Body Format

The following is an example response XML body for https://zvm_ip:port/v1/peersites.

```xml
<ArrayOfstring xmlns="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
    <string>String content</string>
    ...
</ArrayOfstring>
<ArrayOfPeerSiteApi xmlns="http://schemas.zerto.com/zvm/api">
    <IsKeepTargetDisks>Boolean</IsKeepTargetDisks>
</ArrayOfPeerSiteApi>
<ArrayOfstring xmlns="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
    <string>String content</string>
    ...
</ArrayOfstring>
<ArrayOfServiceProfileApi xmlns="http://schemas.zerto.com/zvm/api">
    <ServiceProfileApi>
        <Description>String content</Description>
        <History>P428DT10H30M12.3S</History>
        <JournalWarningThresholdInPercent>4294967295</JournalWarningThresholdInPercent>
        <Link>
            <href>String content</href>
            <identifier>String content</identifier>
            <rel>String content</rel>
            <type>String content</type>
        </Link>
        <MaxJournalSizeInPercent>4294967295</MaxJournalSizeInPercent>
        <Rpo>P428DT10H30M12.3S</Rpo>
        <ServiceProfileIdentifier>String content</ServiceProfileIdentifier>
        <ServiceProfileName>String content</ServiceProfileName>
        <TestInterval>P428DT10H30M12.3S</TestInterval>
    </ServiceProfileApi>
    ...
</ArrayOfServiceProfileApi>
```

Peer Sites API DELETE Method Request and Response Formats

The following is an example request XML body for https://zvm_ip:port/v1/peersites/{SiteIdentifier}.

```xml
<ArrayOfPeerSiteApi xmlns="http://schemas.zerto.com/zvm/api">
    <IsKeepTargetDisks>Boolean</IsKeepTargetDisks>
</ArrayOfPeerSiteApi>
```

Response Body Format

The following is an example response XML body for https://zvm_ip:port/v1/peersites/{SiteIdentifier}.

```xml
<ArrayOfstring xmlns="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
    <string>String content</string>
    ...
</ArrayOfstring>
```

Service Profiles API XML Response Formats

The following is an example response XML body for https://zvm_ip:port/v1/serviceprofiles and, without the array statement, for https://zvm_ip:port/v1/serviceprofiles/{siteIdentifier}.

```xml
<ArrayOfServiceProfileApi xmlns="http://schemas.zerto.com/zvm/api">
    <ServiceProfileApi>
        <Description>String content</Description>
        <History>P428DT10H30M12.3S</History>
        <JournalWarningThresholdInPercent>4294967295</JournalWarningThresholdInPercent>
        <Link>
            <href>String content</href>
            <identifier>String content</identifier>
            <rel>String content</rel>
            <type>String content</type>
        </Link>
        <MaxJournalSizeInPercent>4294967295</MaxJournalSizeInPercent>
        <Rpo>P428DT10H30M12.3S</Rpo>
        <ServiceProfileIdentifier>String content</ServiceProfileIdentifier>
        <ServiceProfileName>String content</ServiceProfileName>
        <TestInterval>P428DT10H30M12.3S</TestInterval>
    </ServiceProfileApi>
    ...
</ArrayOfServiceProfileApi>
```
**Session API XML Request Format**

The following is an example request XML body for `https://zvm_ip:port/v1/session/Add`.

```
<AddSessionDataApi xmlns="http://schemas.zerto.com/zvm/api">
  <AuthenticationMethod>String content</AuthenticationMethod>
</AddSessionDataApi>
```

**Tasks API XML Response Format**

The following is an example response XML body for `https://zvm_ip:port/v1/tasks` and, without the array statement, for `https://zvm_ip:port/v1/tasks/{taskIdentifier}`.

```
<ArrayOfTaskApi xmlns="http://schemas.zerto.com/zvm/api">
  <TaskApi>
    <CompleteReason>String content</CompleteReason>
    <Completed>1999-05-31T11:20:00</Completed>
    <InitiatedBy>String content</InitiatedBy>
    <IsCancellable>true</IsCancellable>
    <Link>
      <href>String content</href>
      <identifier>String content</identifier>
      <rel>String content</rel>
      <type>String content</type>
    </Link>
  </TaskApi>
</ArrayOfTaskApi>
```
The following is an example response XML body for https://zvm_ip:port/v1/tasks/types.

```xml
<ArrayOfstring xmlns="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
    <string>String content</string>
    ...
</ArrayOfstring>
```
Virtualization Sites API XML Response Format

Virtualization sites The following is an example response XML body for `https://zvm_ip:port/v1/virtualizationsites` and, without the array statement, for `https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}`.

```xml
<ArrayOfVirtualizationSiteApi xmlns="http://schemas.zerto.com/zvm/api">
  <VirtualizationSiteApi>
    <SiteIdentifier>String content</SiteIdentifier>
    <VirtualizationSiteName>String content</VirtualizationSiteName>
  </VirtualizationSiteApi>
  ...
</ArrayOfVirtualizationSiteApi>
```

Datastore clusters The following is an example response XML body for `https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/datastoreclusters`.

```xml
<ArrayOfDatastoreClusterNativeApi xmlns="http://schemas.zerto.com/zvm/api">
  <DatastoreClusterNativeApi>
    <DatastoreClusterName>String content</DatastoreClusterName>
  </DatastoreClusterNativeApi>
  ...
</ArrayOfDatastoreClusterNativeApi>
```

Datastores The following is an example response XML body for `https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/datastores`.

```xml
<ArrayOfDatastoreNativeApi xmlns="http://schemas.zerto.com/zvm/api">
  <DatastoreNativeApi>
    <DatastoreIdentifier>String content</DatastoreIdentifier>
    <DatastoreName>String content</DatastoreName>
  </DatastoreNativeApi>
  ...
</ArrayOfDatastoreNativeApi>
```

Folders The following is an example response XML body for `https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/folders`.

```xml
<ArrayOfFolderNativeApi xmlns="http://schemas.zerto.com/zvm/api">
  <FolderNativeApi>
    <FolderIdentifier>String content</FolderIdentifier>
    <FolderName>String content</FolderName>
  </FolderNativeApi>
  ...
</ArrayOfFolderNativeApi>
```

Host clusters The following is an example response XML body for `https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/hostclusters`.

```xml
<ArrayOfHostClusterNativeApi xmlns="http://schemas.zerto.com/zvm/api">
  <HostClusterNativeApi>
    <ClusterIdentifier>String content</ClusterIdentifier>
    <VirtualizationClusterName>String content</VirtualizationClusterName>
  </HostClusterNativeApi>
  ...
</ArrayOfHostClusterNativeApi>
```
Zerto Virtual Replication RESTful API Reference Guide - Version 5.5

XML Request and Response Formats

**Hosts** The following is an example response XML body for https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/hosts and, without the array statement, for https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/hosts/{hostIdentifier}.

```xml
<ArrayOfHostNativeApi xmlns="http://schemas.zerto.com/zvm/api">
   <HostNativeApi>
      <HostIdentifier>String content</HostIdentifier>
      <VirtualizationHostName>String content</VirtualizationHostName>
   </HostNativeApi>
   ...
</ArrayOfHostNativeApi>
```

**Networks** The following is an example response XML body for https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/networks.

```xml
<ArrayOfNetworkNativeApi xmlns="http://schemas.zerto.com/zvm/api">
   <NetworkNativeApi>
      <NetworkIdentifier>String content</NetworkIdentifier>
      <VirtualizationNetworkName>String content</VirtualizationNetworkName>
   </NetworkNativeApi>
   ...
</ArrayOfNetworkNativeApi>
```

**VMware only – Org vDCs** The following is an example response XML body for https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/orgvdcs.

```xml
<ArrayOfOrgVdcNativeApi xmlns="http://schemas.zerto.com/zvm/api">
   <OrgVdcNativeApi>
      <Identifier>String content</Identifier>
      <OrgVdcName>String content</OrgVdcName>
   </OrgVdcNativeApi>
   ...
</ArrayOfOrgVdcNativeApi>
```

**VMware only – Resource Pools** The following is an example response XML body for https://zvm_ip:port/v1/virtualizationsites/{siteIdentifier}/resourcepools.

```xml
<ArrayOfResourcePoolNativeApi xmlns="http://schemas.zerto.com/zvm/api">
   <ResourcePoolNativeApi>
      <ResourcePoolIdentifier>String content</ResourcePoolIdentifier>
      <ResourcepoolName>String content</ResourcepoolName>
   </ResourcePoolNativeApi>
   ...
</ArrayOfResourcePoolNativeApi>
```

**VMs API XML Response Format**

The following is an example response XML body for https://zvm_ip:port/v1/vms and, without the array statement, for https://zvm_ip:port/v1/vms/{vmId}.
<ArrayOfVmApi xmlns="http://schemas.zerto.com/zvm/api">
  <VmApi>
    <ActualRPO>2147483647</ActualRPO>
    <Entities>
      <Protected>VCVpg</Protected>
      <Recovery>VCVpg</Recovery>
      <Source>VCVpg</Source>
      <Target>VCVpg</Target>
    </Entities>
    <IOPS>2147483647</IOPS>
    <LastTest>1999-05-31T11:20:00</LastTest>
    <Link>
      <href>String content</href>
      <identifier>String content</identifier>
      <rel>String content</rel>
      <type>String content</type>
    </Link>
    <Link_
      <href>String content</href>
      <rel>String content</rel>
      <type>String content</type>
    </Link_
    <OrganizationName>String content</OrganizationName>
    <Priority>Low</Priority>
    <ProtectedSite>
      <href>String content</href>
      <identifier>String content</identifier>
      <rel>String content</rel>
      <type>String content</type>
    </ProtectedSite>
    <ProvisionedStorageInMB>2147483647</ProvisionedStorageInMB>
    <RecoverySite>
      <href>String content</href>
      <identifier>String content</identifier>
      <rel>String content</rel>
      <type>String content</type>
    </RecoverySite>
    <SourceSite>String content</SourceSite>
    <Status>Initializing</Status>
    <SubStatus>None</SubStatus>
    <TargetSite>String content</TargetSite>
    <ThroughputInMB>1.26743233E+15</ThroughputInMB>
    <UsedStorageInMB>2147483647</UsedStorageInMB>
    <VmIdentifier>String content</VmIdentifier>
    <VmName>String content</VmName>
    <Volumes>
      <VmVolumeApi>
        <VmVolumeIdentifier>String content</VmVolumeIdentifier>
      </VmVolumeApi>
    </Volumes>
    <VpgName>String content</VpgName>
    <EnabledActions>
      <IsFlrEnabled>False</IsFlrEnabled>
    </EnabledActions>
    <VpgIdentifier>String content</VpgIdentifier>
  </VmApi>
...</ArrayOfVmApi>
VPGs API XML Request and Response Formats

The https://zvm_ip:port/v1/vpgs API has GET, POST and DELETE methods:

- “VPGs API GET Method Response Formats”, below
- “VPGs API POST Method Request and Response Formats”, on page 247
- “VPGs API DELETE Method Request and Response Formats”, on page 249

VPGs API GET Method Response Formats

The https://zvm_ip:port/v1/vpgs GET APIs have response bodies.

Response Body Format

The following is an example response XML body for https://zvm_ip:port/v1/vpgs and, without the array statement, for https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}.

```xml
<ArrayOfVpgApi xmlns="http://schemas.zerto.com/zvm/api">
  <VpgApi>
    <ActiveProcessesApi>
      <RunningFailOverTestApi>
        <Stage>String content</Stage>
      </RunningFailOverTestApi>
    </ActiveProcessesApi>
    <ActualRPO>2147483647</ActualRPO>
    <BackupEnabled>true</BackupEnabled>
    <Entities>
      <Protected>VCVpg</Protected>
      <Recovery>VCVpg</Recovery>
      <Source>VCVpg</Source>
      <Target>VCVpg</Target>
    </Entities>
    <IOPS>2147483647</IOPS>
    <LastTest>1999-05-31T11:20:00</LastTest>
    <Link>
      <href>String content</href>
      <identifier>String content</identifier>
      <rel>String content</rel>
      <type>String content</type>
    </Link>
    <Link__x007B_0_x007D_>
      <href>String content</href>
      <rel>String content</rel>
      <type>String content</type>
    </Link__x007B_0_x007D_>
    <OrganizationName>String content</OrganizationName>
    <Priority>Low</Priority>
    <ProgressPercentage>1.26743233E+15</ProgressPercentage>
    <ProtectedSite>
      <href>String content</href>
      <identifier>String content</identifier>
      <rel>String content</rel>
      <type>String content</type>
    </ProtectedSite>
    <ProvisionedStorageInMB>2147483647</ProvisionedStorageInMB>
  </VpgApi>
</ArrayOfVpgApi>
```
The following is an example response XML body for

```
<VpgCheckpointsSummaryApi xmlns="http://schemas.zerto.com/zvm/api">
  <VpgDateCheckpointInfoApi>
    <Count>4294967295</Count>
    <Date>1999-05-31T11:20:00</Date>
  </VpgDateCheckpointInfoApi>
  <VpgDateCheckpointInfoApi>
    <Count>4294967295</Count>
    <Date>1999-05-31T11:20:00</Date>
  </VpgDateCheckpointInfoApi>
</VpgCheckpointsSummaryApi>
```
The following is an example response XML body for

```xml
<ArrayOfCheckpointApi xmlns="http://schemas.zerto.com/zvm/api">
  <CheckpointApi>
    <CheckpointIdentifier>String content</CheckpointIdentifier>
    <Tag>String content</Tag>
    <TimeStamp>1999-05-31T12:00:00</TimeStamp>
    <Vss>true</Vss>
  </CheckpointApi>
  ...
</ArrayOfCheckpointApi>
```

The following is an example response XML body for
https://zvm_ip:port/v1/vpgs/entitytypes,
https://zvm_ip:port/v1/vpgs/failovercommitpolicies,
https://zvm_ip:port/v1/vpgs/failovershutdownpolicies,
https://zvm_ip:port/v1/vpgs/priorities,

```xml
<ArrayOfstring xmlns="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
  <string>String content</string>
  ...
</ArrayOfstring>
```

**VPGs API POST Method Request and Response Formats**

The [https://zvm_ip:port/v1/vpgs](https://zvm_ip:port/v1/vpgs) POST APIs have both request and response bodies.

**Request Body Format**

The following is an example request XML body for

```xml
<VpgCreateDataApi xmlns="http://schemas.zerto.com/zvm/api">
  <DatastoreIdentifier>String content</DatastoreIdentifier>
  <OrgVdcIdentifier>String content</OrgVdcIdentifier>
  <Priority>Low</Priority>
  <ResourcePoolIdentifier>String content</ResourcePoolIdentifier>
  <ServiceProfileIdentifier>String content</ServiceProfileIdentifier>
  <SourceSiteIdentifier>String content</SourceSiteIdentifier>
  <TargetSiteIdentifier>String content</TargetSiteIdentifier>
  <VcVappIdentifier>String content</VcVappIdentifier>
  <VcdVappIdentifier>String content</VcdVappIdentifier>
  <VmsIdentifiers>
    <string xmlns="http://schemas.microsoft.com/2003/10/Serialization/Arrays">String content</string>
    ...
  </VmsIdentifiers>
  <VpgName>String content</VpgName>
  <ZorgIdentifier>String content</ZorgIdentifier>
</VpgCreateDataApi>
```

The following is an example request XML body for
https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/CloneStart.

```xml
<CloneStartDataApi xmlns="http://schemas.zerto.com/zvm/api">
  <CheckpointIdentifier>String content</CheckpointIdentifier>
  <DatastoreIdentifier>String content</DatastoreIdentifier>
</CloneStartDataApi>
```
XML Request and Response Formats

The following is an example request XML body for:

```xml
<FailoverDataApi xmlns="http://schemas.zerto.com/zvm/api">
  <CheckpointIdentifier>String content</CheckpointIdentifier>
  <CommitPolicy>Commit</CommitPolicy>
  <ShutdownPolicy>None</ShutdownPolicy>
  <TimeToWaitBeforeShutdownInSec>2147483647</TimeToWaitBeforeShutdownInSec>
  <IsReverseProtection>true</IsReverseProtection>
</FailoverDataApi>
```

The following is an example request XML body for:

```xml
<FailOverTestStartDataApi xmlns="http://schemas.zerto.com/zvm/api">
  <CheckpointIdentifier>String content</CheckpointIdentifier>
</FailOverTestStartDataApi>
```

The following is an example request XML body for:
https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/FailoverTestStop.

```xml
<StopFailoverTestDataApi xmlns="http://schemas.zerto.com/zvm/api">
  <FailoverTestSuccess>true</FailoverTestSuccess>
  <FailoverTestSummary>String content</FailoverTestSummary>
</StopFailoverTestDataApi>
```

The following is an example request XML body for:

```xml
<FailoverCommitDataApi xmlns="http://schemas.zerto.com/zvm/api">
  <IsReverseProtection>true</IsReverseProtection>
</FailoverCommitDataApi>
```

The request bodies for the other VPGs POST APIs, such as:
https://zvm_ip:port/v1/vpgs/{protectionGroupIdentifier}/FailoverRollback, is empty.

Response Body Format

The following is an example response XML body for:

```xml
<string xmlns="http://schemas.microsoft.com/2003/10/Serialization/">String content</string>
```
VPGs API DELETE Method Request and Response Formats

The https://zvm_ip:port/v1/vpgs DELETE APIs have both request and response bodies.

Request Body Format

The following is an example request XML body for https://zvm_ip:port/v1/vpgs/{VpgIdentifier}.

```xml
<VpgDeleteDataApi xmlns="http://schemas.zerto.com/zvm/api">
  <Force>true</Force>
  <KeepRecoveryVolumes>true</KeepRecoveryVolumes>
</VpgDeleteDataApi>
```

Response Body Format

The following is an example response XML body for https://zvm_ip:port/v1/vpgs/{VpgIdentifier}.

```xml
<string xmlns="http://schemas.microsoft.com/2003/10/Serialization/">String content</string>
```

VPG Settings API XML Request and Response Formats

The https://zvm_ip:port/v1/vpgSettings API has GET, POST, PUT and DELETE methods:

- “GET Method Request and Response Formats”, below
- “POST Method Request and Response Formats”, on page 259
- “PUT Method Request and Response Formats”, on page 265
- “DELETE Method Response Formats”, on page 272

GET Method Request and Response Formats

Request Body Formats

All GET method request bodies are empty.

Response Body Formats

The https://zvm_ip:port/v1/vpgSettings GET APIs have response bodies.

The following is an example response XML body for https://zvm_ip:port/v1/vpgSettings and, without the array statement, for https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}.

```xml
<ArrayOfVpgSettingsApi xmlns="http://schemas.zerto.com/zvm/api">
  <VpgSettingsApi>
    <Backup>
      <RepositoryIdentifier>String content</RepositoryIdentifier>
      <RetentionPeriod>String content</RetentionPeriod>
      <Retry>
        <IntervalInMinutes>2147483647</IntervalInMinutes>
        <Number>2147483647</Number>
        <Retry>true</Retry>
      </Retry>
      <Scheduler>
        <DayOfWeek>String content</DayOfWeek>
        <SchedulerPeriod>String content</SchedulerPeriod>
        <TimeOfDay>String content</TimeOfDay>
      </Scheduler>
    </Backup>
  </VpgSettingsApi>
</ArrayOfVpgSettingsApi>
```
<Basic>
  <JournalHistoryInHours>2147483647</JournalHistoryInHours>
  <Name>String content</Name>
  <Priority>String content</Priority>
  <ProtectedSiteIdentifier>String content</ProtectedSiteIdentifier>
  <RecoverySiteIdentifier>String content</RecoverySiteIdentifier>
  <RpoInseconds>4294967295</RpoInseconds>
  <ServiceProfileIdentifier>String content</ServiceProfileIdentifier>
  <TestIntervalInMinutes>2147483647</TestIntervalInMinutes>
  <UseWanCompression>true</UseWanCompression>
  <ZorgIdentifier>String content</ZorgIdentifier>
</Basic>

<BootGroups>
  <BootGroups>
    <VpgSettingsBootGroupApi>
      <BootDelayInSeconds>4294967295</BootDelayInSeconds>
      <BootGroupIdentifier>String content</BootGroupIdentifier>
      <Name>String content</Name>
    </VpgSettingsBootGroupApi>
    ...</VpgSettingsBootGroupApi>
  </BootGroups>
</BootGroups>

<Journal>
  <DatastoreClusterIdentifier>String content</DatastoreClusterIdentifier>
  <DatastoreIdentifier>String content</DatastoreIdentifier>
  <Limitation>
    <HardLimitInMB>2147483647</HardLimitInMB>
    <HardLimitInPercent>2147483647</HardLimitInPercent>
    <WarningThresholdInMB>2147483647</WarningThresholdInMB>
    <WarningThresholdInPercent>2147483647</WarningThresholdInPercent>
  </Limitation>
</Journal>

<Networks>
  <Failover>
    <Hypervisor>
      <DefaultNetworkIdentifier>String content</DefaultNetworkIdentifier>
    </Hypervisor>
  </Failover>
  <FailoverTest>
    <Hypervisor>
      <DefaultNetworkIdentifier>String content</DefaultNetworkIdentifier>
    </Hypervisor>
  </FailoverTest>
</Networks>

<Recovery>
  <DefaultDatastoreClusterIdentifier>String content</DefaultDatastoreClusterIdentifier>
  <DefaultDatastoreIdentifier>String content</DefaultDatastoreIdentifier>
  <DefaultFolderIdentifier>String content</DefaultFolderIdentifier>
  <DefaultHostClusterIdentifier>String content</DefaultHostClusterIdentifier>
  <DefaultHostIdentifier>String content</DefaultHostIdentifier>
  <ResourcePoolIdentifier>String content</ResourcePoolIdentifier>
</Recovery>
<Scripting>
  <PostBackup>
    <Command>String content</Command>
    <Parameters>String content</Parameters>
    <TimeoutInSeconds>2147483647</TimeoutInSeconds>
  </PostBackup>
  <PostRecovery>
    <Command>String content</Command>
    <Parameters>String content</Parameters>
    <TimeoutInSeconds>2147483647</TimeoutInSeconds>
  </PostRecovery>
  <PreRecovery>
    <Command>String content</Command>
    <Parameters>String content</Parameters>
    <TimeoutInSeconds>2147483647</TimeoutInSeconds>
  </PreRecovery>
</Scripting>

<Vms>
  <VpgSettingsVmApi>
    <BootGroupIdentifier>String content</BootGroupIdentifier>
    <Journal>
      <DatastoreClusterIdentifier>String content</DatastoreClusterIdentifier>
      <DatastoreIdentifier>String content</DatastoreIdentifier>
      <Limitation>
        <HardLimitInMB>4294967295</HardLimitInMB>
        <HardLimitInPercent>4294967295</HardLimitInPercent>
        <WarningThresholdInMB>4294967295</WarningThresholdInMB>
        <WarningThresholdInPercent>4294967295</WarningThresholdInPercent>
      </Limitation>
    </Journal>
    <Nics>
      <VpgSettingsVmNicApi>
        <Failover>
          <Hypervisor>
            <DnsSuffix>String content</DnsSuffix>
            <IpConfig>
              <Gateway>String content</Gateway>
              <IsDhcp>true</IsDhcp>
              <PrimaryDns>String content</PrimaryDns>
              <SecondaryDns>String content</SecondaryDns>
              <StaticIp>String content</StaticIp>
              <SubnetMask>String content</SubnetMask>
            </IpConfig>
          </Hypervisor>
          <NetworkIdentifier>String content</NetworkIdentifier>
          <ShouldReplaceMacAddress>true</ShouldReplaceMacAddress>
        </Failover>
      </VpgSettingsVmNicApi>
    </Nics>
  </VpgSettingsVmApi>
</Vms>
<FailoverTest>
  <Hypervisor>
    <DnsSuffix>String content</DnsSuffix>
    <IpConfig>
      <Gateway>String content</Gateway>
      <IsDhcp>true</IsDhcp>
      <PrimaryDns>String content</PrimaryDns>
      <SecondaryDns>String content</SecondaryDns>
      <StaticIp>String content</StaticIp>
      <SubnetMask>String content</SubnetMask>
    </IpConfig>
    <NetworkIdentifier>String content</NetworkIdentifier>
    <ShouldReplaceMacAddress>true</ShouldReplaceMacAddress>
  </Hypervisor>
</FailoverTest>

<NicIdentifier>String content</NicIdentifier>

</VpgSettingsVmNicApi>

<Failover>
  ...
</Failover>

<FailoverTest>
  ...
</FailoverTest>

<NicIdentifier>String content</NicIdentifier>

</VpgSettingsVmNicApi>

</VpgSettingsVmApi>

<Recovery>
  <DatastoreClusterIdentifier>String content</DatastoreClusterIdentifier>
  <DatastoreIdentifier>String content</DatastoreIdentifier>
  <FolderIdentifier>String content</FolderIdentifier>
  <HostClusterIdentifier>String content</HostClusterIdentifier>
  <HostIdentifier>String content</HostIdentifier>
  <ResourcePoolIdentifier>String content</ResourcePoolIdentifier>
</Recovery>

<VmIdentifier>String content</VmIdentifier>

<Volumes>
  <VpgSettingsVmVolumeApi>
    <Datastore>
      <DatastoreClusterIdentifier>String content</DatastoreClusterIdentifier>
      <DatastoreIdentifier>String content</DatastoreIdentifier>
      <IsThin>true</IsThin>
    </Datastore>
    <ExistingVolume>
      <DatastoreIdentifier>String content</DatastoreIdentifier>
      <ExistedVmIdentifier>String content</ExistedVmIdentifier>
      <Mode>String content</Mode>
      <Path>String content</Path>
    </ExistingVolume>
    <IsSwap>true</IsSwap>
    <VolumeIdentifier>String content</VolumeIdentifier>
  </VpgSettingsVmVolumeApi>
  ...
</Volumes>

</VpgSettingsVmApi>
The following is an example response XML body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/backup.

```xml
<VpgSettingsBackupApi xmlns="http://schemas.zerto.com/zvm/api">
  <RepositoryIdentifier>String content</RepositoryIdentifier>
  <RetentionPeriod>String content</RetentionPeriod>
  <Retry>
    <IntervalInMinutes>2147483647</IntervalInMinutes>
    <Number>2147483647</Number>
    <Retry>true</Retry>
  </Retry>
</VpgSettingsBackupApi>
```

The following is an example response XML body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/backup/dayofweek,
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/backup/retentionperiod, and for

```xml
<repositoryIdentifier>String content</repositoryIdentifier>
<retentionPeriod>String content</retentionPeriod>
<retry>
  <intervalInMinutes>2147483647</intervalInMinutes>
  <number>2147483647</number>
  <retry>true</retry>
</retry>
```

The following is an example response XML body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/basic.

```xml
<VpgSettingsBasicApi xmlns="http://schemas.zerto.com/zvm/api">
  <JournalHistoryInHours>2147483647</JournalHistoryInHours>
  <Name>String content</Name>
  <Priority>String content</Priority>
  <ProtectedSiteIdentifier>String content</ProtectedSiteIdentifier>
  <RecoverySiteIdentifier>String content</RecoverySiteIdentifier>
  <RpoInSeconds>4294967295</RpoInSeconds>
  <ServiceProfileIdentifier>String content</ServiceProfileIdentifier>
  <testIntervalInMinutes>2147483647</testIntervalInMinutes>
  <useWanCompression>true</useWanCompression>
  <zorgIdentifier>String content</zorgIdentifier>
</VpgSettingsBasicApi>
```
The following is an example response XML body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/bootgroup.

```xml
<VpgSettingsBootGroupsApi xmlns="http://schemas.zerto.com/zvm/api">
  <BootGroups>
    <VpgSettingsBootGroupApi>
      <BootDelayInSeconds>4294967295</BootDelayInSeconds>
      <BootGroupIdentifier>String content</BootGroupIdentifier>
      <Name>String content</Name>
    </VpgSettingsBootGroupApi>
    <VpgSettingsBootGroupApi>
      <BootDelayInSeconds>4294967295</BootDelayInSeconds>
      <BootGroupIdentifier>String content</BootGroupIdentifier>
      <Name>String content</Name>
    </VpgSettingsBootGroupApi>
  </BootGroups>
</VpgSettingsBootGroupsApi>
```

The following is an example response XML body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/journal.

```xml
<VpgSettingsJournalApi xmlns="http://schemas.zerto.com/zvm/api">
  <DatastoreIdentifier>String content</DatastoreIdentifier>
  <Limitation>
    <HardLimitInMB>2147483647</HardLimitInMB>
    <HardLimitInPercent>2147483647</HardLimitInPercent>
    <WarningThresholdInMB>2147483647</WarningThresholdInMB>
    <WarningThresholdInPercent>2147483647</WarningThresholdInPercent>
  </Limitation>
</VpgSettingsJournalApi>
```

The following is an example response XML body for

```xml
<VpgSettingsNetworksApi xmlns="http://schemas.zerto.com/zvm/api">
  <Failover>
    <Hypervisor>
      <DefaultNetworkIdentifier>String content</DefaultNetworkIdentifier>
    </Hypervisor>
  </Failover>
  <FailoverTest>
    <Hypervisor>
      <DefaultNetworkIdentifier>String content</DefaultNetworkIdentifier>
    </Hypervisor>
  </FailoverTest>
</VpgSettingsNetworksApi>
```

The networksmapping API is for future use.

The following is an example response XML body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/priority.

```xml
<string xmlns="http://schemas.microsoft.com/2003/10/Serialization/">String content</string>
```
The following is an example response XML body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/recovery.

```xml
<VpgSettingsRecoveryApi xmlns="http://schemas.zerto.com/zvm/api">
  <DefaultDatastoreClusterIdentifier>String content</DefaultDatastoreClusterIdentifier>
  <DefaultDatastoreIdentifier>String content</DefaultDatastoreIdentifier>
  <DefaultFolderIdentifier>String content</DefaultFolderIdentifier>
  <DefaultHostClusterIdentifier>String content</DefaultHostClusterIdentifier>
  <DefaultHostIdentifier>String content</DefaultHostIdentifier>
  <ResourcePoolIdentifier>String content</ResourcePoolIdentifier>
</VpgSettingsRecoveryApi>
```

The following is an example response XML body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/scripting.

```xml
<VpgSettingsScriptsApi xmlns="http://schemas.zerto.com/zvm/api">
  <PostBackup>
    <Command>String content</Command>
    <Parameters>String content</Parameters>
    <TimeoutInSeconds>2147483647</TimeoutInSeconds>
  </PostBackup>
  <PostRecovery>
    <Command>String content</Command>
    <Parameters>String content</Parameters>
    <TimeoutInSeconds>2147483647</TimeoutInSeconds>
  </PostRecovery>
  <PreRecovery>
    <Command>String content</Command>
    <Parameters>String content</Parameters>
    <TimeoutInSeconds>2147483647</TimeoutInSeconds>
  </PreRecovery>
</VpgSettingsScriptsApi>
```

The following is an example response XML body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms and, without the array statement, for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}.

```xml
<ArrayOfVpgSettingsVmApi xmlns="http://schemas.zerto.com/zvm/api">
  <VpgSettingsVmApi>
    <BootGroupIdentifier>String content</BootGroupIdentifier>
    <Journal>
      <DatastoreClusterIdentifier>String content</DatastoreClusterIdentifier>
      <DatastoreIdentifier>String content</DatastoreIdentifier>
      <Limitation>
        <HardLimitInMB>4294967295</HardLimitInMB>
        <HardLimitInPercent>4294967295</HardLimitInPercent>
        <WarningThresholdInMB>4294967295</WarningThresholdInMB>
        <WarningThresholdInPercent>4294967295</WarningThresholdInPercent>
      </Limitation>
    </Journal>
  </VpgSettingsVmApi>
</ArrayOfVpgSettingsVmApi>
```
<Nics>
  <VpgSettingsVmNicApi>
    <Failover>
      <Hypervisor>
        <DnsSuffix>String content</DnsSuffix>
        <IpConfig>
          <Gateway>String content</Gateway>
          <IsDhcp>true</IsDhcp>
          <PrimaryDns>String content</PrimaryDns>
          <SecondaryDns>String content</SecondaryDns>
          <StaticIp>String content</StaticIp>
          <SubnetMask>String content</SubnetMask>
        </IpConfig>
        <NetworkIdentifier>String content</NetworkIdentifier>
        <ShouldReplaceMacAddress>true</ShouldReplaceMacAddress>
      </Hypervisor>
      <FailoverTest>
        <Hypervisor>
          <DnsSuffix>String content</DnsSuffix>
          <IpConfig>
            <Gateway>String content</Gateway>
            <IsDhcp>true</IsDhcp>
            <PrimaryDns>String content</PrimaryDns>
            <SecondaryDns>String content</SecondaryDns>
            <StaticIp>String content</StaticIp>
            <SubnetMask>String content</SubnetMask>
          </IpConfig>
          <NetworkIdentifier>String content</NetworkIdentifier>
          <ShouldReplaceMacAddress>true</ShouldReplaceMacAddress>
        </Hypervisor>
      </FailoverTest>
      <NicIdentifier>String content</NicIdentifier>
    </VpgSettingsVmNicApi>
  </Failover>
  <VpgSettingsVmNicApi>
    <Failover>
      ...
    </Failover>
    <FailoverTest>
      ...
    </FailoverTest>
    <NicIdentifier>String content</NicIdentifier>
  </VpgSettingsVmNicApi>
</Nics>

<Recovery>
  <DatastoreClusterIdentifier>String content</DatastoreClusterIdentifier>
  <DatastoreIdentifier>String content</DatastoreIdentifier>
  <FolderIdentifier>String content</FolderIdentifier>
  <HostClusterIdentifier>String content</HostClusterIdentifier>
  <HostIdentifier>String content</HostIdentifier>
  <ResourcePoolIdentifier>String content</ResourcePoolIdentifier>
</Recovery>
The following is an example response XML body for 
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/nics and, without the 
array statement, for https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/nics/{nicIdentifier}.

```xml
<ArrayOfVpgSettingsVmNicApi xmlns="http://schemas.zerto.com/zvm/api">
  <VpgSettingsVmNicApi>
    <Failover>
      <DnsSuffix>String content</DnsSuffix>
      <IpConfig>
        <Gateway>String content</Gateway>
        <IsDhcp>true</IsDhcp>
        <PrimaryDns>String content</PrimaryDns>
        <SecondaryDns>String content</SecondaryDns>
        <StaticIp>String content</StaticIp>
        <SubnetMask>String content</SubnetMask>
      </IpConfig>
      <NetworkIdentifier>String content</NetworkIdentifier>
      <ShouldReplaceMacAddress>true</ShouldReplaceMacAddress>
    </Failover>
  </VpgSettingsVmNicApi>
</ArrayOfVpgSettingsVmNicApi>
```
<FailoverTest>
  <Hypervisor>
    <DnsSuffix>String content</DnsSuffix>
    <IpConfig>
      <Gateway>String content</Gateway>
      <IsDhcp>true</IsDhcp>
      <PrimaryDns>String content</PrimaryDns>
      <SecondaryDns>String content</SecondaryDns>
      <StaticIp>String content</StaticIp>
      <SubnetMask>String content</SubnetMask>
    </IpConfig>
    <NetworkIdentifier>String content</NetworkIdentifier>
    <ShouldReplaceMacAddress>true</ShouldReplaceMacAddress>
  </Hypervisor>
</FailoverTest>

 InvalidArgumentExceptions

The following is an example response XML body for
https://zvm_ip:port/v1/vpgSettings/\{vpgSettingsIdentifier\}/vms/\{vmIdentifier\}/volumes and, without
the array statement, for https://zvm_ip:port/v1/vpgSettings/volumes/\{volumeId\}.

<ArrayOfVpgSettingsVmVolumeApi xmlns="http://schemas.zerto.com/zvm/api">
  <VpgSettingsVmVolumeApi>
    <Datastore>
      <DatastoreIdentifier>String content</DatastoreIdentifier>
      <IsThin>true</IsThin>
    </Datastore>
  </VpgSettingsVmVolumeApi>
  <ExistingVolume>
    <DatastoreIdentifier>String content</DatastoreIdentifier>
    <ExistedVmIdentifier>String content</ExistedVmIdentifier>
    <Mode>String content</Mode>
    <Path>String content</Path>
  </ExistingVolume>
  <IsSwap>true</IsSwap>
  <VolumeIdentifier>String content</VolumeIdentifier>
</VpgSettingsVmVolumeApi>

<VpgSettingsVmVolumeApi>
  <Datastore>
    <DatastoreClusterIdentifier>String content</DatastoreClusterIdentifier>
    <DatastoreIdentifier>String content</DatastoreIdentifier>
    <IsThin>true</IsThin>
  </Datastore>
</VpgSettingsVmVolumeApi>
```xml
<ExistingVolume>
  <DatastoreIdentifier>String content</DatastoreIdentifier>
  <ExistedVmIdentifier>String content</ExistedVmIdentifier>
  <Mode>String content</Mode>
  <Path>String content</Path>
</ExistingVolume>
(IsSwap>true</IsSwap>
</VpgSettingsVmVolumeApi>
</ArrayOfVpgSettingsVmVolumeApi>
</VpgSettingsApi>
</Backup>
</VpgSettingsApi>
</Basic>
</VpgSettingsApi>
</BootGroups>
</VpgSettingsApi>
</Basic>

POST Method Request and Response Formats

Request Body Formats

The following is an example request XML body for https://zvm_ip:port/v1/vpgSettings.

```xml
<VpgSettingsApi xmlns="http://schemas.zerto.com/zvm/api">
  <Backup>
    <RepositoryIdentifier>String content</RepositoryIdentifier>
    <RetentionPeriod>String content</RetentionPeriod>
    <Retry>
      <IntervalInMinutes>2147483647</IntervalInMinutes>
      <Number>2147483647</Number>
      <Retry>true</Retry>
    </Retry>
    <Scheduler>
      <DayOfWeek>String content</DayOfWeek>
      <SchedulerPeriod>String content</SchedulerPeriod>
      <TimeOfDay>String content</TimeOfDay>
    </Scheduler>
  </Backup>

  <Basic>
    <JournalHistoryInHours>2147483647</JournalHistoryInHours>
    <Name>String content</Name>
    <Priority>String content</Priority>
    <ProtectedSiteIdentifier>String content</ProtectedSiteIdentifier>
    <RecoverySiteIdentifier>String content</RecoverySiteIdentifier>
    <RpoInSeconds>4294967295</RpoInSeconds>
    <ServiceProfileIdentifier>String content</ServiceProfileIdentifier>
    <TestIntervalInMinutes>2147483647</TestIntervalInMinutes>
    <UseWanCompression>true</UseWanCompression>
    <ZorgIdentifier>String content</ZorgIdentifier>
  </Basic>

  <BootGroups>
    <BootGroups>
      <VpgSettingsBootGroupApi>
        <BootDelayInSeconds>4294967295</BootDelayInSeconds>
        <BootGroupIdentifier>String content</BootGroupIdentifier>
        <Name>String content</Name>
      </VpgSettingsBootGroupApi>
    </BootGroups>
    ...
  </VpgSettingsBootGroupApi>
</BootGroups>
</VpgSettingsApi>
```
<Journal>
    <DatastoreClusterIdentifier>String content</DatastoreClusterIdentifier>
    <DatastoreIdentifier>String content</DatastoreIdentifier>
    <Limitation>
        <HardLimitInMB>2147483647</HardLimitInMB>
        <HardLimitInPercent>2147483647</HardLimitInPercent>
        <WarningThresholdInMB>2147483647</WarningThresholdInMB>
        <WarningThresholdInPercent>2147483647</WarningThresholdInPercent>
    </Limitation>
</Journal>

<Networks>
    <Failover>
        <Hypervisor>
            <DefaultNetworkIdentifier>String content</DefaultNetworkIdentifier>
        </Hypervisor>
    </Failover>
    <FailoverTest>
        <Hypervisor>
            <DefaultNetworkIdentifier>String content</DefaultNetworkIdentifier>
        </Hypervisor>
    </FailoverTest>
</Networks>

<Recovery>
    <DefaultDatastoreClusterIdentifier>String content</DefaultDatastoreClusterIdentifier>
    <DefaultDatastoreIdentifier>String content</DefaultDatastoreIdentifier>
    <DefaultFolderIdentifier>String content</DefaultFolderIdentifier>
    <DefaultHostClusterIdentifier>String content</DefaultHostClusterIdentifier>
    <DefaultHostIdentifier>String content</DefaultHostIdentifier>
    <ResourcePoolIdentifier>String content</ResourcePoolIdentifier>
</Recovery>

<Scripting>
    <PostBackup>
        <Command>String content</Command>
        <Parameters>String content</Parameters>
        <TimeoutInSeconds>2147483647</TimeoutInSeconds>
    </PostBackup>
    <PostRecovery>
        <Command>String content</Command>
        <Parameters>String content</Parameters>
        <TimeoutInSeconds>2147483647</TimeoutInSeconds>
    </PostRecovery>
    <PreRecovery>
        <Command>String content</Command>
        <Parameters>String content</Parameters>
        <TimeoutInSeconds>2147483647</TimeoutInSeconds>
    </PreRecovery>
</Scripting>

<Vms>
<VpgSettingsVmApi>
  <BootGroupIdentifier>String content</BootGroupIdentifier>
  <Journal>
    <DatastoreClusterIdentifier>String content</DatastoreClusterIdentifier>
    <DatastoreIdentifier>String content</DatastoreIdentifier>
    <Limitation>
      <HardLimitInMB>4294967295</HardLimitInMB>
      <HardLimitInPercent>4294967295</HardLimitInPercent>
      <WarningThresholdInMB>4294967295</WarningThresholdInMB>
      <WarningThresholdInPercent>4294967295</WarningThresholdInPercent>
    </Limitation>
  </Journal>
</VpgSettingsVmApi>

<Nics>
  <VpgSettingsVmNicApi>
    <Failover>
      <Hypervisor>
        <DnsSuffix>String content</DnsSuffix>
        <IpConfig>
          <Gateway>String content</Gateway>
          <IsDhcp>true</IsDhcp>
          <PrimaryDns>String content</PrimaryDns>
          <SecondaryDns>String content</SecondaryDns>
          <StaticIp>String content</StaticIp>
          <SubnetMask>String content</SubnetMask>
        </IpConfig>
        <NetworkIdentifier>String content</NetworkIdentifier>
        <ShouldReplaceMacAddress>true</ShouldReplaceMacAddress>
      </Hypervisor>
    </Failover>
    <FailoverTest>
      <Hypervisor>
        <DnsSuffix>String content</DnsSuffix>
        <IpConfig>
          <Gateway>String content</Gateway>
          <IsDhcp>true</IsDhcp>
          <PrimaryDns>String content</PrimaryDns>
          <SecondaryDns>String content</SecondaryDns>
          <StaticIp>String content</StaticIp>
          <SubnetMask>String content</SubnetMask>
        </IpConfig>
        <NetworkIdentifier>String content</NetworkIdentifier>
        <ShouldReplaceMacAddress>true</ShouldReplaceMacAddress>
      </Hypervisor>
    </FailoverTest>
    <NicIdentifier>String content</NicIdentifier>
  </VpgSettingsVmNicApi>
  <VpgSettingsVmNicApi>
    <Failover>
      ...<Failover>
    <FailoverTest>
      ...<FailoverTest>
    <NicIdentifier>String content</NicIdentifier>
  </VpgSettingsVmNicApi>
</Nics>
<Recovery>
  <DatastoreClusterIdentifier>String content</DatastoreClusterIdentifier>
  <DatastoreIdentifier>String content</DatastoreIdentifier>
  <FolderIdentifier>String content</FolderIdentifier>
  <HostClusterIdentifier>String content</HostClusterIdentifier>
  <HostIdentifier>String content</HostIdentifier>
  <ResourcePoolIdentifier>String content</ResourcePoolIdentifier>
</Recovery>

<VmIdentifier>String content</VmIdentifier>

<VpgSettingsVmApi>
  <Datastore>
    <DatastoreClusterIdentifier>String content</DatastoreClusterIdentifier>
    <DatastoreIdentifier>String content</DatastoreIdentifier>
    <IsThin>true</IsThin>
  </Datastore>
  <ExistingVolume>
    <DatastoreIdentifier>String content</DatastoreIdentifier>
    <ExistedVmIdentifier>String content</ExistedVmIdentifier>
    <Mode>String content</Mode>
    <Path>String content</Path>
  </ExistingVolume>
  <IsSwap>true</IsSwap>
  <VolumeIdentifier>String content</VolumeIdentifier>
</VpgSettingsVmApi>

</Vms>

<VpgIdentifier>String content</VpgIdentifier>

<VpgSettingsIdentifier>String content</VpgSettingsIdentifier>
</VpgSettingsApi>

The request body for the API https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/commit is empty.

The following is an example request XML body for https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms.

<VpgSettingsVmApi xmlns="http://schemas.zerto.com/zvm/api">
  <BootGroupIdentifier>String content</BootGroupIdentifier>
  <Journal>
    <DatastoreClusterIdentifier>String content</DatastoreClusterIdentifier>
    <DatastoreIdentifier>String content</DatastoreIdentifier>
    <Limitation>
      <HardLimitInMB>4294967295</HardLimitInMB>
      <HardLimitInPercent>4294967295</HardLimitInPercent>
      <WarningThresholdInMB>4294967295</WarningThresholdInMB>
      <WarningThresholdInPercent>4294967295</WarningThresholdInPercent>
    </Limitation>
  </Journal>
</VpgSettingsVmApi>
<Nics>
  <VpgSettingsVmNicApi>
    <Failover>
      <Hypervisor>
        <DnsSuffix>String content</DnsSuffix>
        <IpConfig>
          <Gateway>String content</Gateway>
          <IsDhcp>true</IsDhcp>
          <PrimaryDns>String content</PrimaryDns>
          <SecondaryDns>String content</SecondaryDns>
          <StaticIp>String content</StaticIp>
          <SubnetMask>String content</SubnetMask>
        </IpConfig>
        <NetworkIdentifier>String content</NetworkIdentifier>
      </Hypervisor>
    </Failover>
    <FailoverTest>
      <Hypervisor>
        <DnsSuffix>String content</DnsSuffix>
        <IpConfig>
          <Gateway>String content</Gateway>
          <IsDhcp>true</IsDhcp>
          <PrimaryDns>String content</PrimaryDns>
          <SecondaryDns>String content</SecondaryDns>
          <StaticIp>String content</StaticIp>
          <SubnetMask>String content</SubnetMask>
        </IpConfig>
        <NetworkIdentifier>String content</NetworkIdentifier>
      </Hypervisor>
    </FailoverTest>
    <NicIdentifier>String content</NicIdentifier>
  </VpgSettingsVmNicApi>
</Nics>

<Recovery>
  <DatastoreClusterIdentifier>String content</DatastoreClusterIdentifier>
  <DatastoreIdentifier>String content</DatastoreIdentifier>
  <FolderIdentifier>String content</FolderIdentifier>
  <HostClusterIdentifier>String content</HostClusterIdentifier>
  <HostIdentifier>String content</HostIdentifier>
  <ResourcePoolIdentifier>String content</ResourcePoolIdentifier>
</Recovery>
The following is an example request XML body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/nics.

```xml
<VmIdentifier>String content</VmIdentifier>
<Volumes>
  <VpgSettingsVmVolumeApi>
    <Datastore>
      <DatastoreClusterIdentifier>String content</DatastoreClusterIdentifier>
      <DatastoreIdentifier>String content</DatastoreIdentifier>
      <IsThin>true</IsThin>
    </Datastore>
    <ExistingVolume>
      <ExistedVmIdentifier>String content</ExistedVmIdentifier>
      <Mode>String content</Mode>
      <Path>String content</Path>
    </ExistingVolume>
    <IsSwap>true</IsSwap>
    <VolumeIdentifier>String content</VolumeIdentifier>
  </VpgSettingsVmVolumeApi>
  ...
</Volumes>
</VpgSettingsVmApi>

<Failover xmlns="http://schemas.zerto.com/zvm/api">
  <Failover>
    <Hypervisor>
      <DnsSuffix>String content</DnsSuffix>
      <IpConfig>
        <Gateway>String content</Gateway>
        <IsDhcp>true</IsDhcp>
        <PrimaryDns>String content</PrimaryDns>
        <SecondaryDns>String content</SecondaryDns>
        <StaticIp>String content</StaticIp>
        <SubnetMask>String content</SubnetMask>
      </IpConfig>
      <NetworkIdentifier>String content</NetworkIdentifier>
      <ShouldReplaceMacAddress>true</ShouldReplaceMacAddress>
    </Hypervisor>
  </Failover>
  <FailoverTest>
    <Hypervisor>
      <DnsSuffix>String content</DnsSuffix>
      <IpConfig>
        <Gateway>String content</Gateway>
        <IsDhcp>true</IsDhcp>
        <PrimaryDns>String content</PrimaryDns>
        <SecondaryDns>String content</SecondaryDns>
        <StaticIp>String content</StaticIp>
        <SubnetMask>String content</SubnetMask>
      </IpConfig>
      <NetworkIdentifier>String content</NetworkIdentifier>
      <ShouldReplaceMacAddress>true</ShouldReplaceMacAddress>
    </Hypervisor>
  </FailoverTest>
</Failover>
```

<VpgSettingsVmNicApi>
  <NicIdentifier>String content</NicIdentifier>
</VpgSettingsVmNicApi>
Response Body Formats

The following is an example response XML body for https://zvm_ip:port/v1/vpgSettings:

```xml
<string xmlns="http://schemas.microsoft.com/2003/10/Serialization/">String content</string>
```

The following is an example response XML body for https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/commit:

```xml
<string xmlns="http://schemas.microsoft.com/2003/10/Serialization/">String content</string>
```

The response XML bodies for the following APIs are empty:

https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/nics

PUT Method Request and Response Formats

Request Body Formats

The following is an example request XML body for

https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}:

```xml
<VpgSettingsApi xmlns="http://schemas.zerto.com/zvm/api">
  <Backup>
    <RepositoryIdentifier>String content</RepositoryIdentifier>
    <RetentionPeriod>String content</RetentionPeriod>
    <Retry>
      <IntervalInMinutes>2147483647</IntervalInMinutes>
      <Number>2147483647</Number>
      <Retry>true</Retry>
    </Retry>
    <Scheduler>
      <DayOfWeek>String content</DayOfWeek>
      <SchedulerPeriod>String content</SchedulerPeriod>
      <TimeOfDay>String content</TimeOfDay>
    </Scheduler>
  </Backup>
  <Basic>
    <JournalHistoryInHours>2147483647</JournalHistoryInHours>
    <Name>String content</Name>
    <Priority>String content</Priority>
    <ProtectedSiteIdentifier>String content</ProtectedSiteIdentifier>
    <RecoverySiteIdentifier>String content</RecoverySiteIdentifier>
    <RpoInSeconds>4294967295</RpoInSeconds>
    <ServiceProfileIdentifier>String content</ServiceProfileIdentifier>
    <TestIntervalInMinutes>2147483647</TestIntervalInMinutes>
    <UseWanCompression>true</UseWanCompression>
    <ZorgIdentifier>String content</ZorgIdentifier>
  </Basic>
</VpgSettingsApi>
```
<Vms>
  <VpgSettingsVmApi>
    <BootGroupIdentifier>String content</BootGroupIdentifier>
    <Journal>
      <DatastoreClusterIdentifier>String content</DatastoreClusterIdentifier>
      <DatastoreIdentifier>String content</DatastoreIdentifier>
      <Limitation>
        <HardLimitInMB>4294967295</HardLimitInMB>
        <HardLimitInPercent>4294967295</HardLimitInPercent>
        <WarningThresholdInMB>4294967295</WarningThresholdInMB>
        <WarningThresholdInPercent>4294967295</WarningThresholdInPercent>
      </Limitation>
    </Journal>
  </VpgSettingsVmApi>
  <Nics>
    <VpgSettingsVmNicApi>
      <Failover>
        <Hypervisor>
          <DnsSuffix>String content</DnsSuffix>
          <IpConfig>
            <Gateway>String content</Gateway>
            <IsDhcp>true</IsDhcp>
            <PrimaryDns>String content</PrimaryDns>
            <SecondaryDns>String content</SecondaryDns>
            <StaticIp>String content</StaticIp>
            <SubnetMask>String content</SubnetMask>
          </IpConfig>
          <NetworkIdentifier>String content</NetworkIdentifier>
          <ShouldReplaceMacAddress>true</ShouldReplaceMacAddress>
        </Hypervisor>
        <FailoverTest>
          <Hypervisor>
            <DnsSuffix>String content</DnsSuffix>
            <IpConfig>
              <Gateway>String content</Gateway>
              <IsDhcp>true</IsDhcp>
              <PrimaryDns>String content</PrimaryDns>
              <SecondaryDns>String content</SecondaryDns>
              <StaticIp>String content</StaticIp>
              <SubnetMask>String content</SubnetMask>
            </IpConfig>
            <NetworkIdentifier>String content</NetworkIdentifier>
            <ShouldReplaceMacAddress>true</ShouldReplaceMacAddress>
          </Hypervisor>
          <FailoverTest>
            <NicIdentifier>String content</NicIdentifier>
          </FailoverTest>
        </FailoverTest>
      </VpgSettingsVmNicApi>
    </Nics>
  </VpgSettingsVmApi>
</Vms>
The following is an example request XML body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/backup.

```xml
<VpgSettingsBackupApi xmlns="http://schemas.zerto.com/zvm/api">
  <RepositoryIdentifier>String content</RepositoryIdentifier>
  <RetentionPeriod>String content</RetentionPeriod>
  <Retry>
    <IntervalInMinutes>2147483647</IntervalInMinutes>
    <Number>2147483647</Number>
    <Retry>true</Retry>
  </Retry>
  <Scheduler>
    <DayOfWeek>String content</DayOfWeek>
    <SchedulerPeriod>String content</SchedulerPeriod>
    <TimeOfDay>String content</TimeOfDay>
  </Scheduler>
</VpgSettingsBackupApi>
```
The following is an example request XML body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/basic.

```xml
<VpgSettingsBasicApi xmlns="http://schemas.zerto.com/zvm/api">
  <JournalHistoryInHours>2147483647</JournalHistoryInHours>
  <Name>String content</Name>
  <Priority>String content</Priority>
  <ProtectedSiteIdentifier>String content</ProtectedSiteIdentifier>
  <RecoverySiteIdentifier>String content</RecoverySiteIdentifier>
  <RpoInSeconds>4294967295</RpoInSeconds>
  <ServiceProfileIdentifier>String content</ServiceProfileIdentifier>
  <TestIntervalInMinutes>2147483647</TestIntervalInMinutes>
  <UseWanCompression>true</UseWanCompression>
  <ZorgIdentifier>String content</ZorgIdentifier>
</VpgSettingsBasicApi>
```

The following is an example request XML body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/bootgroup.

```xml
<VpgSettingsBootGroupsApi xmlns="http://schemas.zerto.com/zvm/api">
  <BootGroups>
    <VpgSettingsBootGroupApi>
      <BootDelayInSeconds>4294967295</BootDelayInSeconds>
      <BootGroupIdentifier>String content</BootGroupIdentifier>
      <Name>String content</Name>
    </VpgSettingsBootGroupApi>
    <VpgSettingsBootGroupApi>
      <BootDelayInSeconds>4294967295</BootDelayInSeconds>
      <BootGroupIdentifier>String content</BootGroupIdentifier>
      <Name>String content</Name>
    </VpgSettingsBootGroupApi>
  </BootGroups>
</VpgSettingsBootGroupsApi>
```

The following is an example request XML body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/journal.

```xml
<VpgSettingsJournalApi xmlns="http://schemas.zerto.com/zvm/api">
  <DatastoreClusterIdentifier>String content</DatastoreClusterIdentifier>
  <DatastoreIdentifier>String content</DatastoreIdentifier>
  <Limitation>
    <HardLimitInMB>2147483647</HardLimitInMB>
    <HardLimitInPercent>2147483647</HardLimitInPercent>
    <WarningThresholdInMB>2147483647</WarningThresholdInMB>
    <WarningThresholdInPercent>2147483647</WarningThresholdInPercent>
  </Limitation>
</VpgSettingsJournalApi>
```

The following is an example request XML body for

```xml
<VpgSettingsNetworksApi xmlns="http://schemas.zerto.com/zvm/api">
  <Failover>
    <Hypervisor>
      <DefaultNetworkIdentifier>String content</DefaultNetworkIdentifier>
    </Hypervisor>
  </Failover>
  <FailoverTest>
    <Hypervisor>
      <DefaultNetworkIdentifier>String content</DefaultNetworkIdentifier>
    </Hypervisor>
  </FailoverTest>
</VpgSettingsNetworksApi>
```
The following is an example request XML body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/recovery.

```xml
<VpgSettingsRecoveryApi xmlns="http://schemas.zerto.com/zvm/api">
  <DefaultDatastoreClusterIdentifier>String content</DefaultDatastoreClusterIdentifier>
  <DefaultDatastoreIdentifier>String content</DefaultDatastoreIdentifier>
  <DefaultFolderIdentifier>String content</DefaultFolderIdentifier>
  <DefaultHostClusterIdentifier>String content</DefaultHostClusterIdentifier>
  <DefaultHostIdentifier>String content</DefaultHostIdentifier>
  <ResourcePoolIdentifier>String content</ResourcePoolIdentifier>
</VpgSettingsRecoveryApi>
```

The following is an example request XML body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/scripting.

```xml
<VpgSettingsScriptsApi xmlns="http://schemas.zerto.com/zvm/api">
  <PostBackup>
    <Command>String content</Command>
    <Parameters>String content</Parameters>
    <TimeoutInSeconds>2147483647</TimeoutInSeconds>
  </PostBackup>
  <PostRecovery>
    <Command>String content</Command>
    <Parameters>String content</Parameters>
    <TimeoutInSeconds>2147483647</TimeoutInSeconds>
  </PostRecovery>
  <PreRecovery>
    <Command>String content</Command>
    <Parameters>String content</Parameters>
    <TimeoutInSeconds>2147483647</TimeoutInSeconds>
  </PreRecovery>
</VpgSettingsScriptsApi>
```

The following is an example request XML body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}.

```xml
<VpgSettingsVmApi xmlns="http://schemas.zerto.com/zvm/api">
  <BootGroupIdentifier>String content</BootGroupIdentifier>
  <Journal>
    <DatastoreClusterIdentifier>String content</DatastoreClusterIdentifier>
    <DatastoreIdentifier>String content</DatastoreIdentifier>
    <Limitation>
      <HardLimitInMB>4294967295</HardLimitInMB>
      <HardLimitInPercent>4294967295</HardLimitInPercent>
      <WarningThresholdInMB>4294967295</WarningThresholdInMB>
      <WarningThresholdInPercent>4294967295</WarningThresholdInPercent>
    </Limitation>
  </Journal>
</VpgSettingsVmApi>
```
<Nics>
  <VpgSettingsVmNicApi>
    <Failover>
      <Hypervisor>
        <DnsSuffix>String content</DnsSuffix>
        <IpConfig>
          <Gateway>String content</Gateway>
          <IsDhcp>true</IsDhcp>
          <PrimaryDns>String content</PrimaryDns>
          <SecondaryDns>String content</SecondaryDns>
          <StaticIp>String content</StaticIp>
          <SubnetMask>String content</SubnetMask>
        </IpConfig>
        <NetworkIdentifier>String content</NetworkIdentifier>
        <ShouldReplaceMacAddress>true</ShouldReplaceMacAddress>
      </Hypervisor>
    </Failover>

    <FailoverTest>
      <Hypervisor>
        <DnsSuffix>String content</DnsSuffix>
        <IpConfig>
          <Gateway>String content</Gateway>
          <IsDhcp>true</IsDhcp>
          <PrimaryDns>String content</PrimaryDns>
          <SecondaryDns>String content</SecondaryDns>
          <StaticIp>String content</StaticIp>
          <SubnetMask>String content</SubnetMask>
        </IpConfig>
        <NetworkIdentifier>String content</NetworkIdentifier>
        <ShouldReplaceMacAddress>true</ShouldReplaceMacAddress>
      </Hypervisor>
    </FailoverTest>

    <NicIdentifier>String content</NicIdentifier>
  </VpgSettingsVmNicApi>
</Nics>

<Recovery>
  <DatastoreClusterIdentifier>String content</DatastoreClusterIdentifier>
  <DatastoreIdentifier>String content</DatastoreIdentifier>
  <FolderIdentifier>String content</FolderIdentifier>
  <HostClusterIdentifier>String content</HostClusterIdentifier>
  <HostIdentifier>String content</HostIdentifier>
  <ResourcePoolIdentifier>String content</ResourcePoolIdentifier>
</Recovery>
XML Request and Response Formats

The following is an example request XML body for
https://zvm_ip:port/v1/vpgSettings/{vpgSettingsIdentifier}/vms/{vmIdentifier}/volumes/{volumeId}:

```xml
<VmIdentifier>String content</VmIdentifier>
<Volumes>
  <VpgSettingsVmVolumeApi>
    <Datastore>
      <DatastoreClusterIdentifier>String content</DatastoreClusterIdentifier>
      <DatastoreIdentifier>String content</DatastoreIdentifier>
      <IsThin>true</IsThin>
    </Datastore>
    <ExistingVolume>
      <DatastoreIdentifier>String content</DatastoreIdentifier>
      <ExistedVmIdentifier>String content</ExistedVmIdentifier>
      <Mode>String content</Mode>
      <Path>String content</Path>
    </ExistingVolume>
    <IsSwap>true</IsSwap>
    <VolumeIdentifier>String content</VolumeIdentifier>
  </VpgSettingsVmVolumeApi>
  ...  
</Volumes>
</VpgSettingsVmApi>
```

Response Body Formats

All PUT method response bodies are empty.

DELETE Method Response Formats

All DELETE method request and response bodies are empty.

VRAs API XML Request and Response Formats

The https://zvm_ip:port/v1/vras API has GET, POST, PUT and DELETE methods:

- “VRAs API GET Method Response Formats”, below
- “VRAs API POST Method Request and Response Formats”, on page 274
- “VRAs API PUT Method Request and Response Formats”, on page 275
- “VRAs API DELETE Method Response Formats”, on page 275
**VRAs API GET Method Response Formats**

The https://zvm_ip:port/v1/vras GET APIs have response bodies.

The following is an example response XML body for https://zvm_ip:port/v1/vras and, without the array statement, for https://zvm_ip:port/v1/vras/{vraIdentifier}.

```
<ArrayOfVraApi xmlns="http://schemas.zerto.com/zvm/api">
  <VraApi>
    <DatastoreClusterName>String content</DatastoreClusterName>
    <DatastoreIdentifier>String content</DatastoreIdentifier>
    <DatastoreName>String content</DatastoreName>
    <HostIdentifier>String content</HostIdentifier>
    <HostVersion>String content</HostVersion>
    <IpAddress>String content</IpAddress>
    <Link>
      <href>String content</href>
      <identifier>String content</identifier>
      <rel>String content</rel>
      <type>String content</type>
    </Link>
    <Link__x007B_0_x007D_>
      <href>String content</href>
      <rel>String content</rel>
      <type>String content</type>
    </Link__x007B_0_x007D_>
    <MemoryInGB>2147483647</MemoryInGB>
    <NetworkIdentifier>String content</NetworkIdentifier>
    <NetworkName>String content</NetworkName>
    <Progress>2147483647</Progress>
    <ProtectedCounters>
      <Vms>2147483647</Vms>
      <Volumes>2147483647</Volumes>
      <Vpgs>2147483647</Vpgs>
    </ProtectedCounters>
    <RecoveryCounters>
      <Vms>2147483647</Vms>
      <Volumes>2147483647</Volumes>
      <Vpgs>2147483647</Vpgs>
    </RecoveryCounters>
    <SelfProtectedVpgs>2147483647</SelfProtectedVpgs>
    <Status>Installed</Status>
    <VraGroup>String content</VraGroup>
    <VraIdentifier>18446744073709551615</VraIdentifier>
    <VraName>String content</VraName>
    <VraNetworkDataApi>
      <DefaultGateway>String content</DefaultGateway>
      <SubnetMask>String content</SubnetMask>
      <VraIPAddress>String content</VraIPAddress>
      <VraIPConfigurationTypeApi>String content</VraIPConfigurationTypeApi>
    </VraNetworkDataApi>
    <VraVersion>String content</VraVersion>
  </VraApi>
  ...
</ArrayOfVraApi>
```
The following is an example response XML body for https://zvm_ip:port/v1/vras/ipconfigurationtypes and for https://zvm_ip:port/v1/vras/statuses.

```xml
<ArrayOfstring xmlns="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
  <string>String content</string>
  ...
</ArrayOfstring>
```

**VRAs API POST Method Request and Response Formats**

The https://zvm_ip:port/v1/v ras POST APIs have both request and response bodies.

**Request Body Format**

The following is an example request XML body for https://zvm_ip:port/v1/vras.

```xml
<VraCreateDataApi xmlns="http://schemas.zerto.com/zvm/api">
  <DatastoreIdentifier>String content</DatastoreIdentifier>
  <GroupName>String content</GroupName>
  <HostIdentifier>String content</HostIdentifier>
  <HostRootPassword>String content</HostRootPassword>
  <MemoryInGb>2147483647</MemoryInGb>
  <NetworkIdentifier>String content</NetworkIdentifier>
  <UsePublicKeyInsteadOfCredentials>true</UsePublicKeyInsteadOfCredentials>
  <VraNetworkDataApi>
    <DefaultGateway>String content</DefaultGateway>
    <SubnetMask>String content</SubnetMask>
    <VraIPAddress>String content</VraIPAddress>
    <VraIPConfigurationTypeApi>String content</VraIPConfigurationTypeApi>
  </VraNetworkDataApi>
</VraCreateDataApi>
```

The following is an example request XML body for https://zvm_ip:port/v1/vras/upgrade.

```xml
< UpgradeVrasDataApi xmlns="http://schemas.zerto.com/zvm/api">
  <VraIdentifiers>
    <string>String content</string>
  ...
  <string>String content</string>
  </VraIdentifiers>
</UpgradeVrasDataApi>
```

The request body for https://zvm_ip:port/v1/vras/{vraIdentifier}/upgrade is empty.

**Response Body Format**

The following is an example response XML body for:

- https://zvm_ip:port/v1/vras
- https://zvm_ip:port/v1/vras/upgrade

```xml
<string xmlns="http://schemas.microsoft.com/2003/10/Serialization/">String content</string>
```
VRAs API PUT Method Request and Response Formats

The https://zvm_ip:port/v1/vras PUT APIs have both request and response bodies.

Request Body Format

The following is an example request XML body for https://zvm_ip:port/v1/vras/{vraIdentifier}:

```xml
<VraEditDataApi xmlns="http://schemas.zerto.com/zvm/api">
  <GroupName>String content</GroupName>
  <HostRootPassword>String content</HostRootPassword>
  <UsePublicKeyInsteadOfCredentials>true</UsePublicKeyInsteadOfCredentials>
  <VraNetworkDataApi>
    <DefaultGateway>String content</DefaultGateway>
    <SubnetMask>String content</SubnetMask>
    <VraIPAddress>String content</VraIPAddress>
    <VraIPConfigurationTypeApi>String content</VraIPConfigurationTypeApi>
  </VraNetworkDataApi>
</VraEditDataApi>
```

Response Body Format

The following is an example response XML body for https://zvm_ip:port/v1/vras/{vraIdentifier}:

```xml
<string xmlns="http://schemas.microsoft.com/2003/10/Serialization/">String content</string>
```

VRAs API DELETE Method Response Formats

The https://zvm_ip:port/v1/vras/{vraIdentifier} DELETE API has a response body.

Response Body Format

The following is an example response XML body for https://zvm_ip:port/v1/vras/{vraIdentifier}:

```xml
<string xmlns="http://schemas.microsoft.com/2003/10/Serialization/">String content</string>
```
Zerto Virtual Replication RESTful API Reference Guide - Version 5.5
XML Request and Response Formats

ZORG API XML Response Format

The following is an example response XML body for https://zvm_ip:port/v1/zorgs and, without the array statement, for https://zvm_ip:port/v1/zorgs/{zorgIdentifier}.

```xml
<ArrayOfZorgApi xmlns="http://schemas.zerto.com/zvm/api">
  <ZorgApi>
    <Link>
      <href>String content</href>
      <identifier>String content</identifier>
      <rel>String content</rel>
      <type>String content</type>
    </Link>
    <ZorgIdentifier>String content</ZorgIdentifier>
    <ZorgName>String content</ZorgName>
  </ZorgApi>
  ...
</ArrayOfZorgApi>
```