

# Administering View Cloud Pod Architecture

VMware Horizon 6.0

This document supports the version of each product listed and supports all subsequent versions until the document is replaced by a new edition. To check for more recent editions of this document, see <http://www.vmware.com/support/pubs>.

EN-001464-00

**vmware**<sup>®</sup>

You can find the most up-to-date technical documentation on the VMware Web site at:

<http://www.vmware.com/support/>

The VMware Web site also provides the latest product updates.

If you have comments about this documentation, submit your feedback to:

[docfeedback@vmware.com](mailto:docfeedback@vmware.com)

Copyright © 2014 VMware, Inc. All rights reserved. [Copyright and trademark information.](#)

**VMware, Inc.**  
3401 Hillview Ave.  
Palo Alto, CA 94304  
[www.vmware.com](http://www.vmware.com)

# Contents

Administering View Cloud Pod Architecture	5
<b>1 Introduction to Cloud Pod Architecture</b>	<b>7</b>
Understanding Cloud Pod Architecture	7
Configuring and Managing a Cloud Pod Architecture Environment	8
Cloud Pod Architecture Limitations	8
<b>2 Designing a Cloud Pod Architecture Topology</b>	<b>9</b>
Creating Cloud Pod Architecture Sites	9
Entitling Users and Groups in a Pod Federation	10
Finding and Allocating Desktops in a Pod Federation	10
Global Entitlement Example	12
Cloud Pod Architecture Topology Limits	12
Cloud Pod Architecture Port Requirements	13
Security Considerations for Cloud Pod Architecture Topologies	13
<b>3 Setting Up a Cloud Pod Architecture Environment</b>	<b>15</b>
Initialize the Cloud Pod Architecture Feature	15
Join Pods to the Pod Federation	16
Find and Change a Pod Name	17
Create and Configure a Global Entitlement	18
Create and Configure a Site	21
Assign a Home Site to a User or Group	22
Test a Cloud Pod Architecture Configuration	23
Sample Scenario: Setting Up a Basic Cloud Pod Architecture Environment	24
<b>4 Managing a Cloud Pod Architecture Environment</b>	<b>29</b>
View a Cloud Pod Architecture Configuration	29
View Pod Federation Health in View Administrator	31
View Desktop Sessions in a Pod Federation	31
Determine the Effective Home Site for a User	32
Add a Pod to a Site	33
Remove a Pod From a Pod Federation	33
Modifying Global Entitlements	34
Remove a Home Site Association	40
Disable the Cloud Pod Architecture Feature	41
<b>5 Invutil Command Reference</b>	<b>43</b>
Invutil Command Usage	43
Initializing the Cloud Pod Architecture Feature	46
Disabling the Cloud Pod Architecture Feature	47

Managing Pod Federations	47
Managing Sites	49
Managing Global Entitlements	52
Managing Home Sites	58
Viewing a Cloud Pod Architecture Configuration	60
Managing SSL Certificates	65

Index	67
-------	----

# Administering View Cloud Pod Architecture

---

*Administering View Cloud Pod Architecture* describes how to configure and administer a Cloud Pod Architecture environment in VMware Horizon™ with View™, including how to plan a Cloud Pod Architecture topology and set up, monitor, and maintain a Cloud Pod Architecture configuration.

## Intended Audience

This information is intended for anyone who wants to set up and maintain a Cloud Pod Architecture environment. The information is written for experienced Windows or Linux system administrators who are familiar with virtual machine technology and datacenter operations.

## VMware Technical Publications Glossary

VMware Technical Publications provides a glossary of terms that might be unfamiliar to you. For definitions of terms as they are used in VMware technical documentation, go to <http://www.vmware.com/support/pubs>.



# Introduction to Cloud Pod Architecture

# 1

The Cloud Pod Architecture feature uses standard View components to provide cross-datacenter administration, global and flexible user-to-desktop mapping, high availability desktops, and disaster recovery capabilities.

This chapter includes the following topics:

- [“Understanding Cloud Pod Architecture,”](#) on page 7
- [“Configuring and Managing a Cloud Pod Architecture Environment,”](#) on page 8
- [“Cloud Pod Architecture Limitations,”](#) on page 8

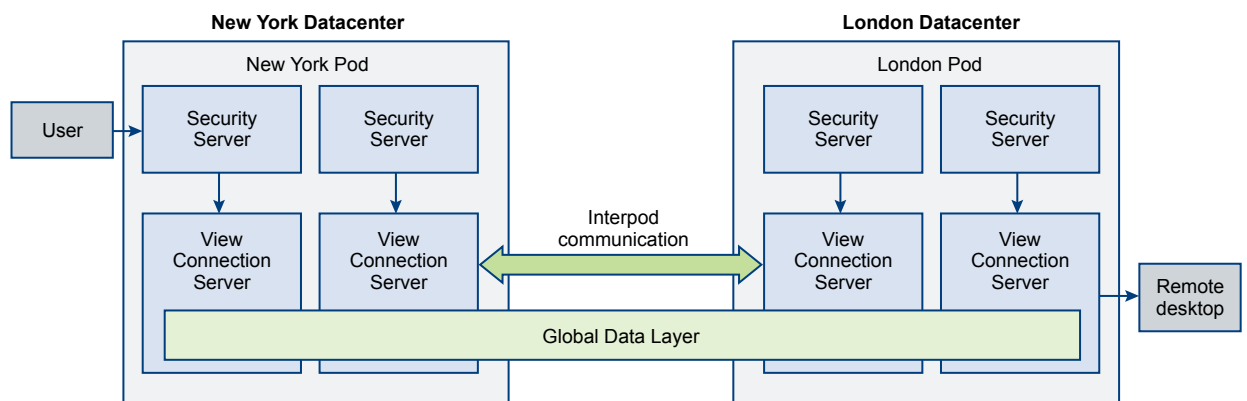
## Understanding Cloud Pod Architecture

With the Cloud Pod Architecture feature, you can link together multiple View pods to provide a single large desktop brokering and management environment.

A View pod consists of a set of View Connection Server instances, shared storage, a database server, and the vSphere and network infrastructures required to host desktop virtual machines. In a traditional View implementation, you manage each pod independently. With the Cloud Pod Architecture feature, you can join together multiple pods to form a single View implementation called a pod federation.

A pod federation can span multiple sites and datacenters and simultaneously simplify the administration effort required to manage a large-scale View deployment.

The following diagram is an example of a basic Cloud Pod Architecture topology.



In the example topology, two previously standalone View pods in different datacenters are joined together to form a single pod federation. An end user in this environment can connect to a View Connection Server instance in the New York datacenter and receive a session on a desktop in the London datacenter.

## Sharing Key Data in the Global Data Layer

View Connection Server instances in a pod federation use the Global Data Layer to share key data. Shared data includes information about the pod federation topology, user and group entitlements, policies, and other Cloud Pod Architecture configuration information.

In a Cloud Pod Architecture environment, shared data is replicated on every View Connection Server instance in a pod federation. Entitlement and topology configuration information stored in the Global Data Layer determines where and how desktops are allocated across the pod federation.

View sets up the Global Data Layer on each View Connection Server instance in a pod federation when you initialize the Cloud Pod Architecture feature.

## Sending Messages Between Pods

View Connection Server instances communicate in a Cloud Pod Architecture environment by using an interpod communication protocol called the View InterPod API (VIPA).

View Connection Server instances use the VIPA interpod communication channel to launch new desktops, find existing desktops, and share health status data and other information. View configures the VIPA interpod communications channel when you initialize the Cloud Pod Architecture feature.

## Configuring and Managing a Cloud Pod Architecture Environment

You use the `lmvutil` command line tool to configure and manage a Cloud Pod Architecture environment. `lmvutil` is installed as part of the View installation. You can use View Administrator to view pod health and desktop session information.

## Cloud Pod Architecture Limitations

The Cloud Pod Architecture feature has certain limitations.

- This release does not support using the HTML Access feature. With HTML Access, end users can use a Web browser to connect to remote desktops and are not required to install any client software on their local systems.
- This release does not support using remote Windows-based applications hosted on a Microsoft RDS host.



# Designing a Cloud Pod Architecture Topology

---

# 2

Before you begin to configure the Cloud Pod Architecture feature, you must make decisions about your Cloud Pod Architecture topology. Cloud Pod Architecture topologies can vary, depending on your goals, the needs of your users, and your existing View implementation. If you are joining existing View pods to a pod federation, your Cloud Pod Architecture topology is typically based on your existing network topology.

This chapter includes the following topics:

- [“Creating Cloud Pod Architecture Sites,”](#) on page 9
- [“Entitling Users and Groups in a Pod Federation,”](#) on page 10
- [“Finding and Allocating Desktops in a Pod Federation,”](#) on page 10
- [“Global Entitlement Example,”](#) on page 12
- [“Cloud Pod Architecture Topology Limits,”](#) on page 12
- [“Cloud Pod Architecture Port Requirements,”](#) on page 13
- [“Security Considerations for Cloud Pod Architecture Topologies,”](#) on page 13

## Creating Cloud Pod Architecture Sites

In a Cloud Pod Architecture environment, a site is a collection of well-connected pods in the same physical location, typically in a single datacenter. The Cloud Pod Architecture feature treats pods in the same site equally.

When you initialize the Cloud Pod Architecture feature, it places all pods into a default site called Default First Site. If you have a large implementation, you might want to create additional sites and add pods to those sites.

The Cloud Pod Architecture feature assumes that pods within the same site are on the same LAN, and that pods in different sites are on different LANs. Because WAN-connected pods have slower network performance, the Cloud Pod Architecture feature gives preference to desktops that are in the local pod or site when it allocates desktops to users.

Sites can be a useful part of a disaster recovery solution. For example, you can assign pods in different datacenters to different sites and then entitle users and groups to desktop pools that span those sites. If a datacenter in one site becomes unavailable, you can use desktops from the available site to satisfy user desktop requests.

For information about creating sites, see [“Create and Configure a Site,”](#) on page 21.

## Entitling Users and Groups in a Pod Federation

In a traditional View environment, you use View Administrator to create entitlements. These local entitlements entitle users and groups to a specific desktop pool on a View Connection Server instance.

In a Cloud Pod Architecture environment, you create global entitlements to entitle users or groups to multiple desktops across multiple pods in the pod federation. When you use global entitlements, you do not need to configure and manage local entitlements. Global entitlements simplify administration, even in a pod federation that contains a single pod.

View stores global entitlements in the Global Data Layer. Because global entitlements are shared data, global entitlement information is available on all View Connection Server instances in the pod federation.

---

**NOTE** As a best practice, you should not configure local and global entitlements for the same desktop pool. If you use both types of entitlements for the same desktop pool, the same desktop might appear as both a local and a global entitlement in the list of desktops that Horizon Client shows to an end user.

---

Each global entitlement contains a list of member users or groups, a list of the desktop pools that can provide desktops for entitled users, and a scope policy. The desktop pools in a global entitlement can be either floating or dedicated pools. You specify whether a global entitlement is floating or dedicated during global entitlement creation.

A global entitlement's scope policy specifies where View looks for desktops when it allocates desktops to users in the global entitlement. It also determines whether View looks for desktops in any pod in the pod federation, in pods that reside in the same site, or only in the pod to which the user is connected.

## Finding and Allocating Desktops in a Pod Federation

View Connection Server instances in a Cloud Pod Architecture environment use shared global entitlement and topology configuration information from the Global Data Layer to determine where to search for and how to allocate desktops across the pod federation.

When a user requests a desktop from a global entitlement, the Cloud Pod Architecture feature searches for an available desktop in the pools that are associated with that global entitlement. By default, the Cloud Pod Architecture feature gives preference to desktops in the local pod, the local site, and pods in other sites, in that order.

For global entitlements that contain dedicated desktop pools, the Cloud Pod Architecture feature uses the default search behavior only the first time a user requests a desktop. After the Cloud Pod Architecture feature allocates a dedicated desktop, it returns the user directly to the same desktop.

You can modify desktop search and allocation behavior for individual global entitlements by setting the scope policy and configuring home sites.

### Configuring Scope Policy to Control Desktop Search

When you create a global entitlement, you must specify its scope policy. The scope policy determines the scope of the search when the Cloud Pod Architecture feature looks for desktops to satisfy a desktop request from the global entitlement.

You can set the scope policy so that the Cloud Pod Architecture feature searches for desktops only on the pod to which the user is connected, only on pods within the same site as the user's pod, or across all pods in the pod federation.

For global entitlements that contain dedicated desktop pools, the scope policy affects where the Cloud Pod Architecture feature looks for desktops only the first time a user requests a dedicated desktop. After the Cloud Pod Architecture feature allocates a dedicated desktop, it returns the user directly to the same desktop.

For information about configuring the scope policy for a global entitlement, see [“Create and Configure a Global Entitlement,”](#) on page 18.

## Configuring Home Sites to Control Desktop Placement

A home site is the affinity between a user and a Cloud Pod Architecture site. With home sites, you can ensure that a user always receives desktops from a specific site rather than receiving desktops based on the user's current location. The Cloud Pod Architecture feature includes the following types of home site assignments.

### Global home site

You can assign home sites to users and groups. If a user who has a home site belongs to a group that is associated with a different home site, the home site associated with the user takes precedence over the group home site assignment.

Global home sites are useful for controlling where roaming users receive desktops. For example, if a user has a home site in New York but is visiting London, the Cloud Pod Architecture feature looks in the New York site to satisfy the user's desktop request rather than allocating a desktop closer to the user. Global home site assignments apply for all global entitlements.

---

**IMPORTANT** Global entitlements do not recognize home sites by default. To make a global entitlement use home sites, you must specify the `--fromHome` option when you create or modify the global entitlement.

---

### Per-global-entitlement home site

When you create a home site for a user or group, you can use the `--entitlementName` option to specify a global entitlement. Per-global-entitlement home sites override global home site assignments.

For example, if a user who has a home site in New York accesses a global entitlement that associates that user with the London home site, the Cloud Pod Architecture feature looks in the London site to satisfy the user's desktop request rather than allocating a desktop from the New York site.

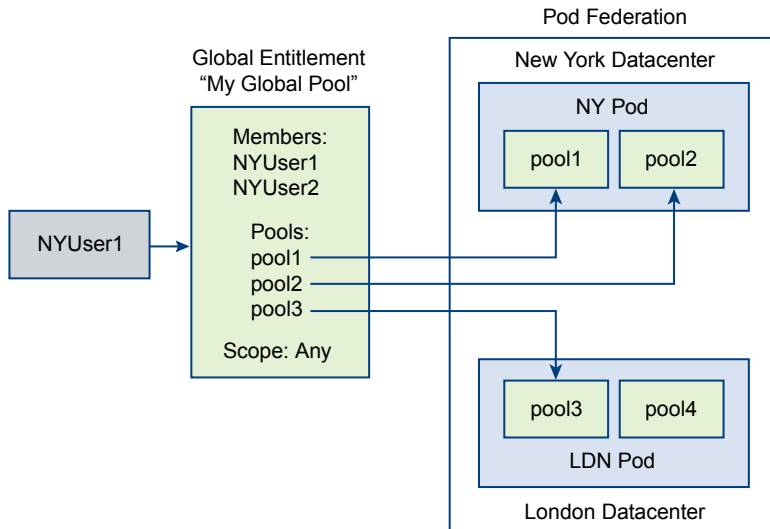
When creating a per-global-entitlement home site, you must explicitly entitle all Active Directory user groups that contain the home site users. If you have nested user groups, it is not sufficient to entitle only the parent group. In this case, the parent group is explicitly entitled to the global entitlement, but the subgroups are not, and the `--createGroupHomeSite` option fails.

Configuring home sites is optional. If a user does not have a home site, the Cloud Pod Architecture feature searches for and allocates desktops as described in [“Finding and Allocating Desktops in a Pod Federation,”](#) on page 10.

For information about creating home sites, see [“Assign a Home Site to a User or Group,”](#) on page 22. For information about creating global entitlements, see [“Create and Configure a Global Entitlement,”](#) on page 18.

## Global Entitlement Example

The following diagram is a conceptual example of a global entitlement. In this example, NYUser1 is a member of the global entitlement called My Global Pool. My Global Pool provides an entitlement to three floating desktop pools, called pool1, pool2, and pool3. pool1 and pool2 are in a pod called NY Pod in the New York datacenter and pool3 and pool4 are in a pod called LDN Pod in the London datacenter.



Because My Global Pool has a scope policy of ANY, the Cloud Pod Architecture feature looks for desktops across both NY Pod and LDN Pod when NYUser1 requests a desktop. The Cloud Pod Architecture feature does not try to allocate a desktop from pool4 because pool4 is not part of My Global Pool.

If NYUser1 logs into NY Pod, the Cloud Pod Architecture feature allocates a desktop from pool1 or pool2, if a desktop is available. If a desktop is not available in either pool1 or pool2, the Cloud Pod Architecture feature allocates a desktop from pool3.

For information about creating global entitlements, see [“Create and Configure a Global Entitlement,”](#) on page 18.

## Cloud Pod Architecture Topology Limits

A typical Cloud Pod Architecture topology consists of two or more View pods, which are linked together in a pod federation. Pod federations are subject to certain limits.

**Table 2-1.** Pod Federation Limits

Component	Limit
Desktops	20,000
Pods	4
Sites	2
View Connection Server instances	20

## Cloud Pod Architecture Port Requirements

Certain network ports must be opened on the Windows firewall for the Cloud Pod Architecture feature to work. When you install View Connection Server, the installation program can optionally configure the required firewall rules for you. These rules open the ports that are used by default. If you change the default ports after installation, or if there are other firewalls in your network, you must manually configure the Windows firewall.

**Table 2-2.** Ports Opened During View Connection Server Installation

TCP Port	Description
22389	The Global Data Layer LDAP instance runs on this port. Shared data is replicated on every View Connection Server instance in a pod federation. Each View Connection Server instance in a pod federation runs a second LDAP instance to store shared data.
8472	The View Interpod API (VIPA) interpod communication channel runs on this port. View Connection Server instances use the VIPA interpod communication channel to launch new desktops, find existing desktops, and share health status data and other information.

## Security Considerations for Cloud Pod Architecture Topologies

To use the `lmvutil` command to configure and manage a Cloud Pod Architecture environment, you must run the command as a user who has the Administrators role. Users who have the Administrators role on the root access group are super users.

When a View Connection Server instance is part of a replicated group of View Connection Server instances, the rights of super users are extended to other View Connection Server instances in the pod. Similarly, when a pod is joined to a pod federation, the rights of super users are extended to all of the View Connection Server instances in all of the pods in the pod federation. These rights are necessary to modify global entitlements and perform other operations on the Global Data Layer.

If you do not want certain super users to be able to perform operations on the Global Data Layer, you can remove the Administrators role assignment and assign the Local Administrators role instead. Users who have the Local Administrators role have super user rights only on their local View Connection Server instance and on any instances in a replicated group.

For information about assigning roles in View Administrator, see "Configuring Role-Based Delegated Administration" in the *View Administration* document.



# Setting Up a Cloud Pod Architecture Environment

---

# 3

Setting up a Cloud Pod Architecture environment involves initializing the Cloud Pod Architecture feature, joining pods to the pod federation, and creating global entitlements. You can optionally create sites and assign home sites.

This chapter includes the following topics:

- [“Initialize the Cloud Pod Architecture Feature,”](#) on page 15
- [“Join Pods to the Pod Federation,”](#) on page 16
- [“Find and Change a Pod Name,”](#) on page 17
- [“Create and Configure a Global Entitlement,”](#) on page 18
- [“Create and Configure a Site,”](#) on page 21
- [“Assign a Home Site to a User or Group,”](#) on page 22
- [“Test a Cloud Pod Architecture Configuration,”](#) on page 23
- [“Sample Scenario: Setting Up a Basic Cloud Pod Architecture Environment,”](#) on page 24

## Initialize the Cloud Pod Architecture Feature

Before you configure a Cloud Pod Architecture environment, you must initialize the Cloud Pod Architecture feature.

You can initialize the Cloud Pod Architecture feature from any View Connection Server instance in a pod. You need to initialize the Cloud Pod Architecture feature only once, on the first pod in a pod federation. When you add pods to the pod federation, the new pods join the initialized pod.

### Prerequisites

- Verify that the latest version of View is installed on all View Connection Server instances in the pod and in all pods that are to join the pod federation.
- Verify that the latest version of View Agent is running on all desktops in the pod and in all pods that are to join the pod federation.
- Become familiar with the `lmvutil` command authentication options and requirements and verify that you have sufficient privileges to run the `lmvutil` command. See [“lmvutil Command Authentication,”](#) on page 43.

**Procedure**

- ◆ On any View Connection Server instance in the pod, run the `lmvutil` command with the `--initialize` option.

```
lmvutil --initialize
```

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --initialize
```

In this example, the user is authenticating as `adminEast` in the domain `domainEast`. Because the user types `"*` instead of an actual password, the command prompts the user for a password. You must provide authentication information, even if you are logged in as the same administrator user that runs the `lmvutil` command.

During the initialization process, View sets up the Global Data Layer on each View Connection Server instance in the pod, configures the VIPA interpod communication channel, and establishes a replication agreement between each View Connection Server instance. The initialization process can take several minutes.

After the initialization process is finished, the pod federation contains a single initialized pod. The Cloud Pod Architecture feature assigns a default name to the pod based on the host name of the View Connection Server instance on which you ran the initialization command. For example, if the host name is `CS1`, the default pod name is `Cluster-CS1`. You can change the default pod name. See [“Find and Change a Pod Name,”](#) on page 17.

**What to do next**

To add more pods to the pod federation, see [“Join Pods to the Pod Federation,”](#) on page 16.

## Join Pods to the Pod Federation

During the Cloud Pod Architecture initialization process, the Cloud Pod Architecture feature creates a pod federation that contains a single pod. You can use the `lmvutil` command to join additional pods to the pod federation. Joining additional pods is optional.

---

**IMPORTANT** Do not stop or start a View Connection Server instance while you are joining it to a pod federation. The View Connection Server service might not restart correctly. You can stop and start the View Connection Server after it is successfully joined to the pod federation.

---

**Prerequisites**

- Make sure that the View Connection Server instances that you want to join have different host names. You cannot join servers that have the same name, even if they are in different domains.
- Become familiar with the `lmvutil` command authentication options and requirements and verify that you have sufficient privileges to run the `lmvutil` command. See [“lmvutil Command Authentication,”](#) on page 43.
- Initialize the Cloud Pod Architecture feature. See [“Initialize the Cloud Pod Architecture Feature,”](#) on page 15.



**Procedure**

- ◆ On a View Connection Server instance in the pod that you are joining to the pod federation, run the `lmvutil` command with the `--join` option.

Repeat this command for each pod that you want to join to the pod federation.

---

**IMPORTANT** You must run this command on a View Connection Server instance in the pod that you are joining to the pod federation.

---

```
lmvutil --join --joinServer serveraddress --userName domain\username --password password
```

Option	Description
<code>--joinServer</code>	Specifies the DNS name or IP address of any View Connection Server instance in any pod that has been initialized or is already part of the pod federation.
<code>--userName</code>	Specifies the name of a View administrator user on the already initialized pod. Use the format <code>domain\username</code> .
<code>--password</code>	Specifies the password of the user specified in the <code>--userName</code> option.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --join --joinServer 123.456.789.1 --userName mydomain\admin2 --password 123secret
```

After you finish joining pods to the pod federation, the pods begin to share health data. You can view health data on the dashboard in View Administrator. See [“View Pod Federation Health in View Administrator,”](#) on page 31.

---

**NOTE** A short delay might occur before health data is available in View Administrator.

---

**What to do next**

When you join a pod to the pod federation, the Cloud Pod Architecture feature assigns a default name to the pod. You can change the default name to a name that reflects your network topology. See [“Find and Change a Pod Name,”](#) on page 17.

## Find and Change a Pod Name

The Cloud Pod Architecture feature assigns default names to the pods in a pod federation. You can use `lmvutil` commands to list the names of the pods in your pod federation and change the default names to names that reflect your network topology. Changing pod names is optional.

The Cloud Pod Architecture feature assigns a default pod name to the first pod in a pod federation when you initialize the Cloud Pod Architecture feature. Subsequent pods receive default names when you join them to the pod federation.

**Prerequisites**

- Become familiar with the `lmvutil` command authentication options and requirements and verify that you have sufficient privileges to run the `lmvutil` command. See [“lmvutil Command Authentication,”](#) on page 43.
- Initialize the Cloud Pod Architecture feature. See [“Initialize the Cloud Pod Architecture Feature,”](#) on page 15.

**Procedure**

- 1 Run the `lmvutil` command with the `--listPods` option to list the names of the pods in your Cloud Pod Architecture topology.

You can run this command on any View Connection Server in the pod federation.

```
lmvutil --listPods
```

For example:

```
lmvutil --authAs adminEast --authDomain example --authPassword "*"
--listPods
```

The command lists the pod name, pod description (if any), and the site to which the pod belongs.

- 2 Run the `lmvutil` command with the `--updatePod` option to change the pod name.

You must run this command on a View Connection Server instance in the pod.

```
lmvutil --updatePod --podName podname [--newPodName podname] [--description text]
```

Option	Description
<b>--podName</b>	Specifies the name of the pod to change.
<b>--newPodName</b>	Specifies the new name for the pod. A pod name can contain between 1 and 64 characters.
<b>--description</b>	(Optional) Provides a description of the pod. The description can contain between 1 and 1024 characters.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*"
--updatePod --podName "Default Pod" --newPodName "East Pod 2"
```

## Create and Configure a Global Entitlement

You use global entitlements to entitle users and groups to desktops in a Cloud Pod Architecture environment. A global entitlement provides the link between users and their desktops, regardless of where those desktops reside in the pod federation. You must create and configure at least one global entitlement to use the Cloud Pod Architecture feature.

A global entitlement contains a list of member users or groups, a list of the desktop pools that can provide desktops for entitled users, and a set of desktop policies. You can add both users and groups, only users, or only groups, to a global entitlement. You can add a particular desktop pool to only one global entitlement.

**Prerequisites**

- Decide which type of global entitlement to create, the users, groups, and pools to include in the global entitlement, and the scope of the global entitlement. See [“Entitling Users and Groups in a Pod Federation,”](#) on page 10.
- Decide whether the global entitlement should use home sites. See [“Configuring Home Sites to Control Desktop Placement,”](#) on page 11.
- Create the desktop pools to include in the global entitlement. For information about creating desktop pools in View, see the *Setting Up Desktop and Application Pools in View* document.
- Create the users and groups to include in the global entitlement.
- Become familiar with the `lmvutil` command authentication options and requirements and verify that you have sufficient privileges to run the `lmvutil` command. See [“lmvutil Command Authentication,”](#) on page 43.

- Initialize the Cloud Pod Architecture feature. See [“Initialize the Cloud Pod Architecture Feature,”](#) on page 15.

### Procedure

- 1 Run the `lmvutil` command with the `--createGlobalEntitlement` option to create the global entitlement.

You can run this command on any View Connection Server in the pod federation.

```
lmvutil --createGlobalEntitlement --entitlementName name --scope scope [--isDedicated | --isFloating] [--description text] [--disabled] [--fromHome] [--multipleSessionAutoClean] [--requireHomeSite] [--defaultProtocol value] [--preventProtocolOverride] [--allowReset]
```

Option	Description
<b>--entitlementName</b>	Specifies the name of the global entitlement. The name can contain between 1 and 64 characters. The global entitlement name appears in the list of available entitlements for the user in Horizon Client.
<b>--scope</b>	Specifies the scope of the global entitlement. Valid values are as follows: <ul style="list-style-type: none"> <li>■ ANY - View looks for desktops on any pod in the pod federation.</li> <li>■ SITE - View looks for desktops only on pods within the same site as the pod to which the user is connected.</li> <li>■ LOCAL - View looks for desktops only in the pod to which the user is connected.</li> </ul>
<b>--isDedicated</b>	Creates a dedicated entitlement. A dedicated entitlement can contain only dedicated desktop pools. To create a floating entitlement, use the <code>--isFloating</code> option. An entitlement can be either dedicated or floating. You cannot specify the <code>--isDedicated</code> option with the <code>--multipleSessionAutoClean</code> option.
<b>--isFloating</b>	Creates a floating entitlement. A floating entitlement can contain only floating desktop pools. To create a dedicated entitlement, specify the <code>--isDedicated</code> option. An entitlement can be either floating or dedicated.
<b>--description</b>	(Optional) Specifies a description of the global entitlement. The description can contain between 1 and 1024 characters.
<b>--disabled</b>	(Optional) Creates the global entitlement in the disabled state.
<b>--fromHome</b>	(Optional) If the user has a home site, causes View to look for desktops on the user's home site. If the user does not have a home site, View begins searching for desktops on the site to which the user is currently connected.
<b>--multipleSessionAutoClean</b>	(Optional) Logs off extra user sessions for the same entitlement. Multiple floating desktop sessions can occur when a pod that contains a session goes offline, the user logs in again and starts another session, and the problem pod comes back online with the original session. When multiple sessions occur, Horizon Client prompts the user to select a session. This option determines what happens to sessions that the user does not select. If you do not specify this option, users must manually end their own extra sessions, either by logging off in Horizon Client or by launching the sessions and logging them off.
<b>--requireHomeSite</b>	(Optional) Causes the global entitlement to be available only if the user has a home site. This option is applicable only when the <code>--fromHome</code> option is also specified.
<b>--defaultProtocol</b>	(Optional) Specifies a default display protocol for desktops in the global entitlement. Valid values are RDP and PCOIP.

Option	Description
<b>--preventProtocolOverride</b>	(Optional) When this option is specified, users cannot override the default display protocol for desktops in the global entitlement. If you do not specify this option, users can override the default display protocol.
<b>--allowReset</b>	(Optional) When this option is specified, users can reset desktops in the global entitlement. If you do not specify this option, users cannot reset desktops.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --
createGlobalEntitlement -entitlementName "Agent Sales" --scope LOCAL
--isDedicated
```

- 2 On the View Connection Server instance in the pod that contains the desktop pool to add to the global entitlement, run the `lmvutil` command with the `--addPoolAssociation` option.

Repeat the command for each desktop pool that you want to add to the global entitlement.

**IMPORTANT** You must run this command on a View Connection Server instance in the pod that contains the desktop pool to add to the global entitlement.

```
lmvutil --addPoolAssociation --entitlementName name --poolId poolid
```

Option	Description
<b>--entitlementName</b>	Specifies the name of the global entitlement to which to add the desktop pool. The name must match the name of an existing global entitlement.
<b>--poolId</b>	Specifies the ID of the desktop pool to add to the global entitlement. The pool ID must match the desktop pool name as it appears on the pod.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --addPoolAssociation --
entitlementName "Agent Sales" --poolId "Sales B"
```

- 3 To add a user to the global entitlement, run the `lmvutil` command with the `--addUserEntitlement` option.

You can run this command on any View Connection Server instance in the pod federation. Repeat the command for each user that you want to add to the global entitlement.

```
lmvutil --addUserEntitlement --userName domain\username --entitlementName name
```

Option	Description
<b>--userName</b>	Specifies the name of a user to add to the global entitlement. Use the format <code>domain\username</code> .
<b>--entitlementName</b>	Specifies the name of the global entitlement to which to add the user. The name must match the name of an existing global entitlement.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --addUserEntitlement --
userName domainCentral\adminCentral --entitlementName "Agent Sales"
```

- 4 To add a group to the global entitlement, run the `lmvutil` command with the `--addGroupEntitlement` option.

You can run this command on any View Connection Server instance in the pod federation. Repeat the command for each group that you want to add to the global entitlement.

```
lmvutil --addGroupEntitlement --groupName domain\groupname --entitlementName name
```

Option	Description
<code>--groupName</code>	Specifies the name of a group to add to the global entitlement. Use the format <code>domain\groupname</code> .
<code>--entitlementName</code>	Specifies the name of the global entitlement to which to add the group. The name must match the name of an existing global entitlement.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*"
--addGroupEntitlement --groupName domainCentral\adminCentralGroup --entitlementName "Agent
Sales"
```

The Cloud Pod Architecture feature stores the global entitlement in the Global Data Layer, which replicates the global entitlement on every pod in the pod federation. When an entitled user uses Horizon Client to connect to a desktop, the global entitlement name appears in the list of available desktop pools.

When you add a desktop pool to a global entitlement, View checks to make sure that the desktop pool's default display protocol, protocol override, and virtual machine reset policies support the equivalent policies set for the global entitlement.

If a View administrator changes the pool-level display protocol or protocol override policy after a pool is associated with a global entitlement, users can receive a desktop launch error when they select the global entitlement. If a View administrator changes the pool-level virtual machine reset policy after a pool is associated with the global entitlement, users can receive an error if they try to reset the virtual machine.

## Create and Configure a Site

By default, the Cloud Pod Architecture feature places all pods into a default site called Default First Site. If your Cloud Pod Architecture topology contains multiple pods, you might want to group those pods into different sites. The Cloud Pod Architecture feature treats pods in the same site equally.

### Prerequisites

- Decide whether your Cloud Pod Architecture topology should include sites. See [“Creating Cloud Pod Architecture Sites,”](#) on page 9.
- Determine the names of the pods to add to the site. See [“Find and Change a Pod Name,”](#) on page 17.
- Become familiar with the `lmvutil` command authentication options and requirements and verify that you have sufficient privileges to run the `lmvutil` command. See [“lmvutil Command Authentication,”](#) on page 43.
- Initialize the Cloud Pod Architecture feature. See [“Initialize the Cloud Pod Architecture Feature,”](#) on page 15.

**Procedure**

- 1 On any View Connection Server instance in the pod federation, run the `lmvutil` command with the `--createSite` option.

```
lmvutil --createSite --siteName sitename [--description text]
```

Option	Description
<code>--siteName</code>	Specifies the name of the new site. The site name can contain between 1 and 64 characters.
<code>--description</code>	(Optional) Provides a description of the site. The description can contain between 1 and 1024 characters.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --createSite
--siteName "Eastern Region"
```

- 2 On any View Connection Server instance in the pod federation, run the `lmvutil` command with the `--assignPodToSite` option to assign a pod to the site.

Repeat this command for each pod that you want to add to the site.

```
lmvutil --assignPodToSite --podName podname --siteName sitename
```

Option	Description
<code>--podName</code>	Specifies the name of the pod to assign to the site.
<code>--siteName</code>	Specifies the name of the site.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*"
--assignPodToSite --podName "East Pod 1" --siteName "Eastern Region"
```

## Assign a Home Site to a User or Group

A home site is the affinity between a user and a Cloud Pod Architecture site. Home sites ensure that users always receive desktops from a particular datacenter, even when they are traveling. Creating home sites is optional.

**Prerequisites**

- Decide whether to assign home sites to users or groups in your Cloud Pod Architecture environment. See [“Configuring Home Sites to Control Desktop Placement,”](#) on page 11 .
- Group the pods in your pod federation into sites. See [“Create and Configure a Site,”](#) on page 21.
- Global entitlements do not use home sites by default. When creating a global entitlement, you must specify the `--fromHome` option to cause View to use a user's home site when allocating desktops from that global entitlement. See [“Create and Configure a Global Entitlement,”](#) on page 18.
- Become familiar with the `lmvutil` command authentication options and requirements and verify that you have sufficient privileges to run the `lmvutil` command. See [“lmvutil Command Authentication,”](#) on page 43.
- Initialize the Cloud Pod Architecture feature. See [“Initialize the Cloud Pod Architecture Feature,”](#) on page 15.

**Procedure**

- To create a home site for a user, run the `lmvutil` command with the `--createUserHomeSite` option.

You can run the command on any View Connection Server instance in the pod federation.

```
lmvutil --createUserHomeSite --userName domain\username --siteName name [--entitlementName name]
```

Option	Description
<code>--userName</code>	Specifies the name of the user. Use the format <code>domain\username</code> .
<code>--siteName</code>	Specifies the name of the site to associate with the user as the home site.
<code>--entitlementName</code>	(Optional) Specifies the name of a global entitlement to associate with the home site. When a user selects the specified global entitlement, the home site overrides the user's own home site. If you do not specify this option, the command creates a global user home site.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --createUserHomeSite --userName example\adminEast --siteName "Eastern Region" --entitlementName "Agent Sales"
```

- To create a home site for a group, run the `lmvutil` command with the `--createGroupHomeSite` option.

You can run the command on any View Connection Server instance in the pod federation.

```
lmvutil --createGroupHomeSite --userName domain\username --siteName name [--entitlementName name]
```

Option	Description
<code>--groupName</code>	Specifies the name of the group. Use the format <code>domain\groupname</code> .
<code>--siteName</code>	Specifies the name of the site to associate with the group as the home site.
<code>--entitlementName</code>	(Optional) Specifies the name of a global entitlement to associate with the home site. When a user selects the specified global entitlement, the home site overrides the user's own home site. If you do not specify this option, the command creates a global group home site.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --createGroupHomeSite --groupName example\adminEastGroup --siteName "Eastern Region" --entitlementName "Agent Sales"
```

## Test a Cloud Pod Architecture Configuration

After you initialize and configure a Cloud Pod Architecture environment, you should perform certain steps to verify that your environment is set up properly.

**Prerequisites**

- Install the latest version of Horizon Client on a supported computer or mobile device.
- Verify that you have credentials for a user in one of your newly created global entitlements.

**Procedure**

- 1 Launch Horizon Client.
- 2 Connect to any View Connection Server instance in the pod federation by using the credentials of a user in one of your newly created global entitlements.

After you connect to the View Connection Server instance, the global entitlement name should appear in the list of available desktop pools.

- 3 Select the global entitlement and connect to a desktop.

A desktop should launch successfully. Exactly which desktop launches depends on the individual configuration of the global entitlement, pods, and desktop pools. As a general rule, the Cloud Pod Architecture feature attempts to allocate a desktop from the pod to which you are connected.

#### What to do next

If the global entitlement does not appear when you connect to the View Connection Server instance, use the `lmvutil` command with the `--listUserEntitlements`, `--listGroupEntitlements`, and `--listAssociatedPools` options to verify that the entitlement is configured correctly. If the global entitlement appears but the desktop does not launch, all desktop pools might be fully assigned to other users.

## Sample Scenario: Setting Up a Basic Cloud Pod Architecture Environment

This sample scenario demonstrates how the Cloud Pod Architecture feature can solve a real-world problem. It includes step-by-step instructions that you can follow to complete a Cloud Pod Architecture configuration.

In this scenario, a health insurance company has a mobile sales force that operates across two regions, the Central region and the Eastern region. Sales agents use mobile devices to present insurance policy quotes to customers and customers view and sign digital documents.

Rather than store customer data on their mobile devices, sales agents use standardized View floating desktops. Access to customer data is kept secure in the health insurance company's datacenters.

The health insurance company has two datacenters, one in each region. Occasional capacity problems cause sales agents to look for available desktops in a non-local datacenter, and WAN latency problems sometimes occur. If sales agents disconnect from desktops but leave their sessions logged in, they must remember which datacenter hosted their sessions to reconnect to their desktops.

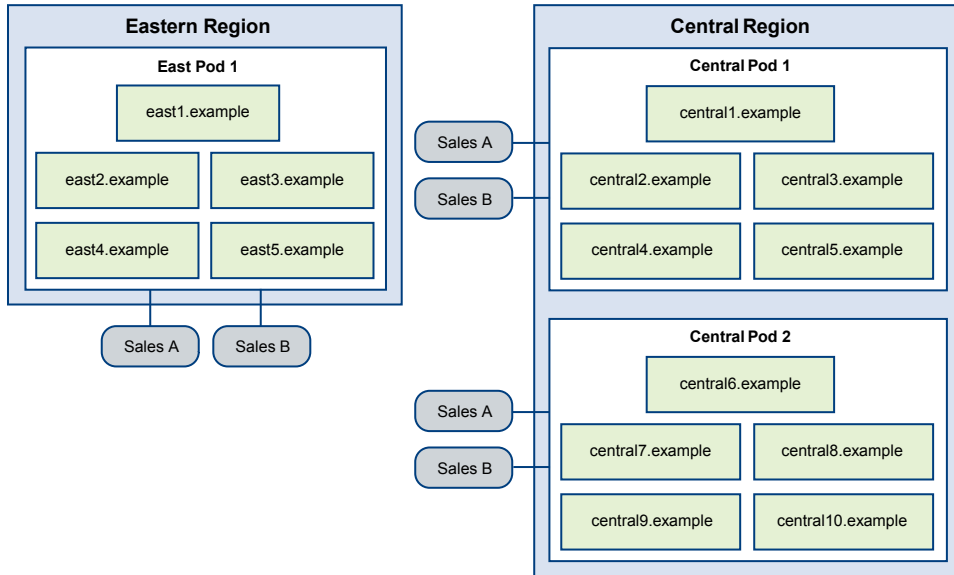
To solve these problems, the health insurance company designs a Cloud Pod Architecture topology, initializes the Cloud Pod Architecture feature, joins its existing pods to the pod federation, creates sites for each of its datacenters, entitles its sales agents to all of its desktop pools, and implements a single View URL.



## Designing the Cloud Pod Architecture Topology

The insurance company designs a Cloud Pod Architecture topology that includes two sites, one for each region.

**Figure 3-1.** Sample Scenario Cloud Pod Architecture Topology



In this topology, the Eastern region site contains a single pod that consists of five View Connection Server instances called east1.example through east5.example.

The Central region site contains two pods, each of which contains five View Connection Server instances. The View Connection Servers in the first pod are called central1.example through central5.example and the View Connection Server instances in the second pod are called central6.example through central10.example.

Each pod in the topology contains two desktop pools, called Sales A and Sales B, of sales agent desktops.

## Initializing the Cloud Pod Architecture Feature

A View administrator runs the `lmvutil` command with the `--initialize` option to initialize the Cloud Pod Architecture feature.

A View administrator can run this command on any View Connection Server instance in a pod. In this example, the View administrator runs the command on east1.example.

```
lmvutil --authAs adminEast --authDomain example --authPassword secret123 --initialize
```

The five View Connection Server instances in the insurance company's Eastern region are now part of a pod federation.

## Joining Each Pod to the Pod Federation

A View administrator runs the `lmvutil` command with the `--join` option on each pod in the Central region to join those pods to the pod federation.

A View administrator can run this command on any View Connection Server instance in a pod. In this example, the View administrator runs the command for the first pod on central1.example.

```
lmvutil --authAs adminCentral --authDomain example --authPassword secret123 --join --joinServer east1.example --userName example\adminEast --password secret123
```

After the first Central region pod is joined to the pod federation, the View administrator joins the second Central region pod to the pod federation. In this example, the View administrator runs the command for the second pod on central6.example.

```
lmvutil --authAs adminCentral --authDomain example --authPassword secret456 --join --joinServer east1.example --userName example\adminEast --password secret123
```

After the second command finishes, all 10 View Connection Server instances across both pods in the insurance company's Central region are part of the pod federation.

## Creating Sites for the Eastern and Central Region Datacenters

A View administrator creates a site for the insurance company's Central and Eastern datacenters and adds pods to those sites.

- 1 The View administrator runs the `lmvutil` command with the `--createSite` option to create each site. A View administrator can run these commands on any View Connection Server instance in the pod federation. In this example, the View administrator runs the commands on `east1.example`.

This command creates a site for the Eastern region datacenter:

```
lmvutil --authAs adminEast --authDomain example --authPassword "*" --createSite --siteName "Eastern Region"
```

This command creates a site for the Central region datacenter:

```
lmvutil --authAs adminEast --authDomain example --authPassword "*" --createSite --siteName "Central Region"
```

- 2 The View administrator runs the `lmvutil` command with the `--listPods` option to determine the default names of the pods in the pod federation. A View administrator can run this command on any View Connection Server instance in the pod federation. In this example, the View administrator runs the command on `east1.example`.

```
lmvutil --authAs adminEast --authDomain example --authPassword "*" --listPods
```

- 3 The View administrator runs the `lmvutil` command with the `--updatePod` option for each pod to change its default name. A View administrator can run this command on any View Connection Server instance in the pod federation. In this example, the View administrator runs the commands on `east1.example`.

This command changes the name of the Eastern region pod to East Pod 1:

```
lmvutil --authAs adminEast --authDomain example --authPassword "*" --updatePod --podName "Cluster-east1" --newPodName "East Pod 1"
```

This command changes the name of the first Central region pod to Central Pod 1:

```
lmvutil --authAs adminEast --authDomain example --authPassword "*" --updatePod --podName "Cluster-central1" --newPodName "Central Pod 1"
```

This command changes the name of the second Central region pod to Central Pod 2:

```
lmvutil --authAs adminEast --authDomain example --authPassword "*" --updatePod --podName "Cluster-central6" --newPodName "Central Pod 2"
```

- 4 The View administrator runs the `lmvutil` command with the `--assignPodToSite` option for each pod to assign it to the newly created sites. A View administrator can run this command on any View Connection Server instance in the pod federation. In this example, the View administrator runs the commands on `east1.example`.

This command adds the East Pod 1 pod to the Eastern Region site:

```
lmvutil --authAs adminEast --authDomain example --authPassword "*" --assignPodToSite --podName "East Pod 1" --siteName "Eastern Region"
```

This command adds the Central Pod 1 pod to the Central Region site:

```
lmvutil --authAs adminEast --authDomain example --authPassword "*" --assignPodToSite --
podName "Central Pod 1" --siteName "Central Region"
```

This command adds the Central Pod 2 pod to the Central Region site:

```
lmvutil --authAs adminEast --authDomain example --authPassword "*" --assignPodToSite --
podName "Central Pod 2" --siteName "Central Region"
```

The pod federation site topology now reflects the geographic distribution of pods in the insurance company's network.

## Creating a Single Global Entitlement

A View administrator creates a single global entitlement to entitle all sales agents to all desktops in the sales agent desktop pools across all pods in the pod federation.

- 1 The View administrator runs the `lmvutil` command with the `--createGlobalEntitlement` option to create the global entitlement. A View administrator can run this command on any View Connection Server instance in the pod federation. In this example, the View administrator runs the command on `east1.example`.

```
lmvutil --authAs adminEast --authDomain example --authPassword "*" --createGlobalEntitlement
--entitlementName "Agent Sales" --isFloating --scope ANY
```

- 2 The View administrator runs the `lmvutil` command with the `--addPoolAssociation` option to add each desktop pool to the newly created Agent Sales global entitlement. A View administrator must run these commands on the View Connection Server instances where the pools are located. Each pod in the insurance company's View implementation has two pools, called Sales A and Sales B.

- To add the pools in the Eastern region pod to the Agent Sales global entitlement, the View administrator runs the following commands on `east1.example`.

```
lmvutil --authAs adminEast --authDomain example --authPassword "*" --addPoolAssociation
--entitlementName "Agent Sales" --poolId "Sales A"
```

```
lmvutil --authAs adminEast --authDomain example --authPassword "*" --addPoolAssociation
--entitlementName "Agent Sales" --poolId "Sales B"
```

- To add the pools in the first Central region pod to the Agents Sales global entitlement, the View administrator runs the following commands on `central1.example`.

```
lmvutil --authAs adminCentral --authDomain example --authPassword "*" --
addPoolAssociation --entitlementName "Agent Sales" --poolId "Sales A"
```

```
lmvutil --authAs adminCentral --authDomain example --authPassword "*" --
addPoolAssociation --entitlementName "Agent Sales" --poolId "Sales B"
```

- To add the pools in the second Central region pod to the Agent Sales global entitlement, the View administrator runs the following commands on `central6.example`.

```
lmvutil --authAs adminCentral --authDomain example --authPassword "*" --
addPoolAssociation --entitlementName "Agent Sales" --poolId "Sales A"
```

```
lmvutil --authAs adminCentral --authDomain example --authPassword "*" --
addPoolAssociation --entitlementName "Agent Sales" --poolId "Sales B"
```

- 3 The View administrator runs the `lmvutil` command with the `--addGroupEntitlement` option to add the Sales Agents group to the Agent Sales global entitlement.

The Sales Agent group is defined in Active Directory and contains all sales agent users. Adding the Sales Agent group to the Agent Sales global entitlement enables sales agents to access the Sales A and Sales B desktop pools on the pods in the Eastern and Central regions.

A View administrator can run this command on any View Connection Server instance in the pod federation. In this example, the View administrator runs the command on central6.example.

```
lmvutil --authAs adminCentral --authDomain example --authPassword "*" --addGroupEntitlement  
--entitlementName "Agent Sales" --groupName example\Sales
```

## Using a Single View URL

The insurance company uses a single View URL and employs a DNS service to resolve sales.example to the nearest pod in the nearest datacenter. With this arrangement, sales agents do not need to remember different URLs for each pod and are always directed to the nearest datacenter, regardless of where they are located.

When a sales agent connects to the View URL in Horizon Client, the Agent Sales global entitlement appears on the list of available desktop pools. When a sales agent selects the global entitlement, the Cloud Pod Architecture feature delivers the nearest available desktop in the pod federation. If all of the desktops in the local datacenter are in use, the Cloud Pod Architecture feature selects a desktop from the other datacenter. If a sales agent leaves a desktop session logged in, the Cloud Pod Architecture feature returns the sales agent to that desktop, even if the sales agent has since traveled to a different region.

# Managing a Cloud Pod Architecture Environment

# 4

You use `lmvutil` commands to view, modify, and maintain your Cloud Pod Architecture environment. You can use View Administrator to monitor the health of pods in the pod federation.

This chapter includes the following topics:

- [“View a Cloud Pod Architecture Configuration,”](#) on page 29
- [“View Pod Federation Health in View Administrator,”](#) on page 31
- [“View Desktop Sessions in a Pod Federation,”](#) on page 31
- [“Determine the Effective Home Site for a User,”](#) on page 32
- [“Add a Pod to a Site,”](#) on page 33
- [“Remove a Pod From a Pod Federation,”](#) on page 33
- [“Modifying Global Entitlements,”](#) on page 34
- [“Remove a Home Site Association,”](#) on page 40
- [“Disable the Cloud Pod Architecture Feature,”](#) on page 41

## View a Cloud Pod Architecture Configuration

You can use `lmvutil` commands to view a Cloud Pod Architecture configuration information, including information about global entitlements, home sites, and pod federation topology.

You can run these commands on any View Connection Server instance in the pod federation.

### Prerequisites

Become familiar with the `lmvutil` command authentication options and requirements and verify that you have sufficient privileges to run the `lmvutil` command. See [“`lmvutil` Command Authentication,”](#) on page 43.

### Procedure

- To list all of the global entitlements in your configuration, run the `lmvutil` command with the `--listGlobalEntitlements` option.

```
lmvutil --listGlobalEntitlements
```

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --  
listGlobalEntitlements
```

- To list the desktop pools in a global entitlement, run the `lmvutil` command with the `--listAssociatedPools` option.

The `--entitlementName` option specifies the name of the global entitlement for which to list the associated desktop pools.

```
lmvutil --listAssociatedPools -entitlementName name
```

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --
listAssociatedPools -entitlementName "Agent Sales"
```

- To list the users or groups associated with a global entitlement, run the `lmvutil` command with the `--listEntitlements` option.

```
lmvutil --listEntitlements [--userName domain\username | --groupName domain\groupname | --
entitlementName name]
```

You must type one of the following options.

Option	Description
<code>--userName</code>	Specifies the name of the user for whom you want to list global entitlements. Use the format <code>domain\username</code> . When you use this option, the command lists all the global entitlements associated with the specified user.
<code>--groupName</code>	Specifies the name of the group for which you want to list global entitlements. Use the format <code>domain\groupname</code> . When you use this option, the command lists all of the global entitlements associated with the specified group.
<code>--entitlementName</code>	Specifies the name of a global entitlement. When you use this option, the command lists all of the users and groups in the specified global entitlement.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --listEntitlements --
userName example\adminEast
```

- To list the home sites for a user, run the `lmvutil` command with the `--showUserHomeSites` option.

```
lmvutil --showUserHomeSites --userName domain\username [--entitlementName name]
```

Option	Description
<code>--userName</code>	Specifies the name of the user for whom to list home sites. Use the format <code>domain\username</code> .
<code>--entitlementName</code>	(Optional) Specifies the name of a global entitlement. Use this option to list the home sites for a user and global entitlement combination.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --showUserHomeSites --
userName example\adminEast
```

- To list the home sites for a group, run the `lmvutil` command with the `--showGroupHomeSites` option.

```
lmvutil --showGroupHomeSites --groupName domain\groupname [--entitlementName name]
```

Option	Description
<code>--groupName</code>	Specifies the name of the group for which to list home sites. Use the format <code>domain\groupname</code> .
<code>--entitlementName</code>	(Optional) Specifies the name of a global entitlement. Use this option to list the home sites for a group and global entitlement combination.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --showGroupHomeSites --
groupName example\adminEastGroup
```

- To list the pods in your pod federation, run the `lmvutil` command with the `--listPods` option.

```
lmvutil --listPods
```

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --listPods
```

- To list the sites in your pod federation, run the `lmvutil` command with the `--listSites` option.

```
lmvutil --listSites
```

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --listSites
```

## View Pod Federation Health in View Administrator

View constantly monitors the health of the pod federation by checking the health of each pod and View Connection Server instances in those pods. You can view the health of a pod federation in View Administrator.

You can also view the health of a pod federation from the command line by using the `vdmadmin` command with the `-H` option. For information about `vdmadmin` syntax, see the *View Administration* document.

---

**NOTE** View event databases are not shared across pods in a pod federation.

---

### Procedure

- ◆ In View Administrator, select **Inventory > Dashboard**.

The Remote Pods section in the System Health pane lists all pods, their member View Connection Server instances, and the known health status for each View Connection Server instance.

A green health icon indicates that the View Connection Server instance is online and available for the Cloud Pod Architecture feature. A red health icon indicates that the View Connection Server instance is offline or the Cloud Pod Architecture feature cannot connect to the View Connection Server instance to confirm its availability.

## View Desktop Sessions in a Pod Federation

You can use View Administrator to search for and view desktop sessions across the pod federation.

You can search for desktop sessions by user, pod, or brokering pod. The user is the end-user that is logged in to the desktop, the pod is the pod on which the desktop is hosted, and the brokering pod is the pod to which the user was connected when the desktop was first allocated.

**Procedure**

- 1 In View Administrator, select **Inventory > Search Sessions**.
- 2 Select search criteria and begin the search.

Option	Action
<b>Search by user</b>	<ol style="list-style-type: none"> <li>a Select <b>User</b> from the drop-down menu.</li> <li>b Click in the text box.</li> <li>c Select search criteria in the Find User dialog box and click <b>OK</b>.</li> <li>d Click <b>Search</b> to begin the search.</li> </ol>
<b>Search by pod</b>	<ol style="list-style-type: none"> <li>a Select <b>Pod</b> from the drop-down menu and select a pod from the list of pods that appears.</li> <li>b Click <b>Search</b> to begin the search.</li> </ol>
<b>Search by brokering pod</b>	<ol style="list-style-type: none"> <li>a Select <b>Brokering Pod</b> from the drop-down menu and select a pod from the list of pods that appears.</li> <li>b Click <b>Search</b> to begin the search.</li> </ol>

The search results include the names of the user, machine, pool, pod, brokering pod ID, site, and global entitlements associated with each session. The session start time, duration, and state also appear in the search results.

**NOTE** The brokering pod ID is not immediately populated for new sessions in the search results. This ID usually appears in View Administrator between two and three minutes after a session begins.

## Determine the Effective Home Site for a User

Because you can assign home sites to both users and groups, a single user can have multiple home sites. In addition, home sites associated with global entitlements can override home sites associated with users and groups. For these reasons, it can be difficult to determine the effective home site for a particular user. You can use the `lmvutil` command to list a user's effective home site.

**Prerequisites**

Become familiar with the `lmvutil` command authentication options and requirements and verify that you have sufficient privileges to run the `lmvutil` command. See [“lmvutil Command Authentication,”](#) on page 43.

**Procedure**

- ◆ On any View Connection Server instance in the pod federation, run the `lmvutil` command with the `--resolveUserHomeSite` option.

```
lmvutil --resolveUserHomeSite --entitlementName name --userName domain\username
```

Option	Description
<b>--entitlementName</b>	Specifies the name of a global entitlement. This option enables you to determine the effective home site for a user and global entitlement combination, which might be different from the home site that is configured for the user.
<b>--userName</b>	Specifies the name of the user whose home site you want to list. Use the format <code>domain\username</code> .

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --resolveUserHomeSite --userName domainCentral\adminCentral
```



## Add a Pod to a Site

You can use the `lmvutil` command with the `--assignPodToSite` option to add a pod to an existing site.

### Prerequisites

Become familiar with the `lmvutil` command authentication options and requirements and verify that you have sufficient privileges to run the `lmvutil` command. See [“lmvutil Command Authentication,”](#) on page 43.

### Procedure

- ◆ On any View Connection Server instance in the pod federation, run the `lmvutil` command with the `--assignPodToSite` option to assign a pod to the site.

Repeat this command for each pod that you want to add to the site.

```
lmvutil --assignPodToSite --podName podname --siteName sitename
```

Option	Description
<code>--podName</code>	Specifies the name of the pod to assign to the site.
<code>--siteName</code>	Specifies the name of the site.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*"
--assignPodToSite --podName "East Pod 1" --siteName "Eastern Region"
```

## Remove a Pod From a Pod Federation

You can use `lmvutil` commands to remove a pod that was previously joined to a pod federation. You might want to remove a pod from a pod federation if it is being recommissioned for another purpose or if it was wrongly configured.

You cannot remove the last pod in a pod federation. If you want to disable the Cloud Pod Architecture feature, use the `lmvutil` command with the `--uninitialize` option. See [“Disable the Cloud Pod Architecture Feature,”](#) on page 41.

---

**IMPORTANT** Do not stop or start a View Connection Server instance while it is being unjoined from a pod federation. The View Connection Server service might not restart correctly.

---

### Prerequisites

- Become familiar with the `lmvutil` command authentication options and requirements and verify that you have sufficient privileges to run the `lmvutil` command. See [“lmvutil Command Authentication,”](#) on page 43.
- Determine the name of the pod to remove. You can use the `lmvutil` command with the `--listPods` option to list the names of the pods in a pod federation.

**Procedure**

- If the pod is available, run the `lmvutil` command with the `--unjoin` option.

You must run this command on a View Connection Server instance in the pod that you want to remove from the pod federation.

```
lmvutil --unjoin
```

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --unjoin
```

- If the pod is not available (for example, in the case of a hardware failure), run the `lmvutil` command with the `--ejectPod` option.

You can run this command on any View Connection Server instance in the pod federation.

---

**IMPORTANT** In most circumstances, you should use the `lmvutil` command with the `--unjoin` option to remove a pod from a pod federation.

---

```
lmvutil --ejectPod --pod podname
```

The `--pod` option specifies the name of the pod to remove.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --ejectPod --pod "East Pod 1"
```

## Modifying Global Entitlements

You can add and remove desktop pools, users, and groups from global entitlements. You can also delete global entitlements and modify global entitlement attributes and policies.

### Add a Desktop Pool to a Global Entitlement

You can add a desktop pool to an existing global entitlement. You can add a particular desktop pool to only one global entitlement.

**Prerequisites**

- Create the desktop pool to add to the global entitlement. For information about creating desktop pools, see the *Setting Up Desktop and Application Pools in View* document.
- Become familiar with the `lmvutil` command authentication options and requirements and verify that you have sufficient privileges to run the `lmvutil` command. See [“lmvutil Command Authentication,”](#) on page 43.

**Procedure**

- ◆ On the View Connection Server instance in the pod that contains the desktop pool to add to the global entitlement, run the `lmvutil` command with the `--addPoolAssociation` option.

Repeat the command for each desktop pool that you want to add to the global entitlement.

---

**IMPORTANT** You must run this command on a View Connection Server instance in the pod that contains the desktop pool to add to the global entitlement.

---

```
lmvutil --addPoolAssociation --entitlementName name --poolId poolid
```

Option	Description
<code>--entitlementName</code>	Specifies the name of the global entitlement to which to add the desktop pool. The name must match the name of an existing global entitlement.
<code>--poolId</code>	Specifies the ID of the desktop pool to add to the global entitlement. The pool ID must match the desktop pool name as it appears on the pod.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --addPoolAssociation --entitlementName "Agent Sales" --poolId "Sales B"
```

The Cloud Pod Architecture feature stores the global entitlement in the Global Data Layer, which replicates the global entitlement on every pod in the pod federation. When an entitled user uses Horizon Client to connect to a desktop, the global entitlement name appears in the list of available desktop pools.

When you add a desktop pool to a global entitlement, View checks to make sure that the desktop pool's default display protocol, protocol override, and virtual machine reset policies support the equivalent policies set for the global entitlement.

If a View administrator changes the pool-level display protocol or protocol override policy after a pool is associated with a global entitlement, users can receive a desktop launch error when they select the global entitlement. If a View administrator changes the pool-level virtual machine reset policy after a pool is associated with the global entitlement, users can receive an error if they try to reset the virtual machine.

## Remove a Desktop Pool From a Global Entitlement

You can remove a desktop pool from a global entitlement.

### Prerequisites

Become familiar with the `lmvutil` command authentication options and requirements and verify that you have sufficient privileges to run the `lmvutil` command. See [“lmvutil Command Authentication,”](#) on page 43.

**Procedure**

- ◆ On any View Connection Server instance in the pod federation, run the `lmvutil` command with the `--removePoolAssociation` option.

```
lmvutil --removePoolAssociation -entitlementName name --poolID poolid
```

Option	Description
<code>--entitlementName</code>	Specifies the name of the global entitlement to modify.
<code>--poolID</code>	Specifies the pool ID of the desktop pool to remove from the global entitlement.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*"
--removePoolAssociation -entitlementName "Agent Sales" --poolID "Sales B"
```

**Add a User or Group to a Global Entitlement**

You can add a user or group to an existing global entitlement.

**Prerequisites**

- Create the user or group to add to the global entitlement.
- Become familiar with the `lmvutil` command authentication options and requirements and verify that you have sufficient privileges to run the `lmvutil` command. See [“lmvutil Command Authentication,”](#) on page 43.

**Procedure**

- To add a user to the global entitlement, run the `lmvutil` command with the `--addUserEntitlement` option.

You can run this command on any View Connection Server instance in the pod federation. Repeat the command for each user that you want to add to the global entitlement.

```
lmvutil --addUserEntitlement --userName domain\username --entitlementName name
```

Option	Description
<code>--userName</code>	Specifies the name of a user to add to the global entitlement. Use the format <code>domain\username</code> .
<code>--entitlementName</code>	Specifies the name of the global entitlement to which to add the user. The name must match the name of an existing global entitlement.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --addUserEntitlement --
userName domainCentral\adminCentral --entitlementName "Agent Sales"
```

- To add a group to the global entitlement, run the `lmvutil` command with the `--addGroupEntitlement` option.

You can run this command on any View Connection Server instance in the pod federation. Repeat the command for each group that you want to add to the global entitlement.

```
lmvutil --addGroupEntitlement --groupName domain\groupname --entitlementName name
```

Option	Description
<code>--groupName</code>	Specifies the name of a group to add to the global entitlement. Use the format <code>domain\groupname</code> .
<code>--entitlementName</code>	Specifies the name of the global entitlement to which to add the group. The name must match the name of an existing global entitlement.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*"
--addGroupEntitlement --groupName domainCentral\adminCentralGroup --entitlementName "Agent
Sales"
```

## Remove a User or Group From a Global Entitlement

You can use `lmvutil` commands to remove a user or group from a global entitlement.

### Prerequisites

Become familiar with the `lmvutil` command authentication options and requirements and verify that you have sufficient privileges to run the `lmvutil` command. See [“lmvutil Command Authentication,”](#) on page 43.

### Procedure

- To remove a user from a global entitlement, run the `lmvutil` command with `--removeUserEntitlement` option.

You can run this command on any View Connection Server instance in the pod federation.

```
lmvutil --removeUserEntitlement --userName domain\username --entitlementName name
```

Option	Description
<code>--userName</code>	Specifies the name of the user to remove from the global entitlement. Use the format <code>domain\username</code> .
<code>--entitlementName</code>	Specifies the name of the global entitlement to modify.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*"
--removeUserEntitlement --userName domainCentral\adminCentral --entitlementName "Agent Sales"
```

- To remove a group from a global entitlement, run the `lmvutil` command with the `--removeGroupEntitlement` option.

You can run this command on any View Connection Server instance in the pod federation.

```
lmvutil --removeGroupEntitlement --groupName domain\groupname --entitlementName name
```

Option	Description
<code>--groupName</code>	Specifies the name of the group to remove from the global entitlement. Use the format <code>domain\groupname</code> .
<code>--entitlementName</code>	Specifies the name of the global entitlement to modify.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*"
--removeGroupEntitlement --groupName domainCentral\adminCentralGroup --entitlementName
"Agent Sales"
```

## Modify Attributes or Policies for a Global Entitlement

You can modify the scope, description, and other attributes of a global entitlement. You can also disable or enable a previously disabled global entitlement.

### Prerequisites

Become familiar with the `lmvutil` command authentication options and requirements and verify that you have sufficient privileges to run the `lmvutil` command. See [“Imvutil Command Authentication,”](#) on page 43.

### Procedure

- ◆ On any View Connection Server instance in the pod federation, run the `lmvutil` command with the `--updateGlobalEntitlement` option.

```
lmvutil --updateGlobalEntitlement --entitlementName name [--scope scope] [--description text]
[--disabled] [--enabled] [--fromHome] [--disableFromHome] [--multipleSessionAutoClean] [--
disableMultipleSessionAutoClean] [--requireHomeSite] [--disableRequiredHomeSite] [--
defaultProtocol protocol]
```

Option	Description
<code>--entitlementName</code>	Specifies the name of the global entitlement to modify.
<code>--scope</code>	(Optional) Specifies the scope of the global entitlement. Valid values are as follows: <ul style="list-style-type: none"> <li>■ ANY - View looks for desktops on any pod in the pod federation.</li> <li>■ SITE - View looks for desktops only on pods within the same site as the pod to which the user is connected.</li> <li>■ LOCAL - View looks for desktops only in the pod to which the user is connected.</li> </ul>
<code>--description</code>	(Optional) Specifies a description of the global entitlement. The description can contain between 1 and 1024 characters.
<code>--disabled</code>	(Optional) Disables a previously enabled global entitlement.
<code>--enabled</code>	(Optional) Enables a previously disabled global entitlement.
<code>--fromHome</code>	(Optional) If the user has a home site, causes View to look for desktops on the user's home site. If the user does not have a home site, View begins searching for desktops on the site to which the user is currently connected.
<code>--disableFromHome</code>	(Optional) Disables the <code>--fromHome</code> option functionality if the <code>--fromHome</code> option was previously specified for the global entitlement.

Option	Description
<b>--multipleSessionAutoClean</b>	(Optional) Logs off extra user sessions for the same entitlement. Multiple floating desktop sessions can occur when a pod that contains a session goes offline, the user logs in again and starts another session, and the problem pod comes back online with the original session. When multiple sessions occur, Horizon Client prompts the user to select a session. This option determines what happens to sessions that the user does not select. If you do not specify this option, users must manually end their own extra sessions, either by logging off in Horizon Client or by launching the sessions and logging them off.
<b>--disableMultipleSessionAutoClean</b>	(Optional) Disables the <code>--multipleSessionAutoClean</code> option functionality if the <code>--multipleSessionAutoClean</code> option was previously specified for the global entitlement.
<b>--requireHomeSite</b>	(Optional) Causes the global entitlement to be available only if the user has a home site. This option is applicable only when the <code>--fromHome</code> option is also specified.
<b>--disableRequiredHomeSite</b>	(Optional) Disables the <code>--requireHomeSite</code> option functionality if the <code>--requireHomeSite</code> option was previously specified for the global entitlement.
<b>--defaultProtocol</b>	(Optional) Specifies a default display protocol for desktops in the global entitlement. Valid values are RDP and PCOIP.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --
updateGlobalEntitlement -entitlementName "Agent Sales" --scope ANY
--isDedicated
```

## Delete a Global Entitlement

You can use the `lmvutil` command to permanently delete a global entitlement. When you delete a global entitlement, all of the users who are dependent on that global entitlement for desktops can no longer access their desktops. Any existing desktop sessions remain connected.

To modify the list of users or groups in a global entitlement, use the `lmvutil` command with the `--addUserEntitlement` or `--addGroupEntitlement` option. To disable a global entitlement, use the `lmvutil` command with the `--updateGlobalEntitlement` and `--disabled` options.

### Prerequisites

Become familiar with the `lmvutil` command authentication options and requirements and verify that you have sufficient privileges to run the `lmvutil` command. See [“lmvutil Command Authentication,”](#) on page 43.

### Procedure

- ◆ On any View Connection Server instance in the pod federation, run the `lmvutil` command with the `--deleteGlobalEntitlement` option.

```
lmvutil --deleteGlobalEntitlement --entitlementName name
```

The `--entitlementName` option specifies the name of the global entitlement to delete.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*"
--deleteGlobalEntitlement --entitlementName "Agent Sales"
```

## Remove a Home Site Association

You can use `lmvutil` commands to remove associations between a user or group and a home site. You can also remove the association between a home site and a global entitlement for a specified user or group.

### Prerequisites

Become familiar with the `lmvutil` command authentication options and requirements and verify that you have sufficient privileges to run the `lmvutil` command. See [“lmvutil Command Authentication,”](#) on page 43.

### Procedure

- To remove the association between a home site and a user, run the `lmvutil` command with the `--deleteUserHomeSite` option.

You can run this command on any View Connection Server instance in the pod federation.

```
lmvutil --deleteUserHomeSite --userName domain\username [--entitlementName name]
```

Option	Description
<code>--userName</code>	Specifies the name of the user. Use the format <code>domain\username</code> .
<code>--entitlementName</code>	(Optional) Specifies the name of a global entitlement. Use this option to remove the association between the home site and a global entitlement for the specified user.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --deleteUserHomeSite --
userName domainCentral\adminCentral
```

- To remove the association between a home site and a group, run the `lmvutil` command with the `--deleteGroupHomeSite` option.

You can run this command on any View Connection Server instance in the pod federation.

```
lmvutil --deleteGroupHomeSite --groupName domain\groupname [--entitlementName name]
```

Option	Description
<code>--groupName</code>	Specifies the name of the group. Use the format <code>domain\groupname</code> .
<code>--entitlementName</code>	(Optional) Specifies the name of a global entitlement. Use this option to remove the association between the home site and a global entitlement for the specified group.

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --deleteGroupHomeSite
--groupName domainCentral\adminCentralGroup
```



## Disable the Cloud Pod Architecture Feature

You can use the `lmvutil` command with the `--uninitialize` option to disable the Cloud Pod Architecture feature. You need to run this command on only one pod in the pod federation. When you disable the Cloud Pod Architecture feature, your entire Cloud Pod Architecture configuration, including sites, home sites, and global entitlements, is deleted.

### Prerequisites

- Become familiar with the `lmvutil` command authentication options and requirements and verify that you have sufficient privileges to run the `lmvutil` command. See [“lmvutil Command Authentication,”](#) on page 43.
- If there are other pods in the pod federation, use the `lmvutil` command with the `--unjoin` option to remove them. See [“Remove a Pod From a Pod Federation,”](#) on page 33.

### Procedure

- ◆ On any View Connection Server instance in the pod federation, run the `lmvutil` command with the `--uninitialize` option.

```
lmvutil --uninitialize
```

For example:

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --uninitialize
```



# Imvutil Command Reference

---

You use the `lmvutil` command line interface to configure and manage a Cloud Pod Architecture implementation.

This chapter includes the following topics:

- [“Imvutil Command Usage,”](#) on page 43
- [“Initializing the Cloud Pod Architecture Feature,”](#) on page 46
- [“Disabling the Cloud Pod Architecture Feature,”](#) on page 47
- [“Managing Pod Federations,”](#) on page 47
- [“Managing Sites,”](#) on page 49
- [“Managing Global Entitlements,”](#) on page 52
- [“Managing Home Sites,”](#) on page 58
- [“Viewing a Cloud Pod Architecture Configuration,”](#) on page 60
- [“Managing SSL Certificates,”](#) on page 65

## Imvutil Command Usage

The syntax of the `lmvutil` command controls its operation.

Use the following form of the `lmvutil` command from a Windows command prompt.

```
lmvutil command_option [additional_option_argument] ...
```

The additional options that you can use depend on the command option.

By default, the path to the `lmvutil` command executable file is `C:\Program Files\VMware\VMware View\Server\tools\bin`. To avoid entering the path on the command line, add the path to your `PATH` environment variable.

## Imvutil Command Authentication

To use the `lmvutil` command to configure and manage a Cloud Pod Architecture environment, you must run the command as a user who has the Administrators role.

You can use View Administrator to assign the Administrators role to a user. For more information, see the *View Administration* document.

The `lmvutil` command includes options to specify the user name, domain, and password to use for authentication.

**Table 5-1.** lmvutil Command Authentication Options

Option	Description
--authAs	Specifies the user name of a View administrator user. Do not use <i>domain\username</i> or user principal name (UPN) format.
--authDomain	Specifies the fully qualified domain name for the View administrator user specified in the --authAs option.
--authPassword	Specifies the password for the View administrator user specified in the --authAs option. Typing "*" instead of a password causes the lmvutil command to prompt for the password and does not leave sensitive passwords in the command history on the command line.

For example, the following lmvutil command logs in the user domainEast\adminEast and initializes the Cloud Pod Architecture feature.

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --initialize
```

You must use the authentication options with all lmvutil command options except for --help and --verbose.

## lmvutil Command Output

The lmvutil command returns 0 when an operation succeeds and a failure-specific non-zero code when an operation fails.

The lmvutil command writes error messages to standard error. When an operation produces output, or when verbose logging is enabled (--verbose option), the lmvutil command writes output to standard output.

The lmvutil command outputs US English only.

## lmvutil Command Options

You use the command options of the lmvutil command to specify the operation that you want to perform.

---

**IMPORTANT** All options are preceded by two dashes (--).

---

**Table 5-2.** lmvutil Command Options

Option	Description
--activatePendingCertificate	Activates a pending SSL certificate. See <a href="#">“Activating a Pending Certificate,”</a> on page 65.
--addGroupEntitlement	Associates a user group with a global entitlement. See <a href="#">“Adding a User or Group to a Global Entitlement,”</a> on page 57.
--addPoolAssociation	Associates a desktop pool with a global entitlement. See <a href="#">“Adding a Desktop Pool to a Global Entitlement,”</a> on page 56.
--addUserEntitlement	Associates a user with a global entitlement. See <a href="#">“Adding a User or Group to a Global Entitlement,”</a> on page 57.
--assignPodToSite	Assigns a pod to a site. See <a href="#">“Assigning a Pod to a Site,”</a> on page 50.
--createGlobalEntitlement	Creates a global entitlement. See <a href="#">“Creating a Global Entitlement,”</a> on page 52.
--createSite	Creates a site. See <a href="#">“Creating a Site,”</a> on page 50.
--createGroupHomeSite	Associates a user group with a home site. See <a href="#">“Configuring a Home Site,”</a> on page 59.

**Table 5-2.** lmvutil Command Options (Continued)

Option	Description
--createPendingCertificate	Creates a pending SSL certificate. See <a href="#">“Creating a Pending Certificate,”</a> on page 65.
--createUserHomeSite	Associates a user with a home site. See <a href="#">“Configuring a Home Site,”</a> on page 59.
--deleteGlobalEntitlement	Deletes a global entitlement. See <a href="#">“Deleting a Global Entitlement,”</a> on page 55.
--deleteSite	Deletes a site. See <a href="#">“Deleting a Site,”</a> on page 51.
--deleteGroupHomeSite	Removes the association between a user group and a home site. See <a href="#">“Deleting a Home Site,”</a> on page 59.
--deleteUserHomeSite	Removes the association between a user and a home site. See <a href="#">“Deleting a Home Site,”</a> on page 59.
--editSite	Modifies the name or description of a site. See <a href="#">“Changing a Site Name or Description,”</a> on page 51.
--ejectPod	Removes an unavailable pod from a pod federation. See <a href="#">“Removing a Pod From a Pod Federation,”</a> on page 48.
--help	Lists the lmvutil command options.
--initialize	Initializes the Cloud Pod Architecture feature. See <a href="#">“Initializing the Cloud Pod Architecture Feature,”</a> on page 46.
--join	Joins a pod to a pod federation. See <a href="#">“Joining a Pod to the Pod Federation,”</a> on page 47.
--listAssociatedPools	Lists the desktop pools that are associated with a global entitlement. See <a href="#">“Listing the Desktop Pools in a Global Entitlement,”</a> on page 61.
--listEntitlements	Lists associations between users or user groups and global entitlements. <a href="#">“Listing the Users or Groups in a Global Entitlement,”</a> on page 61.
--listGlobalEntitlements	Lists all global entitlements. See <a href="#">“Listing Global Entitlements,”</a> on page 61.
--listPods	Lists the pods in a Cloud Pod Architecture topology. See <a href="#">“Listing the Pods or Sites in a Cloud Pod Architecture Topology,”</a> on page 64.
--listSites	Lists the sites in a Cloud Pod Architecture topology. See <a href="#">“Listing the Pods or Sites in a Cloud Pod Architecture Topology,”</a> on page 64.
--listUserAssignments	Lists the dedicated desktop pod assignments for a user and global entitlement combination. See <a href="#">“Listing User Assignments,”</a> on page 63.
--removePoolAssociation	Removes the association between a desktop pool and a global entitlement. See <a href="#">“Removing a Desktop Pool From a Global Entitlement,”</a> on page 56.
--resolveUserHomeSite	Shows the effective home site for a user. See <a href="#">“Listing the Effective Home Site for a User,”</a> on page 63.
--removeGroupEntitlement	Removes a user group from a global entitlement. See <a href="#">“Removing a User or Group From a Global Entitlement,”</a> on page 58.
--removeUserEntitlement	Removes a user from a global entitlement. See <a href="#">“Removing a User or Group From a Global Entitlement,”</a> on page 58.

**Table 5-2.** lmvutil Command Options (Continued)

Option	Description
<code>--showGroupHomeSites</code>	Shows all of the home sites for a group. See <a href="#">“Listing the Home Sites for a User or Group,”</a> on page 62.
<code>--showUserHomeSites</code>	Shows all of the home sites for a user. See <a href="#">“Listing the Home Sites for a User or Group,”</a> on page 62.
<code>--uninitialize</code>	Disables the Cloud Pod Architecture feature. See <a href="#">“Disabling the Cloud Pod Architecture Feature,”</a> on page 47.
<code>--unjoin</code>	Removes an available pod from a pod federation. See <a href="#">“Removing a Pod From a Pod Federation,”</a> on page 48.
<code>--updateGlobalEntitlement</code>	Modifies a global entitlement. See <a href="#">“Modifying a Global Entitlement,”</a> on page 54.
<code>--updatePod</code>	Modifies the name or description of a pod. See <a href="#">“Changing a Pod Name or Description,”</a> on page 49.
<code>--verbose</code>	Enables verbose logging. You can add this option to any other option to obtain detailed command output. The <code>lmvutil</code> command writes to standard output.

For `lmvutil` command authentication options, see [“lmvutil Command Authentication,”](#) on page 43.

## Initializing the Cloud Pod Architecture Feature

You can use the `lmvutil` command with the `--initialize` option to initialize the Cloud Pod Architecture feature. When you initialize the Cloud Pod Architecture feature, View sets up the Global Data Layer on each View Connection Server instance in the pod and configured the VIPA interpod communication channel.

### Syntax

```
lmvutil --initialize
```

### Usage Notes

Before you initialize the Cloud Pod Architecture feature, verify that the latest version of View is installed on each View Connection Server instance in the pod and the latest version of View Agent is running on every desktop.

You can run this command on any View Connection Server instance in a pod. You need to run the command only once, on one View Connection Server instance. You do not need to run this command for other pods in a pod federation because all other pods join the initialized pod.

This command returns an error message if the Cloud Pod Architecture feature is already initialized or if the command is unable to complete the operation.

### Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --initialize
```

## Disabling the Cloud Pod Architecture Feature

You can use the `lmvutil` command with the `--uninitialize` option to disable the Cloud Pod Architecture feature.

### Syntax

```
lmvutil --uninitialize
```

### Usage Notes

You must use the `lmvutil` command with the `--unjoin` option to remove any other pods in the pod federation before you run this command.

You need to run this command on only one View Connection Server instance in a pod. You can run the command on any View Connection Server instance in the pod. In a pod federation topology, you need to run this command for only one pod.

This command returns an error message if the Cloud Pod Architecture feature is not initialized, if the command cannot find the pod, or if there are other pods in the pod federation.

### Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --uninitialize
```

## Managing Pod Federations

The `lmvutil` command provides options to configure and modify pod federations.

- [Joining a Pod to the Pod Federation](#) on page 47  
You can use the `lmvutil` command with the `--join` option to join a pod to the pod federation.
- [Removing a Pod From a Pod Federation](#) on page 48  
You can use the `lmvutil` command with the `--unjoin` or `--ejectPod` option to remove a pod from a pod federation.
- [Changing a Pod Name or Description](#) on page 49  
You can use the `lmvutil` command with the `--updatePod` option to update or modify the name or description of a pod.

### Joining a Pod to the Pod Federation

You can use the `lmvutil` command with the `--join` option to join a pod to the pod federation.

#### Syntax

```
lmvutil --join joinServer serveraddress --userName domain\username --password password
```

#### Usage Notes

You must use this command on each pod that you want to join to the pod federation. You can run the command on any View Connection Server instance in the pod that you want to join to the pod federation.

This command returns an error message if you provide invalid credentials, the specified View Connection Server instance does not exist, a pod federation does not exist on the specified join server, or the command cannot complete the join operation.

## Options

You must specify several options when you join a pod to a pod federation.

**Table 5-3.** Options for Joining a Pod to a Pod Federation

Option	Description
<code>--joinServer</code>	Specifies the DNS name or IP address of any View Connection Server instance in any pod that has been initialized or is already part of the pod federation.
<code>--userName</code>	Specifies the name of a View administrator user on the already initialized pod. Use the format <i>domain\username</i> .
<code>--password</code>	Specifies the password of the user specified in the <code>--userName</code> option.

## Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --join --joinServer
123.456.789.1 --userName domainCentral\adminCentral --password secret123
```

## Removing a Pod From a Pod Federation

You can use the `lmvutil` command with the `--unjoin` or `--ejectPod` option to remove a pod from a pod federation.

### Syntax

```
lmvutil --unjoin
lmvutil --ejectPod --pod pod
```

### Usage Notes

You typically use the `lmvutil` command with the `--unjoin` option to remove a pod from a pod federation. You can run the command on any View Connection Server instance in the pod.

You use the `lmvutil` command with the `--ejectPod` option only to remove a pod that is no longer available, for example, in the case of a hardware failure. You can perform this operation on any pod in the pod federation.

---

**IMPORTANT** In most circumstances, you should use the `lmvutil` command with the `--unjoin` option to remove a pod from a pod federation.

---

These commands return an error message if the Cloud Pod Architecture feature is not initialized, the pod is not joined to a pod federation, or if the commands are unable to perform specified operations.

### Options

When you use the `--ejectPod` option, you use the `--pod` option to identify the pod to be removed from the pod federation.

### Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --unjoin

lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --ejectPod --pod "East Pod
1"
```



## Changing a Pod Name or Description

You can use the `lmvutil` command with the `--updatePod` option to update or modify the name or description of a pod.

### Syntax

```
lmvutil --updatePod --podName podname [--newPodName podname] [--description text]
```

### Usage Notes

This command returns an error message if the Cloud Pod Architecture feature is not initialized or if the command is unable to find or update the pod.

### Options

You can specify these options when you update a pod name or description.

**Table 5-4.** Options for Changing a Pod Name or Description

Option	Description
<code>--podName</code>	Specifies the name of the pod to update.
<code>--newPodName</code>	(Optional) Specifies the new name for the pod. A pod name can contain between 1 and 64 characters.
<code>--description</code>	(Optional) Provides a description of the site. The description can contain between 1 and 1024 characters.

### Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*"
--updatePod --podName "East Pod 1" --newPodName "East Pod 2"
```

## Managing Sites

You can use `lmvutil` command options to create, modify, and delete Cloud Pod Architecture sites. A site is a grouping of View pods.

- [Creating a Site](#) on page 50  
You can use the `lmvutil` command with the `--createSite` option to create a site in a Cloud Pod Architecture topology.
- [Assigning a Pod to a Site](#) on page 50  
You use the `lmvutil` command with the `--assignPodToSite` option to assign a pod to a site.
- [Changing a Site Name or Description](#) on page 51  
You can use the `lmvutil` command with the `--editSite` option to edit the name or description of a site.
- [Deleting a Site](#) on page 51  
You can use the `lmvutil` command with the `--deleteSite` option to delete a site.

## Creating a Site

You can use the `lmvutil` command with the `--createSite` option to create a site in a Cloud Pod Architecture topology.

### Syntax

```
lmvutil --createSite --siteName sitename [--description text]
```

### Usage Notes

This command returns an error message if the Cloud Pod Architecture feature is not initialized, the specified site already exists, or the command cannot create the site.

### Options

You can specify these options when you create a site.

**Table 5-5.** Options for Creating a Site

Option	Description
<code>--siteName</code>	Specifies the name of the new site. The site name can contain between 1 and 64 characters.
<code>--description</code>	(Optional) Provide a description of the site. The description can contain between 1 and 1024 characters.

### Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --createSite
--siteName "Eastern Region"
```

## Assigning a Pod to a Site

You use the `lmvutil` command with the `--assignPodToSite` option to assign a pod to a site.

### Syntax

```
lmvutil --assignPodToSite --podName podname --siteName sitename
```

### Usage Notes

Before you can assign a pod to a site, you must create the site. See [“Creating a Site,”](#) on page 50.

This command returns an error message if the Cloud Pod Architecture feature is not initialized, the command cannot find the specified pod or site, or if the command cannot assign the pod to the site.

### Options

You must specify these options when you assign a pod to a site.

**Table 5-6.** Options for Assigning a Pod to a Site

Option	Description
<code>--podName</code>	Specifies the name of the pod to assign to the site.
<code>--siteName</code>	Specifies the name of the site.

You can use the `lmvutil` command with the `--listPods` option to list the names of the pods in a Cloud Pod Architecture topology. See [“Listing the Pods or Sites in a Cloud Pod Architecture Topology,”](#) on page 64.

## Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*"
--assignPodToSite --podName "East Pod 1" --siteName "Eastern Region"
```

## Changing a Site Name or Description

You can use the `lmvutil` command with the `--editSite` option to edit the name or description of a site.

### Syntax

```
lmvutil --editSite --siteName sitename [--newSiteName sitename] [--description text]
```

### Usage Notes

This command returns an error message if the specified site does not exist or if the command cannot find or update the site.

### Options

You can specify these options when you change a site name or description.

**Table 5-7.** Options for Changing a Site Name or Description

Option	Description
<code>--siteName</code>	Specifies the name of the site to edit.
<code>--newSiteName</code>	(Optional) Specifies a new name for the site. The site name can contain between 1 and 64 characters.
<code>--description</code>	(Optional) Specifies the site description text. The description can contain between 1 and 1024 characters.

## Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --editSite
--siteName "Eastern Region" --newSiteName "Western Region"
```

## Deleting a Site

You can use the `lmvutil` command with the `--deleteSite` option to delete a site.

### Syntax

```
lmvutil --deleteSite --sitename sitename
```

### Usage Notes

This command returns an error message if the specified site does not exist or if the command cannot find or delete the site.

### Options

You use the `--sitename` option to specify the name of the site to delete.

## Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*"
--deleteSite --sitename "Eastern Region"
```

## Managing Global Entitlements

You can use `lmvutil` command options to create, modify, and list global entitlements in a Cloud Pod Architecture environment. Global entitlements link users to desktops, regardless of where the desktops are located in the pod federation. They also determine how the Cloud Pod Architecture feature allocates desktops to those users.

- [Creating a Global Entitlement](#) on page 52

You can use the `lmvutil` command with the `--createGlobalEntitlement` to create a global entitlement. A global entitlement links together users and desktops, regardless of where the desktops are located in the pod federation. Global entitlements also include policies that determine how the Cloud Pod Architecture feature allocates desktops to those users.

- [Modifying a Global Entitlement](#) on page 54

You can use the `lmvutil` command with the `--updateGlobalEntitlement` option to modify the scope, description, and other attributes of a global entitlement.

- [Deleting a Global Entitlement](#) on page 55

You can use the `lmvutil` command with the `--deleteGlobalEntitlement` option to delete a global entitlement.

- [Adding a Desktop Pool to a Global Entitlement](#) on page 56

You can use the `lmvutil` command with the `--addPoolAssociation` option to add a desktop pool to a global entitlement.

- [Removing a Desktop Pool From a Global Entitlement](#) on page 56

You can use the `lmvutil` command with the `--removePoolAssociation` option to remove a desktop pool from a global entitlement.

- [Adding a User or Group to a Global Entitlement](#) on page 57

You can use the `lmvutil` command with the `--addUserEntitlement` or `--addGroupEntitlement` option to add a user or group to a global entitlement.

- [Removing a User or Group From a Global Entitlement](#) on page 58

You can use the `lmvutil` command with the `--removeUserEntitlement` or `--removeGroupEntitlement` option to remove a user or group from a global entitlement.

## Creating a Global Entitlement

You can use the `lmvutil` command with the `--createGlobalEntitlement` to create a global entitlement. A global entitlement links together users and desktops, regardless of where the desktops are located in the pod federation. Global entitlements also include policies that determine how the Cloud Pod Architecture feature allocates desktops to those users.

### Syntax

```
lmvutil --createGlobalEntitlement -entitlementName name --scope scope
{--isDedicated | --isFloating} [--description text] [--disabled]
[--fromHome] [--multipleSessionAutoClean] [--requireHomeSite] [--defaultProtocol value]
[--preventProtocolOverride] [--allowReset]
```

### Usage Notes

You can use this command on any View Connection Server instance in a pod federation. View stores new data in the Global Data Layer and replicates that data in all pods in the pod federation.

This command returns an error message if the global entitlement already exists, the scope is invalid, the Cloud Pod Architecture feature is not initialized, or the command cannot create the global entitlement.

## Options

You can specify these options when you create a global entitlement.

**Table 5-8.** Options for Creating Global Entitlements

Option	Description
<code>--entitlementName</code>	Specifies the name of the global entitlement. The name can contain between 1 and 64 characters. The global entitlement name appears in the list of available entitlements for the user in Horizon Client.
<code>--scope</code>	Specifies the scope of the global entitlement. Valid values are as follows: <ul style="list-style-type: none"> <li>■ ANY - View looks for desktops on any pod in the pod federation.</li> <li>■ SITE - View looks for desktops only on pods within the same site as the pod to which the user is connected.</li> <li>■ LOCAL - View looks for desktops only in the pod to which the user is connected.</li> </ul>
<code>--isDedicated</code>	Creates a dedicated entitlement. A dedicated entitlement can contain only dedicated desktop pools. To create a floating entitlement, use the <code>--isFloating</code> option. An entitlement can be either dedicated or floating. You cannot specify the <code>--isDedicated</code> option with the <code>--multipleSessionAutoClean</code> option.
<code>--isFloating</code>	Creates a floating entitlement. A floating entitlement can contain only floating desktop pools. To create a dedicated entitlement, specify the <code>--isDedicated</code> option. An entitlement can be either floating or dedicated.
<code>--disabled</code>	(Optional) Creates the global entitlement in the disabled state.
<code>--description</code>	(Optional) Specifies a description of the global entitlement. The description can contain between 1 and 1024 characters.
<code>--fromHome</code>	(Optional) If the user has a home site, causes View to look for desktops on the user's home site. If the user does not have a home site, View begins searching for desktops on the site to which the user is currently connected.
<code>--multipleSessionAutoClean</code>	(Optional) Logs off extra user sessions for the same entitlement. Multiple floating desktop sessions can occur when a pod that contains a session goes offline, the user logs in again and starts another session, and the problem pod comes back online with the original session. When multiple sessions occur, Horizon Client prompts the user to select a session. This option determines what happens to sessions that the user does not select. If you do not specify this option, users must manually end their own extra sessions, either by logging off in Horizon Client or by launching the sessions and logging them off.
<code>--requireHomeSite</code>	(Optional) Causes the global entitlement to be available only if the user has a home site. This option is applicable only when the <code>--fromHome</code> option is also specified.
<code>--defaultProtocol</code>	(Optional) Specifies a default display protocol for desktops in the global entitlement. Valid values are RDP and PCOIP.
<code>--preventProtocolOverride</code>	(Optional) When this option is specified, users cannot override the default display protocol for desktops in the global entitlement. If you do not specify this option, users can override the default display protocol.
<code>--allowReset</code>	(Optional) When this option is specified, users can reset desktops in the global entitlement. If you do not specify this option, users cannot reset desktops.

## Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --createGlobalEntitlement --
entitlementName "Agent Sales" --scope LOCAL
--isDedicated
```

## Modifying a Global Entitlement

You can use the `lmvutil` command with the `--updateGlobalEntitlement` option to modify the scope, description, and other attributes of a global entitlement.

### Syntax

```
lmvutil --updateGlobalEntitlement --entitlementName name
[--scope scope] [--description text] [--disabled] [--enabled]
[--fromHome] [--disableFromHome] [--multipleSessionAutoClean] [--disableMultipleSessionAutoClean]
[--requireHomeSite] [--disableRequireHomeSite] [--defaultProtocol value]
```

### Usage Notes

You can use this command on any View Connection Server instance in a pod federation. View stores new data in the Global Data Layer and replicates that data among all pods in the pod federation.

This command returns an error message if the global entitlement does not exist, the scope is invalid, the Cloud Pod Architecture feature is not initialized, or the command cannot update the global entitlement.

### Options

You can specify these options when you modify a global entitlement.

**Table 5-9.** Options for Modifying Global Entitlements

Option	Description
<code>--entitlementName</code>	Specifies the name of the global entitlement to modify.
<code>--scope</code>	Specifies the scope of the global entitlement. Valid values are as follows: <ul style="list-style-type: none"> <li>■ ANY - View looks for desktops on any pod in the pod federation.</li> <li>■ SITE - View looks for desktops only on pods within the same site as the pod to which the user is connected.</li> <li>■ LOCAL - View looks for desktops only in the pod to which the user is connected.</li> </ul>
<code>--description</code>	(Optional) Specifies a description of the global entitlement. The description can contain between 1 and 1024 characters.
<code>--disabled</code>	(Optional) Disables a previously enabled global entitlement.
<code>--enabled</code>	(Optional) Enables a previously disabled global entitlement.
<code>--fromHome</code>	(Optional) If the user has a home site, causes View to look for desktops on the user's home site. If the user does not have a home site, View begins searching for desktops on the site to which the user is currently connected.
<code>--disableFromHome</code>	(Optional) Disables the <code>--fromHome</code> option functionality if the <code>--fromHome</code> option was previously specified for the global entitlement.

**Table 5-9.** Options for Modifying Global Entitlements (Continued)

Option	Description
<code>--multipleSessionAutoClean</code>	(Optional) Logs off extra user sessions for the same entitlement. Multiple floating desktop sessions can occur when a pod that contains a session goes offline, the user logs in again and starts another session, and the problem pod comes back online with the original session. When multiple sessions occur, Horizon Client prompts the user to select a session. This option determines what happens to sessions that the user does not select. If you do not specify this option, users must manually end their own extra sessions, either by logging off in Horizon Client or by launching the sessions and logging them off.
<code>--disableMultipleSessionAutoClean</code>	(Optional) Disables the <code>--multipleSessionAutoClean</code> option functionality if the <code>--multipleSessionAutoClean</code> option was previously specified for the global entitlement.
<code>--requireHomeSite</code>	(Optional) Causes the global entitlement to be available only if the user has a home site. This option is applicable only when the <code>--fromHome</code> option is also specified.
<code>--disableRequireHomeSite</code>	(Optional) Disables the <code>--requireHomeSite</code> option functionality if the <code>--requireHomeSite</code> option was previously specified for the global entitlement.
<code>--defaultProtocol</code>	(Optional) Specifies a default display protocol for desktops in the global entitlement. Valid values are RDP and PCOIP.

### Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --updateGlobalEntitlement --entitlementName "Agent Sales" --scope ANY --isDedicated
```

## Deleting a Global Entitlement

You can use the `lmvutil` command with the `--deleteGlobalEntitlement` option to delete a global entitlement.

### Syntax

```
lmvutil --deleteGlobalEntitlement --entitlementName name
```

### Command Usage

This command returns an error message if the specified global entitlement does not exist, the Cloud Pod Architecture feature is not initialized, or the command cannot delete the global entitlement.

### Options

You use the `--entitlementName` option to specify the name of the global entitlement to delete.

### Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --deleteGlobalEntitlement --entitlementName "Agent Sales"
```

## Adding a Desktop Pool to a Global Entitlement

You can use the `lmvutil` command with the `--addPoolAssociation` option to add a desktop pool to a global entitlement.

### Syntax

```
lmvutil --addPoolAssociation --entitlementName name --poolId poolid
```

### Usage Notes

You must perform this command on a View Connection Server instance in the pod that contains the desktop pool. For example, if `pod1` contains the desktop pool to associate with the global entitlement, you must run the command on a View Connection Server instance that resides in `pod1`.

Repeat this command for each desktop pool that you want to become part of the global entitlement. You can add a particular desktop pool to only one global entitlement.

This command returns an error message if the Cloud Pod Architecture feature is not initialized, the specified entitlement does not exist, the desktop pool is already associated with the specified entitlement, the desktop pool does not exist, or the command cannot add the desktop pool to the global entitlement.

### Options

You can specify these options when you add a desktop pool to a global entitlement.

**Table 5-10.** Options for Adding a Desktop Pool to a Global Entitlement

Option	Description
<code>--entitlementName</code>	Specifies the name of the global entitlement.
<code>--poolID</code>	Specifies the ID of the desktop pool to add to the global entitlement. The pool ID must match the desktop pool name as it appears on the pod.

### Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --addPoolAssociation --entitlementName "Agent Sales" --poolId "Sales B"
```

## Removing a Desktop Pool From a Global Entitlement

You can use the `lmvutil` command with the `--removePoolAssociation` option to remove a desktop pool from a global entitlement.

### Syntax

```
lmvutil --removePoolAssociation --entitlementName name --poolID poolid
```

### Usage Notes

This command returns an error message if the Cloud Pod Architecture feature is not initialized, the specified global entitlement or desktop pool does not exist, or if the command cannot remove the desktop pool from the global entitlement.

### Options

You can specify these options when you remove a desktop pool from a global entitlement.



**Table 5-11.** Options for Removing a Desktop Pool from a Global Entitlement

Option	Description
<code>--entitlementName</code>	Specifies the name of the global entitlement.
<code>--poolID</code>	Specifies the ID of the desktop pool to remove from the global entitlement. The pool ID must match the desktop pool name as it appears on the pod.

### Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*"
--removePoolAssociation -entitlementName "Agent Sales" --poolID "Sales B"
```

## Adding a User or Group to a Global Entitlement

You can use the `lmvutil` command with the `--addUserEntitlement` or `--addGroupEntitlement` option to add a user or group to a global entitlement.

### Syntax

```
lmvutil --addUserEntitlement --userName domain\username --entitlementName name
lmvutil --addGroupEntitlement --groupName domain\groupname --entitlementName name
```

### Usage Notes

Repeat this command for each user or group that you want to add to the global entitlement.

These commands return an error message if the specified entitlement, user, or group does not exist or if the command cannot add the user or group to the entitlement.

### Options

You can specify these options when you add a user or group to a global entitlement.

**Table 5-12.** Options for Adding a User or Group to a Global Entitlement

Option	Description
<code>--userName</code>	Specifies the name of a user to add to the global entitlement. Use the format <i>domain\username</i> .
<code>--groupName</code>	Specifies the name of a group to add to the global entitlement. Use the format <i>domain\groupname</i> .
<code>--entitlementName</code>	Specifies the name of the global entitlement to which to add the user or group.

### Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --addUserEntitlement --
userName domainCentral\adminCentral --entitlementName "Agent Sales"
```

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*"
--addGroupEntitlement --groupName domainCentral\adminCentralGroup --entitlementName "Agent Sales"
```

## Removing a User or Group From a Global Entitlement

You can use the `lmvutil` command with the `--removeUserEntitlement` or `--removeGroupEntitlement` option to remove a user or group from a global entitlement.

### Syntax

```
lmvutil --removeUserEntitlement --userName domain\username --entitlementName name
lmvutil --removeGroupEntitlement --groupName domain\groupname --entitlementName name
```

### Usage Notes

These commands return an error message if the Cloud Pod Architecture feature is not initialized, if the specified user name, group name, or entitlement does not exist, or if the command cannot remove the user or group from the entitlement.

### Options

You must specify these options when you remove a user or group from a global entitlement.

**Table 5-13.** Options for Removing a User or Group From a Global Entitlement

Option	Description
<code>--userName</code>	Specifies the name of a user to remove from the global entitlement. Use the format <i>domain\username</i> .
<code>--groupName</code>	Specify the name of a group to remove from the global entitlement. Use the format <i>domain\groupname</i> .
<code>--entitlementName</code>	Specifies the name of the global entitlement from which to remove the user or group.

### Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*"
--removeUserEntitlement --userName domainCentral\adminCentral --entitlementName "Agent Sales"
```

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*"
--removeGroupEntitlement --groupName domainCentral\adminCentralGroup --entitlementName "Agent Sales"
```

## Managing Home Sites

You can use `lmvutil` command options to create, modify, delete, and list home sites. You associate users or groups with a home site to restrict their choice of desktops to a particular site.

- [Configuring a Home Site](#) on page 59

You can use the `lmvutil` command with the `--createUserHomeSite` or `--createGroupHomeSite` option create a home site for a user or group. You can also use these options to associate a home site with a global entitlement.

- [Deleting a Home Site](#) on page 59

You can use the `lmvutil` command with the `--deleteUserHomeSite` or `--deleteGroupHomeSite` option to remove the association between a user or group and a home site.

## Configuring a Home Site

You can use the `lmvutil` command with the `--createUserHomeSite` or `--createGroupHomeSite` option create a home site for a user or group. You can also use these options to associate a home site with a global entitlement.

### Syntax

```
lmvutil --createUserHomeSite --userName domain\username --siteName name [--entitlementName name]
lmvutil --createGroupHomeSite --groupName domain\groupname --siteName name [--entitlementName name]
```

### Usage Notes

You must create a site before you can configure it as a home site. See [“Creating a Site,”](#) on page 50.

These commands return an error message if the Cloud Pod Architecture feature is not initialized, the specified user or group does not exist, the specified site does not exist, the specified entitlement does not exist, or the command cannot create the home site.

### Options

You can specify these options when you create a home site for a user or group.

**Table 5-14.** Options for Creating a Home Site for a User or Group

Option	Description
<code>--userName</code>	Specifies the name of a user to associate with the home site. Use the format <i>domain\username</i> .
<code>--groupName</code>	Specifies the name of a group to associate with the home site. Use the format <i>domain\groupname</i> .
<code>--siteName</code>	Specifies the name of the site to associate with the user or group as the home site.
<code>--entitlementName</code>	(Optional) Specifies the name of a global entitlement to associate with the home site. When a user selects the specified global entitlement, the home site overrides the user's own home site. If you do not specify this option, the command creates a global user or group home site.

### Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --createUserHomeSite --
userName domainEast\adminEast --siteName "Eastern Region" --entitlementName "Agent Sales"
```

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*"
--createGroupHomeSite --groupName domainEast\adminEastGroup --siteName "Eastern Region"
--entitlementName "Agent Sales"
```

## Deleting a Home Site

You can use the `lmvutil` command with the `--deleteUserHomeSite` or `--deleteGroupHomeSite` option to remove the association between a user or group and a home site.

### Syntax

```
lmvutil --deleteUserHomeSite --userName domain\username [--entitlementName name]
lmvutil --deleteGroupHomeSite --groupName domain\groupname [--entitlementName name]
```

## Usage Notes

These commands return an error message if the specified user or group does not exist, the specified entitlement does not exist, or if the command cannot delete the home site setting.

## Options

You can specify these options when you remove the association between a user or group and a home site.

**Table 5-15.** Options for Deleting a Home Site

Option	Description
<code>--userName</code>	Specifies the name of a user. Use the format <i>domain\username</i> .
<code>--groupName</code>	Specifies the name of a group. Use the format <i>domain\groupname</i> .
<code>--entitlementName</code>	(Optional) Specifies the name of a global entitlement. You can use this option to remove the association between the home site and a global entitlement for the specified user or group.

## Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --deleteUserHomeSite --
userName domainEast\adminEast
```

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*"
--deleteGroupHomeSite --groupName domainEast\adminEastGroup
```

## Viewing a Cloud Pod Architecture Configuration

You can use `lmvutil` command options to list information about a Cloud Pod Architecture configuration.

- [Listing Global Entitlements](#) on page 61  
You can use the `lmvutil` command with the `--listGlobalEntitlements` option to list all global entitlements.
- [Listing the Desktop Pools in a Global Entitlement](#) on page 61  
You can use the `lmvutil` command with the `--listAssociatedPools` option to list the desktop pools that are associated with a specific global entitlement.
- [Listing the Users or Groups in a Global Entitlement](#) on page 61  
You can use the `lmvutil` command with the `--listEntitlements` option to list all of the users or groups associated with a specific global entitlement.
- [Listing the Home Sites for a User or Group](#) on page 62  
You can use the `lmvutil` command with the `--showUserHomeSites` or `--showGroupHomeSites` option to list all of the configured home sites for a specific user or group.
- [Listing the Effective Home Site for a User](#) on page 63  
Because you can assign home sites to users and groups and to global entitlements, it is possible to configure more than one home site for a specific user. You can use the `lmvutil` command with the `--resolveUserHomeSite` option enables you to determine the effective home site for a specific user.
- [Listing User Assignments](#) on page 63  
You can use the `lmvutil` command with the `--listUserAssignments` option to list the dedicated desktop pod assignments for a user and global entitlement combination.

- [Listing the Pods or Sites in a Cloud Pod Architecture Topology](#) on page 64

You can use the `lmvutil` command with the `--listPods` or `--listSites` option to view the pods or sites in your Cloud Pod Architecture topology.

## Listing Global Entitlements

You can use the `lmvutil` command with the `--listGlobalEntitlements` option to list all global entitlements.

### Syntax

```
lmvutil --listGlobalEntitlements
```

### Usage Notes

This command returns an error message if the Cloud Pod Architecture feature is not initialized or if the command cannot list the global entitlements.

### Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --listGlobalEntitlements
```

## Listing the Desktop Pools in a Global Entitlement

You can use the `lmvutil` command with the `--listAssociatedPools` option to list the desktop pools that are associated with a specific global entitlement.

### Syntax

```
lmvutil --listAssociatedPools --entitlementName name
```

### Usage Notes

This command returns an error message if the Cloud Pod Architecture feature is not initialized or if the specified global entitlement does not exist.

### Options

You use the `--entitlementName` option to specify the name of the global entitlement for which to list the associated desktop pools.

### Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --listAssociatedPools --entitlementName "Agent Sales"
```

## Listing the Users or Groups in a Global Entitlement

You can use the `lmvutil` command with the `--listEntitlements` option to list all of the users or groups associated with a specific global entitlement.

### Syntax

```
lmvutil --listEntitlements {--userName domain\username | --groupName domain\groupname | --entitlementName name}
```

## Usage Notes

This command returns an error message if the Cloud Pod Architecture feature is not initialized or if the specified user, group, or entitlement does not exist.

## Options

You can specify these options when you list global entitlement associations.

**Table 5-16.** Options for Listing Global Entitlement Associations

Option	Description
--userName	Specifies the name of the user for whom you want to list global entitlements. Use the format <i>domain\username</i> . When you use this option, the command lists all the global entitlements associated with the specified user.
--groupName	Specifies the name of the group for which you want to list global entitlements. Use the format <i>domain\groupname</i> . When you use this option, the command lists all of the global entitlements associated with the specified group.
--entitlementName	Specifies the name of a global entitlement. When you use this option, the command lists all of the users and groups in the specified global entitlement.

## Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --listEntitlements
--userName example\adminEast
```

## Listing the Home Sites for a User or Group

You can use the `lmvutil` command with the `--showUserHomeSites` or `--showGroupHomeSites` option to list all of the configured home sites for a specific user or group.

## Syntax

```
lmvutil --showUserHomeSites --userName domain\username [--entitlementName name]
lmvutil --showGroupHomeSites --groupName domain\groupname [--entitlementName name]
```

## Usage Notes

These commands return an error message if the Cloud Pod Architecture feature is not initialized or if the specified user, group, or global entitlement does not exist.

## Options

You can specify these options when you list the home sites for a user or group.

**Table 5-17.** Options for Listing the Home Sites for a User or Group

Option	Description
--userName	Specifies the name of a user. Use the format <i>domain\username</i> .
--groupName	Specifies the name of a group. Use the format <i>domain\groupname</i> .
--entitlementName	(Optional) Specifies the name of a global entitlement. Use this option if you want to show the home sites for a user or group and global entitlement combination.

## Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --showUserHomeSites --
userName example\adminEast
```

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --showGroupHomeSites --
groupName example\adminEastGroup
```

## Listing the Effective Home Site for a User

Because you can assign home sites to users and groups and to global entitlements, it is possible to configure more than one home site for a specific user. You can use the `lmvutil` command with the `--resolveUserHomeSite` option enables you to determine the effective home site for a specific user.

### Syntax

```
lmvutil --resolveUserHomeSite --entitlementName name --userName domain\username
```

### Usage Notes

This command returns an error message if the Cloud Pod Architecture feature is not initialized or if the specified global entitlement or user does not exist.

### Options

You must specify these options when you list the effective home site for a user.

**Table 5-18.** Options for Listing the Effective Home Site for a User

Option	Description
<code>--entitlementName</code>	Specifies the name of a global entitlement. This option enables you to determine the effective home site for a user and global entitlement combination, which might be different from the home site that is configured for the user.
<code>--userName</code>	Specifies the name of the user whose home site you want to list. Use the format <i>domain\username</i> .

### Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*"
--resolveUserHomeSite --userName domainEast\adminEast
```

## Listing User Assignments

You can use the `lmvutil` command with the `--listUserAssignments` option to list the dedicated desktop pod assignments for a user and global entitlement combination.

### Syntax

```
lmvutil --listUserAssignments {--userName domain\username | --entitlementName name | --podName
name | --siteName name}
```

## Usage Notes

The data produced by this command is managed internally by the Cloud Pod Architecture brokering software.

This command returns an error if the Cloud Pod Architecture feature is not initialized or if the command cannot find the specified user, global entitlement, pod, or site.

## Options

You must specify one of the following options when you list user assignments.

**Table 5-19.** Options for Listing User Assignments

Option	Description
<code>--userName</code>	Specifies the name of the user for whom you want to list assignments. Use the format <i>domain\username</i> . When you use this option, the command lists the global entitlement, pod, and site assignments for the specified user.
<code>--entitlementName</code>	Specifies the name of a global entitlement. When you use this option, the command lists the users assigned to the specified global entitlement.
<code>--podName</code>	Specifies the name of a global entitlement. When you use this option, the command lists the users assigned to the specified global entitlement.
<code>--siteName</code>	Specifies the name of a site. When you use this option, the commands lists the users assigned to the specified site.

## Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --listUserAssignments --podName "East Pod 1"
```

## Listing the Pods or Sites in a Cloud Pod Architecture Topology

You can use the `lmvutil` command with the `--listPods` or `--listSites` option to view the pods or sites in your Cloud Pod Architecture topology.

## Syntax

```
lmvutil --listPods
lmvutil --listSites
```

## Usage Notes

These commands return an error message if the Cloud Pod Architecture feature is not initialized or if the command cannot list the pods or sites.

## Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --listPods
```

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*" --listSites
```



## Managing SSL Certificates

You can use `lmvutil` command options to create and activate pending SSL certificates in a Cloud Pod Architecture environment.

The Cloud Pod Architecture feature uses signed certificates for bidirectional SSL to protect and validate the VIPA interpod communication channel. The certificates are distributed in the Global Data Layer. The Cloud Pod Architecture feature replaces these certificates every seven days.

To change a certificate for a specific View Connection Server instance, you create a pending certificate, wait for the Global Data Layer replication process to distribute the certificate to all View Connection Server instances, and then activate the certificate.

The `lmvutil` command certificate options are intended for use only if a certificate becomes compromised and a View administrator wants to update the certificate sooner than seven days. These options affect only the View Connection Server instance on which they are run. To change all certificates, you must run the options on every View Connection Server instance.

- [Creating a Pending Certificate](#) on page 65

You can use the `lmvutil` command with the `--createPendingCertificate` option to create a pending SSL certificate.

- [Activating a Pending Certificate](#) on page 65

You can use the `lmvutil` command with the `--activatePendingCertificate` option to activate a pending certificate.

### Creating a Pending Certificate

You can use the `lmvutil` command with the `--createPendingCertificate` option to create a pending SSL certificate.

#### Syntax

```
LMVUtil --createPendingCertificate
```

#### Usage Notes

This command returns an error message if the Cloud Pod Architecture feature is not initialized or if the command cannot create the certificate.

#### Example

```
LMVUtil --authAs adminEast --authDomain domainEast --authPassword "*"
--createPendingCertificate
```

### Activating a Pending Certificate

You can use the `lmvutil` command with the `--activatePendingCertificate` option to activate a pending certificate.

#### Syntax

```
lmvutil --activatePendingCertificate
```

## Usage Notes

You must use the `lmvutil` command with the `--createPendingCertificate` option to create a pending certificate before you can use this command. Wait for the Global Data Layer replication process to distribute the certificate to all View Connection Server instances before you activate the pending certificate. VIPA connection failures and resulting brokering problems can occur if you activate a pending certificate before it has been fully replicated to all View Connection Server instances.

This command returns an error message if the Cloud Pod Architecture feature is not initialized or if the command cannot activate the certificate.

## Example

```
lmvutil --authAs adminEast --authDomain domainEast --authPassword "*"
--activatePendingCertificate
```

# Index

## A

allocating desktops **10**  
architectural overview **7**

## C

configuration  
tasks **15**  
viewing **29, 60**

## D

desktop sessions **31**  
desktop pools in global entitlements **34, 35**

## G

global entitlements  
adding a desktop pool **56**  
adding a user or group **57**  
creating **18, 27, 52**  
deleting **39, 55**  
introduction **10**  
listing **61**  
listing desktop pools **61**  
listing users and groups **61**  
managing **52**  
modifying **38, 54**  
removing a desktop pool **56**  
removing a user or group **58**

Global Data Layer **8**

glossary **5**

## H

home sites  
assigning **22**  
creating **59**  
effective **63**  
introduction **11**  
listing **62, 63**  
managing **58**  
removing assignments **40**  
removing associations **59**

## I

initializing **15, 25, 46**  
intended audience **5**  
introduction **7**

## L

limitations **8**  
lmvutil command  
authenticating **43**  
command options **44**  
introduction **43**  
output **44**  
syntax **43**

## M

management interfaces **8, 29**

## N

naming pods **17**

## P

pending certificates  
activating **65**  
creating **65**  
pod federations  
joining pods **16, 25, 47**  
managing **47**  
removing pods **48**  
viewing health **31**  
pod names **49**

## S

scenario **24**  
scope policy settings **10**  
security considerations **13**  
sites  
adding pods **33**  
assigning pods **50**  
changing a name or description **51**  
creating **21, 26, 50**  
deleting **51**  
introduction **9**  
managing **49**  
SSL certificates **65**

## T

TCP port requirements **13**  
testing **23**  
topology  
designing **9, 25**

limits **12**  
viewing **64**

## **U**

unitializing **41, 47**  
users in global entitlements **36, 37**

## **V**

View URL **28**  
VIPA interpod communication channel **8**